MUFITS

Reservoir Simulation Software
version 2024.A

Reference Manual

Author: Andrey Afanasyev

Date: 5 May, 2024

Moscow, Russia

Contents

1. Keywords **7**

1.1. Using keywords 7
1.2. Comments 7
1.3. Repeated data items 8
1.4. Default values 9
1.5. List of operations available in arithmetic and logical expressions 10
1.6. Basic keywords 13
1.7. Kernel keywords of the H-program 39
1.8. BINMIXT keywords 523
1.9. BLACKOIL keywords 539
1.10. COMPS keywords 662
1.11. GASSTORE keywords 824
1.12. MAGFLUID keywords 899
1.13. SIMPLMOD keywords 905
1.14. T2EOS1 keywords 910
1.15. USEREOS keywords 916
1.16. U-program. Section CSV-ENDCSV 918
1.17. U-program. Section IAPWS-ENDIAPWS 928
1.18. U-program. Section OPTIM-ENDOPTIM 931
1.19. U-program. Section SCHED-ENDSCHED 973
1.20. U-program. Section SATFC-ENDSATFC 996
1.21. PVT program (I-program) keywords 1009

2. Mnemonics **1013**

2.1. Using Mnemonics 1013
2.2. Kernel mnemonics 1014
2.3. BINMIXT mnemonics 1046
2.4. BLACKOIL mnemonics 1051
2.5. COMPS mnemonics 1067
2.6. GASSTORE mnemonics 1087
2.7. SIMPLMOD mnemonics 1098
2.8. T2EOS1 mnemonics 1099
2.9. USEREOS mnemonics 1102

3. Options vs EOS modules **1105**

3.1. BINMIXT 1107
3.2. BLACKOIL 1108
3.3. COMPS 1109
3.4. GASSTORE 1110
3.5. SIMPLMOD 1111
3.6. T2EOS1 1112
3.7. USEREOS 1113

4. Thermodynamic states of fluid **1114**

4.1. Overview 1114
4.2. BINMIXT 1114
4.3. BLACKOIL 1114
4.4. COMPS 1115
4.5. GASSTORE 1115
4.6. SIMPLMOD 1115
4.7. T2EOS1 1115
4.8. USEREOS 1116

5. Well rates **1117**
5.1. Kernel 1117
5.2. BLACKOIL 1117
5.3. COMPS 1118
5.4. GASSTORE 1118

6. Units **1119**

7. Useful Notes **1124**
7.1. Grid blocks indexation in dual-porosity simulations 1124
7.2. Inverse modelling 1126
7.3. Library of components for the COMPS module 1127
7.4. Options for modeling edge-water drive 1129

8. New Developments **1133**
8.1. Release 2024.A 1133
8.2. Release 2022.B 1134
8.3. Release 2022.A 1135
8.4. Release 2021.C 1136
8.5. Release 2021.B 1137
8.6. Release 2021.A 1138
8.7. Release 2020.B 1139
8.8. Release 2020.A 1139
8.9. Release 2019.B 1141
8.10. Release 2017.A 1143

9. Testing **1144**
9.1. Overview 1144
9.2. 1st SPE Comparative Study (BLACKOIL) 1145
9.3. 2nd SPE Comparative Study (BLACKOIL) 1154

9.4. 6th SPE Comparative Study; Water injection (BLACKOIL) . 1164
9.5. 6th SPE Comparative Study; Depletion (BLACKOIL) 1173
9.6. 7th SPE Comparative Study; case 3a (BLACKOIL) 1184
9.7. 7th SPE Comparative Study; case 3b (BLACKOIL) 1193
9.8. 7th SPE Comparative Study; case 4a (BLACKOIL) 1202
9.9. 7th SPE Comparative Study; case 4b (BLACKOIL) 1210
9.10. 9th SPE Comparative Study (BLACKOIL) 1219
9.11. 10th SPE Comparative Study; case 1 (BLACKOIL) 1231
9.12. 10th SPE Comparative Study; case 2 (BLACKOIL) 1238
9.13. The Egg model (BLACKOIL) 1244
9.14. CO_2 injection in the 10th SPE reservoir (BINMIXT) 1251
9.15. Benchmark study for CO_2 storage (BINMIXT) 1263
9.16. Benchmark study for CO_2 storage (GASSTORE) 1270
9.17. Benchmark study for CO_2 storage (BLACKOIL) 1277
9.18. Benchmark study for CO_2 storage (BLACKOIL; VAPOIL option) 1287
9.19. Example-H4 (BINMIXT) 1298
9.20. Radial flow from a CO_2 injection well (GASSTORE) 1313
9.21. Heat sweep in a vertical fracture (T2EOS1) 1319
9.22. Five-spot geothermal production/injection (T2EOS1) 1325
9.23. Test for heat conduction 1333
9.24. CO_2 discharge along a fault zone (GASSTORE) 1337
9.25. CO_2 injection into a 2D-layered brine formation (GASSTORE) 1349
9.26. CO_2 injection into a 2D-layered brine formation (BLACKOIL) 1360
9.27. Test for longitudinal mechanical dispersion of solutes (GASSTORE) 1374
9.28. Test for transverse mechanical dispersion of solutes (GASSTORE) 1381

9.29. Test for longitudinal mechanical dispersion of temperature (GASSTORE) 1386
9.30. Test for transverse mechanical dispersion of temperature (GASSTORE) 1392
9.31. Test for modelling the brittle-ductile transition. Problem 2c (BINMIXT) 1397
9.32. Modelling flue gas injection into a saline aquifer 1404
9.33. Modelling cushion gas substitution with CO_2 1411
9.34. Modelling the porosity waves. 1-D study 1421
9.35. Modelling a CCS scenario (BLACKOIL) 1427

References **1436**

1 Keywords

1.1. Using keywords

The keywords are the instructions in the RUN-file to the simulator. By using keywords the grid, the physical model, the relative permeabilities, the initial and boundary conditions, etc. are specified. The keywords

1. must start in column 1 of RUN-file;
2. must be typed in uppercase letters;
3. consist of up to 8 uppercase letters.

1.2. Comments

The comment lines in the RUN-file are

1. lines beginning with '!' or '–';
2. lines not beginning with an uppercase letter outside a keyword instruction.

By convention we will type all comments in lowercase letters, whereas the keywords must be in uppercase letters.

Examples of commented out lines

```
GRID      ##############################################################################

-- this is a comment line
!  this is a comment line too
   this is another comment line

MAKE      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
```

```
-- cartesian                  These are comment
--    grid    nx  ny  nz        lines
      CART    10  10  1  /

XYZBOUND
!   xmin xmax  ymin ymax  zmin zmax      Also a comment line
    100   200  100   200   0    10    /

...

ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

================================================================================

   By convention we use the following commenting:

   We denote by the repeated symbols the following structural elements of this
      RUN-file:
      '###############' - delimits the sections of the RUN-file.
      '<<<<<<<<<<<<<<' - designates an opening bracket.
      '>>>>>>>>>>>>>>' - designates a closing bracket.
      '^^^^^^^^^^^^^^' - designates the keyword, which affects the order of the
                         data input for all the following keywords (both in the
                         current and in the following sections).
      '==============' - merge the keywords and the comments in paragraphs.
```

1.3. Repeated data items

A data item can be repeated in a keyword a number of times by using asterisk symbol. The number ahead of the asterisk indicates that the data item after the asterisk is repeated this number of times. There must be no spaces ahead of and behind the asterisk.

Example of repeated counts

```
BOX
-- imin-imax  jmin-jmax  kmin-kmax
    1    4     1    3     1    1   /

-- This keyword

PORO
   0.25  0.25  0.25  0.25
   0.25  0.25  0.25  0.25
   0.15  0.15  0.15  0.15 /
```

```
11
12 -- is equivalent to any of the following keywords
13
14 PORO
15    4*0.25
16    4*0.25
17    4*0.15 /
18
19 PORO
20    8*0.25 4*0.15 /
21
22 PORO
23    7*0.25 0.25 2*0.15 2*0.15 /
```

1.4. Default values

There are two options for defaulted parameters

1. by using asterisk. The number ahead of the asterisk followed by a space indicates that this number of data items are not altered;

2. by the slash symbol. The slash in a input line indicates a premature end of input. All parameters after the slash will not be altered whereas in the input line all data items after the slash are ignored.

Example of defaulted parameters

```
1  BOX
2  -- imin-imax  jmin-jmax  kmin-kmax
3       1   5       1    6    1    2   /
4
5  -- keywords
6
7  ================================================================================
8
9  -- The keyword
10
11 BOX
12 -- imin-imax  jmin-jmax  kmin-kmax
13       2*         3    1*    2  /
14
15 -- is equivalent to
16
```

```
17 BOX
18 -- imin-imax  jmin-jmax  kmin-kmax
19     1    5     3    6     2    2  /
20
21 -- because the parameters imin,imax,jmax,kmax are not altered
```

1.5. List of operations available in arithmetic and logical expressions

Some keywords (e.g. ARITH) take arithmetic or logical expressions. The binary operations and function available in such expressions are listed below in the table. The operations of the higher level are performed first. The functions are understood as the operations of the highest level.

List of operations in arithmetic and logical expressions

operation	level	expression
a && b	(1)	if $a > 0$ and $b > 0$, then 1, else 0
a || b	(1)	if $a > 0$ or $b > 0$, then 1, else 0
$a > b$	(2)	if $a > b$, then 1, else 0
$a >= b$	(2)	if $a \geq b$, then 1, else 0
$a < b$	(2)	if $a < b$, then 1, else 0
$a <= b$	(2)	if $a \leq b$, then 1, else 0
$a == b$	(2)	if $a = b$, then 1, else 0
$a! = b$	(2)	if $a \neq b$, then 1, else 0
a+b	(3)	$a + b$
a-b	(3)	$a - b$
$a * b$	(4)	$a * b$
a/b	(4)	a/b
a**b	(5)	a^b
ABS(a)	(–)	abs(a); Computes the absolute value of a
ACOS(a)	(–)	arccos(a); produces arccosine of a
AMEND(a,b)	(–)	If a is not defined, then return b. Otherwise returns a.
ASIN(a)	(–)	arcsin(a); produces arcsine of a
ARG(a,b)	(–)	arg(b/a); Computes the argument in radians of b/a
ATAN(a)	(–)	arctan(a); produces arctangent of a

List of operations in arithmetic and logical expressions		
operation	level	expression
CEILING(a)	(–)	Returns the smallest integer greater than or equal to a
COS(a)	(–)	$\cos(a)$; produces the cosine of a
COTAN(a)	(–)	$\cot(a)$; produces the cotangent of a
EXP(a)	(–)	$\exp(a)$; produces exponential value of a
FLOOR(a)	(–)	Returns the greatest integer less than or equal to a
INSIDE(a;b;c)	(–)	If $b \leq a \leq c$, then 1, else 0
ISNAN(a)	(–)	If a is not defined (NaN), then 1, else 0
LINEXY(a;b)	(–)	$\min \sqrt{(x-a)^2+(y-b)^2}$, where (x, y) belongs to the polyline. Returns the distance to a polyline in projection on the Oxy plane. The polyline must be defined by the POLYLINE keyword.
LOG(a)	(–)	$\log(a)$; produces natural logarithm of a
LOG10(a)	(–)	$\log_{10}(a)$; produces the common logarithm of a
LOGE(a)	(–)	$\log(a)$; produces the natural logarithm of a
MAX(a;b)	(–)	$\max(a; b)$; Returns the maximum of two quantities
MIN(a;b)	(–)	$\min(a; b)$; Returns the minimum of two quantities
MOD(a;b)	(–)	Returns the remainder when a is divided by b
NORMAL(a;b)	(–)	Returns a normally distributed random quantity, where a is the mean (expectation) and b is the dispersion
NOT(a)	(–)	If $a > 0$, then 0, else 1
OUTSIDE(a;b;c)	(–)	If $b \leq a \leq c$, then 0, else 1
POLYGON(a;b)	(–)	$\alpha \min \sqrt{(x-a)^2+(y-b)^2}$, where (x, y) belongs to the polygon. Returns the distance to a polygon in projection on the Oxy plane. The distance is negative ($\alpha = -1$) for a point inside the polygon and positive ($\alpha = 1$) for a point outside the polygon. The polygon must be defined by the POLYLINE keyword
POLYGON0(a;b)	(–)	$\alpha \min \sqrt{(x-a)^2+(y-b)^2}$, where (x, y) belongs to the polygon. Returns the distance to a polygon in projection on the Oxy plane. The distance is 0 ($\alpha = 0$) for a point inside the polygon and positive ($\alpha = 1$) for a point outside the polygon. The polygon must be defined by the POLYLINE keyword

List of operations in arithmetic and logical expressions		
operation	level	expression
POLYLINE(a;b;c)	(–)	min $\sqrt{(x-a)^2+(y-b)^2+(z-c)^2}$, where (x,y,z) belongs to the polyline. Returns the distance to a polyline in 3D space. The polyline must be defined by the POLYLINE keyword
RANDOM()	(–)	Produces a random number between 0 and 1 (uniform distribution)
SIGN(a)	(–)	Returns the sign of a
SIN(a)	(–)	$\sin(a)$; produces the sine of a
SQRT(a)	(–)	$\sqrt{a}$; returns the square root of a
TAN(a)	(–)	$\tan(a)$; produces the tangent of a
USERFUNC(id;a)	(–)	Returns the value of the 1D user-defined function of a. The function ID is the first argument
UF2D(id;a;b)	(–)	Returns the value of the 2D user-defined function of a and b. The function ID is the first argument
UF3D(id;a;b;c)	(–)	Returns the value of the 3D user-defined function of a, b and c. The function ID is the first argument

1.6. Basic keywords

The keywords below are available with every program of the MUFITS software including the H-program (the hydrodynamic simulator) and the U-program (the utilities program).

ALLPROG

The input data after that switch are read by all software programs. Alternative positions of the switch are HPROG, IPROG and UPROG. By default the switch is in the ALLPROG position.

ALLPROG syntax

```
-- in any section of the RUN-file

ALLPROG
```

CMDLINE

Passes a command to the OS terminal (command line). The synonym to this keyword is TERMINAL.

CMDLINE syntax

```
-- in any section of the RUN-file

CMDLINE
   'command_line'  Wait_Flag  Rank  /

================================================================================

   command_line   - Command line to be passed to the terminal (Linux, Mac) or
                    the Windows command line;
      Wait_Flag   - if T (default value) the program waits until end of the
                    command_line execution. If F then the program continues
                    running not waiting for end of the command_line
                    execution;
            Rank  - rank of the process which executes the command line. If
                    Rank=-1 then all processes execute the command_line. The
                    default value is 0.
```

DEFUNITS

Switches the program to units used by default.

DEFUNITS syntax

```
-- in RUNSPEC section or within specific brackets

DEFUNITS
```

ECHO

Enables more information output in the LOG-file.

ECHO syntax

```
1 ECHO
```

END

Terminates the simulation.

END syntax

```
-- H-program: in the SCHEDULE or POST section
-- U-program: in the root section

END

-- the keywords below are not readed.
```

ENDINC

Resumes the input from the main file. This keyword should be specified in an include file (see the keyword INCLUDE).

ENDINC syntax

```
-- in an include file

ENDINC

-- the keywords below will not be readed.
```

FIELD

Switches the program to FIELD units.

FIELD syntax

```
-- in RUNSPEC section or within specific brackets

FIELD
```

FILEUNIT

Checks that the current units are consistent with the simulation units specified in the RUNSPEC section by keyword METRIC, FIELD, LAB or DEFUNITS.

FILEUNIT syntax

```
-- in any section

FILEUNIT
   units  /

================================================================================

     units    - should be METRIC, FIELD, LAB, or DEFUNITS. Checks that the
                current units are consistent with the simulation units
                specified in the RUNSPEC section.
```

HPROG

The input data after that switch are read only by the H-program. Alternative positions of the switch are ALLPROG, IPROG and UPROG. By default the switch is in the ALLPROG position.

HPROG syntax

```
-- in any section of the RUN-file

HPROG
```

IF-ENDIF

Executes encompassed keywords for selected LOOP-ENDLOOP iterations.

IF-ENDIF syntax

```
LOOP
  niter /

-- other keywords

IF
  iter1 iter2 iter3 ... /

-- other keywords

ENDIF

-- other keywords

ENDLOOP

================================================================================

  niter - number of times the keywords encompassed by the LOOP-ENDLOOP brackets
          are repeated;
  iter# - iteration number on which the keywords encompassed by the IF-ENDIF
          brackets are executed.
```

INCLUDE

Includes the content of an include file. The input resumes from the main file if either the end of the include file is reached or the keyword ENDINC is encountered. The path to the include files can be changed by the PATH keyword.

INCLUDE syntax

```
INCLUDE
   'filename' /

================================================================================

     filename - the name of include file from which the input is included.
```

IPROG

The input data after that switch are read only by the I-program. Alternative positions of the switch are ALLPROG, HPROG and UPROG. By default the switch is in the ALLPROG position.

IPROG syntax

```
-- in any section of the RUN-file

IPROG
```

LAB

Switches the program to LAB units.

LAB syntax

```
-- in RUNSPEC section or within specific brackets

LAB
```

LOOP-ENDLOOP

Cycles the encompassed keywords the specified number of times. Both brackets must be in the same input (include) file. The nested loops are available.

LOOP-ENDLOOP syntax

```
LOOP
  niter /

-- other keywords

ENDLOOP

================================================================================

  niter - number of times the keywords encompassed by the LOOP-ENDLOOP brackets
          are repeated
```

MESSAGE

Outputs in the LOG-file a message supplied by user. The message string must be encompassed by apostrophes.

MESSAGE syntax

```
MESSAGE
 'string' /

================================================================================

   string - a string outputted in the LOG-file as a user-supplied message.
```

METRIC

Switches the program to METRIC units.

METRIC syntax

```
-- in RUNSPEC section or within specific brackets (H-program).
-- in U-program

METRIC
```

MSG

Enables messages output in the LOG-file.

MSG syntax

```
1 MSG
```

NOECHO

Disables additional information output in the LOG-file.

NOECHO syntax

```
1 NOECHO
```

NOMSG

Disables messages output in the LOG-file.

NOMSG syntax

```
NOMSG
```

NOWARN

Disables warnings output in the LOG-file.

NOWARN syntax

```
1 NOWARN
```

PATH

Specifies the paths to the include files. The program first searches for the files in the current working directory and then in other locations in the sequence they are specified by the keyword.

PATH syntax

```
-- in any section of any program

PATH
   'path-to-include-files1' /
   'path-to-include-files2' /
   'path-to-include-files3' /
   ...
/

================================================================================

   path-to-include-files# - a path to a location where the program searches
                            for the include files and other files loaded into
                            a simulation.
```

SKIP-ENDSKIP

SKIP-ENDSKIP brackets comment out the keywords in between. Any keyword within the brackets do no affect the simulation.

SKIP-ENDSKIP syntax

```
SKIP

...

-- the keywords between SKIP-ENDSKIP are ignored

...

ENDSKIP
```

TERMINAL

Passes a command to the OS terminal (command line). The synonym to this keyword is CMDLINE.

TERMINAL syntax

```
-- in any section of the RUN-file

TERMINAL
   'command_line'  Wait_Flag  Rank  /

================================================================================

   command_line   - Command line to be passed to the terminal (Linux, Mac) or
                    the Windows command line;
      Wait_Flag   - if T (default value) the program waits until end of the
                    command_line execution. If F then the program continues
                    running not waiting for end of the command_line
                    execution;
            Rank  - rank of the process which executes the command line. If
                    Rank=-1 then all processes execute the command_line. The
                    default value is 0.
```

UPROG

The input data after that switch are read only by the U-program. Alternative positions of the switch are ALLPROG, HPROG and IPROG. By default the switch is in the ALLPROG position.

UPROG syntax

```
-- in any section of the RUN-file

UPROG
```

WARN

Enables warnings output in the LOG-file.

WARN syntax

```
1 WARN
```

1.7. Kernel keywords of the H-program

The keywords below are available with every EOS module of the H-program.

ACTION-ENDACTIO

Allows to specify some keywords executed only if some required conditions are reached.

ACTION-ENDACTIO syntax

```
-- in the SCHEDULE section

ACTION
   Name  Nrepeat  MinTimeInterval /
   'LogicalExpression1'  AndOr1   /
   'LogicalExpression2'  AndOr2   /
   'LogicalExpression3'  AndOr3   /
   ...                            /
/

-- Keywords between the brackets are executed
   only if the LogicalExpression becomes true.

-- TIME, TSTEP, DATES and similar keywords are
   prohibited between ACTION-ENDACTIO brakets.

ENDACTIO

================================================================================

               Name - Name of the action (8-byte character);
            Nrepeat - Number of action repeatitions (defalut 1);
     MinTimeInterval - Minimum time interval between two repeatitions
                       (default 0);
  LogicalExpression# - A logical expression;
              AndOr# - Logical AND or OR. These argumets can be used to use a
                       complicated expressions consisting of two or more
                       strings.
```

ACTNUM

Specifies the ACTNUM property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ACTNUM syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ACTNUM
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ACTNUM
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ACTNUM
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ADD

Adds value to a property in a box of cells.

ADD syntax

```
-- in all sections except RUNSPEC and POST

ADD
   mnemonic1  value1  imin1 imax1   jmin1 jmax1   kmin1 kmax1 /
   mnemonic2  value2  imin2 imax2   jmin2 jmax2   kmin2 kmax2 /
   mnemonic3  value3  imin3 imax3   jmin3 jmax3   kmin3 kmax3 /
   ...
/

================================================================================

   mnemonic#   - mnemonic of the property which is modified.
   value#      - value added to the property value in the current input box.
   imin#/imax# - the boundaries of the input box along i-indexation axis.
                 By default these values are equal to the arguments 1 and 2
                 of the keyword BOX.
   jmin#/jmax# - the boundaries of the input box along j-indexation axis.
                 By default these values are equal to the arguments 3 and 4
                 of the keyword BOX.
   kmin#/kmax# - the boundaries of the input box along k-indexation axis.
                 By default these values are equal to the arguments 5 and 6
                 of the keyword BOX.

================================================================================

   The keyword results in the following:

       mnemonic1:=mnemonic1+value1
                          in the box [imin1,imax1]*[jmin1,jmax1]*[kmin1,kmax1]
       mnemonic2:=mnemonic2+value2
                          in the box [imin2,imax2]*[jmin2,jmax2]*[kmin2,kmax2]
       mnemonic3:=mnemonic3+value3
                          in the box [imin3,imax3]*[jmin3,jmax3]*[kmin3,kmax3]
       ...
```

ADDNAM

Adds value to a property for all cells which character names (if assigned) belongs to character template.

ADDNAM syntax

```
-- in all sections except RUNSPEC and POST

ADDNAM
   mnemonic1  value1  template1 /
   mnemonic2  value2  template2 /
   mnemonic3  value3  template3 /
   ...
/

================================================================================

   mnemonic#   - mnemonic of the property which is modified;
   value#      - value added to the property;
   template#   - character name template.

================================================================================

   The keyword results in the following:

       mnemonic1:=mnemonic1+value1 for all cells which character name (if it
                                              is assigned) belong to template1;
       mnemonic2:=mnemonic2+value2 for all cells which character name (if it
                                              is assigned) belong to template2;
       mnemonic3:=mnemonic3+value3 for all cells which character name (if it
                                              is assigned) belong to template3.
       ...
```

ADDPHASE

Adds an additional component in the simulation. This component is not mixing with other components, forming a separate phase. For every PVTNUM region the PVT properties of this phase must be specified specified within PVT-ENDPVT brackets by the DENTAB, ENTHTAB and VISTAB keywords. This additional phase is considered as a liquid phase relative to the phases of EOS-module. The saturation functions depending on saturation of the phase must be specified for every SATNUM region within brackets SAT-ENDSAT by setting the second argument of the opening bracket SAT to the name of the phase.

ADDPHASE syntax

```
-- in RUNSPEC section

ADDPHASE
   phasename /

================================================================================

   phasename  -  name of the phase (component). Only first 4 characters are
                 significant.
```

ADDREG

Adds value to a property in a region.

ADDREG syntax

```
-- in all sections except RUNSPEC and POST

ADDREG
  mnemonic1  value1  region1  regionID1 /
  mnemonic2  value2  region2  regionID2 /
  mnemonic3  value3  region3  regionID3 /
  ...
/

================================================================================

     mnemonic# - mnemonic of the property which is modified;
     value#    - value assigned to the property in the region;
     region#   - mnemonic of the region in which the property is modified;
     regionID# - region number.

================================================================================

   The keyword results in the following:

       mnemonic1:=mnemonic1+value1  in the region region1=regionID1
       mnemonic2:=mnemonic2+value2  in the region region2=regionID2
       mnemonic3:=mnemonic3+value3  in the region region3=regionID3
       ...
```

ADDZCORN

Adds value to the corner point depths in the input box.

ADDZCORN syntax

```
-- within MAKE-ENDMAKE brackets

ADDZCORN
  value1  imin1 imax1 jmin1 jmax1 kmin1 kmax1 m1_1 m2_1 m3_1 m4_1 m5_1 m6_1 /
  value2  imin2 imax2 jmin2 jmax2 kmin2 kmax2 m1_2 m2_2 m3_2 m4_2 m5_2 m6_2 /
  value3  imin3 imax3 jmin3 jmax3 kmin3 kmax3 m1_3 m2_3 m3_3 m4_3 m5_3 m6_3 /
  ...
/

=============================================================================

   value#      - the value added to ZCORN array in the input box;
   imin#/imax# - the boundaries of the input box along i-indexation axis.
                 By default these values are equal to the arguments 1 and 2
                 of the keyword BOX.
   jmin#/jmax# - the boundaries of the input box along j-indexation axis.
                 By default these values are equal to the arguments 3 and 4
                 of the keyword BOX.
   kmin#/kmax# - the boundaries of the input box along k-indexation axis.
                 By default these values are equal to the arguments 5 and 6
                 of the keyword BOX.
   mi_#        - (i=1,...,6). The mode:
                 'I-' - the operation is also applied to the adjacent face of
                        the grid block connected to the considered block in
                        the input box in the negative direction of i-index
                        axis.
                 'I+' - the operation is also applied to the adjacent face of
                        the grid block connected to the considered block in
                        the input box in the positive direction of i-index
                        axis.
                 'J-' - the operation is also applied to the adjacent face of
                        the grid block connected to the considered block in
                        the input box in the negative direction of j-index
                        axis.
                 'J+' - the operation is also applied to the adjacent face of
                        the grid block connected to the considered block in
                        the input box in the positive direction of j-index
                        axis.
                 'K-' - the operation is also applied to the adjacent face of
                        the grid block connected to the considered block in
                        the input box in the negative direction of k-index
                        axis.
                 'K+' - the operation is also applied to the adjacent face of
                        the grid block connected to the considered block in
```

```
45                          the input box in the positive direction of k-index
46                          axis.
```

AQUCON

Creates non-neighbor connections between grid blocks and the numerical aquifer. The aquifer cells must be first defined by the AQUNUMER keyword. This keyword is a simlified version of the BOUNDARY keyword. An alternative option is also AQUCONN.

AQUCON syntax

```
-- within MAKE-ENDMAKE brackets

AQUCON
  aqname1  imin1 imax1  jmin1 jmax1  kmin1 kmax1  d1
                             tranmult1 tranopt1 mode1  /
  aqname2  imin2 imax2  jmin2 jmax2  kmin2 kmax2  d2
                             tranmult2 tranopt2 mode2  /
  aqname3  imin3 imax3  jmin3 jmax3  kmin3 kmax3  d3
                             tranmult3 tranopt3 mode3  /
  ...
/

================================================================================

   aqname#         - the numerical aquifer ID;
   imin#-imax#     - the boundaries of the input box along i-indexation axis.
                     By default these values are equal to '1' and the 2nd
                     argument of the keyword MAKE, respectively;
   jmin#-jmax#     - the boundaries of the input box along j-indexation axis.
                     By default these values are equal to '1' and the 3rd
                     argument of the keyword MAKE, respectively;
   kmin#-kmax#     - the boundaries of the input box along k-indexation axis.
                     By default these values are equal to '1' and the 4th
                     argument of the keyword MAKE, respectively;
   d#              - The direction tag defining for which boundaries
                      of the domain the aquifer connections are created.
                     It must be one of (I-,I+,J-,J+,K-,K+);
   tranmult#       - transmissibility multiplier for the connections between
                     grid blocks and the numerical aquifer;
   tranopt#        - can be used for switching between varios calculations of
                     the transmissibility between grid blocks and the numerical
                     aquifer (default 0);
   mode#=ACTBASED - the blocks are created only for those grid block in the
         or NO       current input box, which have no active blocks connected
                     in the direction defined by di_# tags (default option);
        =ALL       - the blocks are created for all blocks in the current
         or YES      input box in the direction defined by d# tag.
```

AQUCONN

Creates non-neighbor connections between grid blocks and the numerical aquifer. The aquifer cells must be first defined by the AQUNUMER keyword. This keyword is a simlified version of the BOUNDARY keyword. An alternative option is also AQUCON. Note that the default transmissibility calculation option for that keyword differs from that of AQUCON.

AQUCONN syntax

```
-- within MAKE-ENDMAKE brackets

AQUCONN
  aqname1  imin1 imax1  jmin1 jmax1  kmin1 kmax1  d1_1 d2_1 d3_1 d4_1 d5_1 d6_1
                            tranmult1 tranopt1 mode1  /
  aqname2  imin2 imax2  jmin2 jmax2  kmin2 kmax2  d1_2 d2_2 d3_2 d4_2 d5_2 d6_2
                            tranmult2 tranopt2 mode2  /
  aqname3  imin3 imax3  jmin3 jmax3  kmin3 kmax3  d1_3 d2_3 d3_3 d4_3 d5_3 d6_3
                            tranmult3 tranopt3 mode3  /
  ...
/

================================================================================

   aqname#         - the numerical aquifer ID;
   imin#-imax#     - the boundaries of the input box along i-indexation axis.
                     By default these values are equal to '1' and the 2nd
                     argument of the keyword MAKE, respectively;
   jmin#-jmax#     - the boundaries of the input box along j-indexation axis.
                     By default these values are equal to '1' and the 3rd
                     argument of the keyword MAKE, respectively;
   kmin#-kmax#     - the boundaries of the input box along k-indexation axis.
                     By default these values are equal to '1' and the 4th
                     argument of the keyword MAKE, respectively;
   di_#            - (i=1,...,6). The direction tags defining for which
                     boundaries of the domain the aquifer connections are
                     created. It must be one of (I-,I+,J-,J+,K-,K+);
   tranmult#       - transmissibility multiplier for the connections between
                     grid blocks and the numerical aquifer;
   tranopt#        - can be used for switching between varios calculations of
                     the transmissibility between grid blocks and the numerical
                     aquifer (default 1);
   mode#=ACTBASED - the blocks are created only for those grid block in the
         or NO       current input box, which have no active blocks connected
                     in the direction defined by di_# tags (default option);
        =ALL       - the blocks are created for all blocks in the current
```

```
37          or YES      input box in the direction defined by d# tag.
```

AQUNUMER

Specifies the numerical aquifer. The simulator does not associate the aqufier with the primary grid blocks of the geological body. Instead the simulator creates virtual grid blocks for the aquifer. The grid blocks for a particular aquifer ID should appear in the sequence of increasing distance to the simulated reservoir sector. The first grid block for a particular aquifer ID can then connected to the grid by using AQUCON, AQUCONN or BOUNDARY keyword. Note that the FLUXNUM quantity is set equal to the aquifer ID in all grid blocks associate with that aquifer.

AQUNUMER syntax

```
-- within MAKE-ENDMAKE brackets

AQUNUMER
  aqname1 len1 area1 poro1 perm1 depth1 hcnd1 typenum1 actnum1 /
  aqname2 len2 area2 poro2 perm2 depth2 hcnd2 typenum2 actnum2 /
  aqname3 len3 area3 poro3 perm3 depth3 hcnd3 typenum3 actnum3 /
  ...
/

================================================================================

   aqname#        - the numerical aquifer ID;
   len#           - length of the grid block of the aquifer;
   area#          - the cross-section area of the grid block;
   poro#          - porosity;
   perm#          - permeability;
   depth#         - depth of the grid block;
   hcnd#          - heat conductivity for non-isothermal simulations;
   typenum#       - the TYPENUM parameter;
   actnum#        - the ACTNUM parameter.
```

ARITH

Performs an arithmetic operation with arrays and scalar variable. This keyword should be regarded a more convinient analogue to OPERATE. The operations and functions available for use in the arithmetic expressions are listed in section 1.5. It is recommended to place any expression in single quotes.

ARITH syntax

```
-- in any section except RUNSPEC and POST

ARITH
   'expression1'  imin1 imax1  jmin1 jmax1  kmin1 kmax1  /
   'expression2'  imin2 imax2  jmin2 jmax2  kmin2 kmax2  /
   'expression3'  imin3 imax3  jmin3 jmax3  kmin3 kmax3  /
   ...                                                   /
/

================================================================================

   expression# - an arithmetic expression.
   imin#/imax# - the boundaries of the input box along i-indexation axis.
                 By default these values are equal to the arguments 1 and 2
                 of the keyword BOX.
   jmin#/jmax# - the boundaries of the input box along j-indexation axis.
                 By default these values are equal to the arguments 3 and 4
                 of the keyword BOX.
   kmin#/kmax# - the boundaries of the input box along k-indexation axis.
                 By default these values are equal to the arguments 5 and 6
                 of the keyword BOX.
```

ARITHCND

This keyword allows a more flexible arithmetic operations with arrays. The region of gridblocks which the operations are applied to are defined by a logical expression, i.e. the condition line.

ARITHCND syntax

```
-- in any section except RUNSPEC and POST

ARITHCND
   'expression1'  'condline1'  /
   'expression2'  'condline2'  /
   'expression3'  'condline3'  /
   ...                         /
/

================================================================================

   expression# - an arithmetic expression.
   condline#   - a condition line (i.e. expression). The condition is
                 is considered true if is is positive.
```

ARITHCRN

Performs arithmetic operations with the coner point depths.

ARITHCRN syntax

```
-- within MAKE-ENDMAKE brackets

ARITHCRN
   'expression1'  imin1 imax1  jmin1 jmax1  kmin1 kmax1  /
   'expression2'  imin2 imax2  jmin2 jmax2  kmin2 kmax2  /
   'expression3'  imin3 imax3  jmin3 jmax3  kmin3 kmax3  /
   ...                                                   /
/

================================================================================

   expression# - an arithmetic expression.
   imin#/imax# - the boundaries of the 'corner point depth' input box along
                 i-indexation axis. By default imin=1 and imax=2*nx, where nx
                 is the number of grid blocks in the X direction in the current
                 grid.
   jmin#/jmax# - the boundaries of the 'corner point depth' input box along
                 j-indexation axis. By default jmin=1 and jmax=2*ny, where ny
                 is the number of grid blocks in the Y direction in the current
                 grid.
   kmin#/kmax# - the boundaries of the 'corner point depth' input box along
                 k-indexation axis. By default kmin=1 and kmax=2*nz, where nz
                 is the number of grid blocks in the z direction in the current
                 grid.

The following variables can be used in the expressions:
       XCOORDG  - X in the grid coordinates
       YCOORDG  - Y in the grid coordinates
       XCOORD   - X in the map coordinates
       YCOORD   - Y in the map coordinates
       DEPTH    - corner point depth

```

ARITHMET

A synonym to the ARITH keyword.

ARITHMET syntax

```
-- in any section except RUNSPEC and POST

ARITHMET
   'expression1'  imin1 imax1  jmin1 jmax1  kmin1 kmax1  /
   'expression2'  imin2 imax2  jmin2 jmax2  kmin2 kmax2  /
   'expression3'  imin3 imax3  jmin3 jmax3  kmin3 kmax3  /
   ...                                                   /
/

================================================================================

   expression# - an arithmetic expression.
   imin#/imax# - the boundaries of the input box along i-indexation axis.
                 By default these values are equal to the arguments 1 and 2
                 of the keyword BOX.
   jmin#/jmax# - the boundaries of the input box along j-indexation axis.
                 By default these values are equal to the arguments 3 and 4
                 of the keyword BOX.
   kmin#/kmax# - the boundaries of the input box along k-indexation axis.
                 By default these values are equal to the arguments 5 and 6
                 of the keyword BOX.
```

ARITHNAM

Performs an arithmetic operation with arrays and scalar variable. This keyword should be regarded a more convinient analogue to OPERANAM. The operations and functions available for use in the arithmetic expressions are listed in section 1.5. It is recommended to place any expression in single quotes.

ARITHNAM syntax

```
-- in any section except RUNSPEC and POST

ARITHNAM
   'expression1'  template1  /
   'expression2'  template2  /
   'expression3'  template3  /
   ...                       /
/

================================================================================

   expression# - an arithmetic expression;
     template# - character name template.
```

ARITHPIL

Performs arithmetic operations with the coordinates of the pillar line points.

ARITHPIL syntax

```
-- within MAKE-ENDMAKE brackets

ARITHPIL
   'expression1'  imin1 imax1  jmin1 jmax1  kmin1 kmax1  /
   'expression2'  imin2 imax2  jmin2 jmax2  kmin2 kmax2  /
   'expression3'  imin3 imax3  jmin3 jmax3  kmin3 kmax3  /
   ...                                                   /
/

================================================================================

   expression# - an arithmetic expression.
   imin#/imax# - the boundaries of the 'pillar lines' input box along
                 i-indexation axis. By default imin=0 and imax=nx, where nx
                 is the number of grid blocks in the X direction in the current
                 grid.
   jmin#/jmax# - the boundaries of the 'pillar lines' input box along
                 j-indexation axis. By default jmin=0 and jmax=ny, where ny
                 is the number of grid blocks in the Y direction in the current
                 grid.
   kmin#/kmax# - the boundaries of the 'pillar lines' input box along the
                 pillar lines. By default kmin=1 and kmax=numpillr, where
                 numpillr is the number of point on the pillar lines. The
                 latter variable can be changed by the MAKE keyword.

The following variables can be used in the expressions:
       XCOORDG  - X in the grid coordinates
       YCOORDG  - Y in the grid coordinates
       XCOORD   - X in the map coordinates
       YCOORD   - Y in the map coordinates
       DEPTH    - depth of the pillar point
       AUXPIL1  - 1st auxiliary variable at pillars
       AUXPIL2  - 2nd auxiliary variable at pillars

       Note the AUXPIL1 and AUXPIL2 are set to 0 at every new
       occurance of ARITHPIL.
```

ARITHREG

Performs an arithmetic operation with arrays and scalar variable. This keyword should be regarded a more convinient analogue to OPERAREG. The operations and functions available for use in the arithmetic expressions are listed in section 1.5. It is recommended to place any expression in single quotes.

ARITHREG syntax

```
-- in any section except RUNSPEC and POST

ARITHREG
   'expression1'  region1  regionID1 /
   'expression2'  region2  regionID2 /
   'expression3'  region3  regionID3 /
   ...                               /
/

================================================================================

   expression# - an arithmetic expression;
     region#   - mnemonic of the region in which the property is modified;
     regionID# - region number.
```

AUTO

Affects the interpretation of CPVAR, OPERATE, OPERAREG,OPERANAM, EV, OPVAR keywords as well as other similar keywpords. If AUTO is placed ahead of these and similar keywords, then they are understood as the automatic keywords, i.e. executed after every time step.

AUTO syntax

```
-- in the INIT or SCHEDULE section of the H-program
-- within SCHED-ENDSCHED or OPTIM-ENDOPTIM bractkets of the U-program

AUTO
```

AUTOPART

Enables automatic grid decomposition by Metis when ENDMAKE bracket is encountered. The keyword is usually not necessary because this is the default option

AUTOPART syntax

```
-- within MAKE-ENDMAKE brackets

AUTOPART
```

AUXARRAY

Enables auxiliary arrays defined by user. For instance, the arrays can be used for temporary storage of data during complicated operations on arrays.

AUXARRAY syntax

```
-- in RUNSPEC section

AUXARRAY
  arrname1 units1 arrtype1 /
  arrname2 units2 arrtype2 /
  arrname3 units3 arrtype3 /
  ...
/

================================================================================

     arrname#    - name of the new array. The name must start with the '#' sign,
                   and its length must be 8 characters at maximum;
          units#      - the units of the quantity. This argument does not influence
                   any units conversion on both input and output of data. It
                   can be specified solely to improve the convenience of data
                                    usage;
     arrtype#    - type of array:
                   if CELL then the array is associated with the cells.
                      A value of the array corresponds to a cell (default);
                   if CONN then the array is associated with the connections.
                      A value of the array corresponds to a connection.
```

AUXVAR

Creates auxiliary scalar variable.

AUXVAR syntax

```
-- in RUNSPEC section

AUXVAR
  varname1  units1  value1 /
  varname2  units2  value2 /
  varname3  units3  value3 /
  ...
/

================================================================================

     varname#    - name of the auxiliary variable;
     units#      - units of the quantity (is used only for reporting data);
     value#      - initial value of the variable (default 0.0).
```

BLKROCK

Enables the specification of the rock compressibility (see the CBLKROCK keyword) and the reference pressure (see the PBLKROCK keyword) for every grid block. These data override those values defined within ROCK-ENDROCK brackets.

BLKROCK syntax

```
-- in RUNSPEC section

BLKROCK
```

BOUNDARY

Creates the grid blocks for imposing boundary conditions. Note when using the AQUIFER option, the numerical aquifer must be defined by the AQUNUMER keyword before using BOUNDARY.

BOUNDARY syntax

```
-- within MAKE-ENDMAKE brackets

BOUNDARY
  fluxnum1 imin1 imax1 jmin1 jmax1 kmin1 kmax1 d1_1 d2_1 d3_1 d4_1 d5_1 d6_1
           type_1 mode_1 Xcenter_1 Ycenter_1 Radius_1 typenum1 actnum1
           CreateType_1   nu_1  SrsGroup_1  Prefix_1
           aqname1 tranmult1 tranopt1 /
  fluxnum2 imin2 imax2 jmin2 jmax2 kmin2 kmax2 d1_2 d2_2 d3_2 d4_2 d5_2 d6_2
           type_2 mode_2 Xcenter_2 Ycenter_2 Radius_2 typenum2 actnum2
           CreateType_2   nu_2  SrsGroup_2  Prefix_2
           aqname2 tranmult2 tranopt2 /
  fluxnum3 imin3 imax3 jmin3 jmax3 kmin3 kmax3 d1_3 d2_3 d3_3 d4_3 d5_3 d6_3
           type_3 mode_3 Xcenter_3 Ycenter_3 Radius_3 typenum3 actnum3
           CreateType_3   nu_3  SrsGroup_3  Prefix_3
           aqname3 tranmult3 tranopt3 /
  ...
/

================================================================================

   fluxnum#       - FLUXNUM region number assigned to created grid blocks;
   imin#-imax#    - the boundaries of the input box along i-indexation axis.
                    By default these values are equal to '1' and the 2nd
                    argument of the keyword MAKE, respectively;
   jmin#-jmax#    - the boundaries of the input box along j-indexation axis.
                    By default these values are equal to '1' and the 3rd
                    argument of the keyword MAKE, respectively;
   kmin#-kmax#    - the boundaries of the input box along k-indexation axis.
                    By default these values are equal to '1' and the 4th
                    argument of the keyword MAKE, respectively;
   di_#           - (i=1,...,6). The direction tags defining for which
                    boundaries of the domain the cells are created. It must be
                    one of (I-,I+,J-,J+,K-,K+);
   type#=SAMESIZE - the created blocks have the same CELLVOL property as the
                    blocks to which they are connected (default option),
         =INFTHIN  - the created block are "infinitely" thin;
         =CYLINTYP - can be useful for a more detailed modeling of an aquifer.
         =AREAFLUX - can be specified to modeling a FLUX boundary conditions.
                     Instead of the boundar grid blocks the simulator creates
                     a set of point sources. Each grid block at the boundary
```

```
                  will be allocated with a point source.
      =AQUIFER  - creates connections to the numerical aquifer.
mode#=ACTBASED - the blocks are created only for those grid block in the
                  current input box, which have no active blocks connected
                  in the direction defined by di_# tags (default option);
      =ALL      - the blocks are created for all blocks in the current
                  input box in the direction defined by di_# tags;
XCenter_#/
 YCenter_#     - coordinates of the aquifer center in the case of the
                  CYLINTYP condition;
Radius_#       - Aquifer radius in the case of the CYLINTYP condition;
typenum#       - the TYPENUM property assigned to created grid blocks;
actnum#        - the ACTNUM property assigned to created grid blocks.
CreateType_#
  =COLUMN      - In the case of I+, I-, J+ and J- directions, the boundary
                  condition is applied only if the active grid blocks do
                  not exist in the whole column of grid blocks in the
                  corresponding direction (default option);
  =CELL        - In the case of I+, I-, J+ and J- directions, the boundary
                  condition is applied only if the specific grid block
                  at the boundary does not have an active adjeacent grid
                  block in the corresponding direction;
  =NCOLUMN     - In the case of I+, I-, J+ and J- directions, the boundary
                  condition is applied only if the specific grid block
                  at the boundary does not have an active adjeacent grid
                  block in the corresponding direction whereas some of the
                  grid blocks in the adjacent column are active.
nui_#          - not used at present;
SrcGroup_#     - An integer ID of the group to which all created by
                  AREAFLUX point sources belong;
Prefix_#       - A character prefix in the point sources names Default 'S'.
aqname#        - the numerical aquifer ID;
tranmult#      - transmissibility multiplier for the connections between
                  grid blocks and the numerical aquifer;
tranopt#       - can be used for switching between varios calculations of
                  the transmissibility between grid blocks and the numerical
                  aquifer (default 1).

```

BOX

Defines the current input box.

BOX syntax

```
-- in all sections except RUNSPEC and POST

BOX
   imin imax  jmin jmax  kmin kmax /

================================================================================

   imin/imax - the boundaries of the input box along i-indexation axis.
               By default imin=1 and imax=ni, where ni is the 2nd argument
               of the keyword MAKE.
   jmin/jmax - the boundaries of the input box along j-indexation axis.
               By default jmin=1 and jmax=nj, where nj is the 3rd argument
               of the keyword MAKE.
   kmin/kmax - the boundaries of the input box along k-indexation axis.
               By default kmin=1 and kmax=nk, where nk is the 4th argument
               of the keyword MAKE.
```

BRITTAB

Defines parameters of the dynamic fractures model. See also the DYNFRACK keyword. The data should be provided for every rock unit in the ROCK-ENDROCK brackets. If the data for a rock unit are not specified then the DYNFRACK option is not applied to that unit. A synonym to this keyword is DYNFRTAB.

BRITTAB syntax

```
-- within ROCK-ENDROCK brackets

BRITTAB
    lambda1    F1    ximax1  /
    lambda2    F2    ximax2  /
    ...                      /
    lambdaN    FN    ximaxN  /
/

================================================================================

    lambda$    - The lambda paramter (i.e. pore-fluid factor). It must increase
                 down the table;
         F$    - The value of the fractures "productivity" for the corresponding
                 lambda;
      ximax$   - The maixmum fractures aperture.
```

BRITTLE

Enables the dynamic fractures option. A synonym to this keyword is DYNFRACK.

BRITTLE syntax

```
-- in RUNSPEC section

BRITTLE

```

CAPPRES

Initiates a simulation subject to modeling capillary pressure effects.

CAPPRES syntax

```
-- in section RUNSPEC

CAPPRES
```

CARFIN

Defines a local grid refinement. After this keyword the refined grid is set as active by the program.

CARFIN syntax

```
-- within MAKE-ENDMAKE brackets

CARFIN
  name  imin imax  jmin jmax  kmin kmax  nx ny nz  not_used  parent /

================================================================================

     name      - name of the refined grid;
     imin/imax - the boundaries of the refined grid along i-index direction
                 in the parent grid;
     jmin/jmax - the boundaries of the refined grid along j-index direction
                 in the parent grid;
     kmin/kmax - the boundaries of the refined grid along j-index direction
                 in the parent grid;
     nx        - the number of grid blocks in the refined grid along i-index
                 direction;
     ny        - the number of grid blocks in the refined grid along j-index
                 direction;
     nz        - the number of grid blocks in the refined grid along k-index
                 direction;
     not_used  - this data item is not used by the simulator;
     parent    - the parent grid name.
```

CBLKROCK

Specifies the CBLKROCK property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

CBLKROCK syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

CBLKROCK
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

CBLKROCK
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

CBLKROCK
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

CFLCTRL

Enables the time step selection with the Courant-Friedrichs-Lewy condition. The next time step will be reduced if the CFL number at the current time is greater then the maximum allowed CFL number. The time step will be reduced in order to make the CFL number lower than the threshold value. This keyword must be first placed in the RUNSPEC section. See also the DVARCTRL keyword.

CFLCTRL syntax

```
-- in the RUNSPEC or SCHEDULE section

CFLCTRL
   OnOff   MaxCFL OnOff2 /

================================================================================

     OnOff   - If ON then the time step control is enabled.
     MaxCFL  - The maximum allowed Courant-Friedrichs-Lewy number.
     OnOff2  - If ON then the CFL limit is soften for small gradients of
               concentrations and saturation (default OFF).
```

CIRCLE

Creates non-neighboring connections in radial grids to complete the circle.

CIRCLE syntax

```
-- within MAKE-ENDMAKE brackets

CIRCLE
```

COMPDAT

Defines the well completion data. Note, that if this keyword is positioned in SCHEDULE section then the GETWELLS keyword must be specified within MAKE-ENDMAKE brackets.

COMPDAT syntax

```
-- within MAKE-ENDMAKE brackets or in SCHEDULE section

COMPDAT
 name1 iloc1 jloc1 kmin1 kmax1 mode1 satnum1 tran1 d1 kh1 skin1 nu1 dir1 r0_1
                                                                  roughness1 /
 name2 iloc2 jloc2 kmin2 kmax2 mode2 satnum2 tran2 d2 kh2 skin2 nu2 dir2 r0_2
                                                                  roughness2 /
 name3 iloc3 jloc3 kmin3 kmax3 mode3 satnum3 tran3 d3 kh3 skin3 nu3 dir3 r0_3
                                                                  roughness3 /
 ...
/

================================================================================

   name#       - well name;
   nu#         - a parameter not used at present;
   iloc#/jloc# - i-index and j-index of the grid block where the well is
                 completed;
   kmin#/kmax# - k-index range of grid blocks where the well is completed;
   mode#       - if OPEN (default) the completion is opened for the fluid
                 transport; if SHUT the completion is closed off;
   satnum#     - saturation functions region (SATNUM) used for the calculation
                 of the fluxes through the completion;
   tran#       - this field is for explicit specification of the
                 transmissibility;
   d#          - wellbore diameter at the connection;
   kh#         - effective 'permeability*thickness' value at the connection;
   skin#       - skin factor;
   nu#         - not used at present;
   dir#        - direction in which the well penetrates the grid block. The
                 available values are: 'X' - the well penetrates the grid block
                 in the direction of X axis, 'Y' - the well penetrates the
                 grid block in the direction of Y axis, 'Z' - the well
                 penetrates the grid block in the direction of Z axis (default);
   r0_#        - drainage radius;
   roughness#  - relative roughness of pipejunction walls in the Multisegment
                 well mode (default 0.0).
```

COMPDATM

Defines the completion intervals for well trajectories (see the WELTRACK keyword).

COMPDATM syntax

```
-- within MAKE-ENDMAKE brackets

COMPDATM
 name1 branch1 mdu1 mdl1 meth1 mode1 satnum1 tran1 d1 kh1 skin1 nu1 dir1 r0_1
                                                                  roughness1 /
 name2 branch2 mdu2 mdl2 meth2 mode2 satnum2 tran2 d2 kh2 skin2 nu2 dir2 r0_2
                                                                  roughness2 /
 name3 branch3 mdu3 mdl3 meth3 mode3 satnum3 tran3 d3 kh3 skin3 nu3 dir3 r0_3
                                                                  roughness3 /
 ...
/

================================================================================

   name#       - well name;
   branch#     - well branch ID (default 0);
   mdu#/mdl#   - upper/lower limits of the completion interval;
   meth#       - Possible values
                   MD means that mdu#/mdl# are the depths measured along
                   the well bore. These depths are specified in the 4th column
                   of WELTRACK (default option);
                   TVD means that mdu#/mdl# are the true vertical depths. These
                   depths are specified in the 3rd column of WELTRACK;
   nu#         - a parameter not used at present;
   iloc#/jloc# - i-index and j-index of the grid block where the well is
                 completed;
   kmin#/kmax# - k-index range of grid blocks where the well is completed;
   mode#       - if OPEN (default) the completion is opened for the fluid
                 transport; if SHUT the completion is closed off;
   satnum#     - saturation functions region (SATNUM) used for the calculation
                 of the fluxes through the completion;
   tran#       - this field is for explicit specification of the
                 transmissibility;
   d#          - wellbore diameter at the connection;
   kh#         - effective 'permeability*thickness' value at the connection;
   skin#       - skin factor;
   nu#         - not used at present;
   roughness#  - relative roughness of pipejunction walls in the Multisegment
                 well mode (default 0.0).
```

CONTINUE

Requests the simulation restart from specified time in SCHEDULE. The restart option must be enabled by the RESTART keyword in the RUNSPEC section.

CONTINUE syntax

```
-- in the SCHEDULE section

CONTINUE
   timestep /

================================================================================

      timestep   - next timestep after restarting.
```

CONVERT

Initiates the conversion of output files to the ParaViewcompatible format.

CONVERT syntax

```
-- in section POST

CONVERT
```

COORD

Defines the pillar coordinates for corner-point grid. Every pillar - a coordinate line - is defined by two points. The keyword must be followed by a table which rows define the coordinates of two points for every pillar. The input is affected by the BOX keyword. This keyword should be used if the first argument of the MAKE is CART, RADIAL or CORNER.

COORD syntax

```
-- within brackets MAKE-ENDMAKE

COORD
   xa1 ya1 za1   xb1 yb1 zb1   ...
   xa2 ya2 za2   xb2 yb2 zb2   ...
   xa3 ya3 za3   xb3 yb3 zb3   ...
...
   xaN yaN zaN   xbN ybN zbN   ...
/

================================================================================

    xa#-ya#-za# and  - coordinates of different points on a pillar.
    xb#-yb#-zb# and    The total number of points on a pillar can be changed
    ...                by the 6th argument of MAKE. By default, it is 2.
                   N  - the total number of pillars in the current input box.
                        N=(imax-imin+2)*(jmax-jmin+2). The i-index is cycling
                        the fastest following by the j-index.
```

COPY

Copies the value from one property to another property in the grid box.

COPY syntax

```
-- in all sections except RUNSPEC and POST

COPY
   mnemfrom1  mnemto1   imin1 imax1   jmin1 jmax1   kmin1 kmax1 /
   mnemfrom2  mnemto2   imin2 imax2   jmin2 jmax2   kmin2 kmax2 /
   mnemfrom3  mnemto3   imin3 imax3   jmin3 jmax3   kmin3 kmax3 /
   ...
/

================================================================================

   mnemfrom#   - mnemonic of the property from which the value is copied.
   mnemto#     - mnemonic of the property to which the value is copied.
   imin#/imax# - the boundaries of the input box along i-indexation axis.
                 By default these values are equal to the arguments 1 and 2
                 of the keyword BOX.
   jmin#/jmax# - the boundaries of the input box along j-indexation axis.
                 By default these values are equal to the arguments 3 and 4
                 of the keyword BOX.
   kmin#/kmax# - the boundaries of the input box along k-indexation axis.
                 By default these values are equal to the arguments 5 and 6
                 of the keyword BOX.

================================================================================

   The keyword results in the following:

       mnemto1:=mnemfrom1  in the box [imin1,imax1]*[jmin1,jmax1]*[kmin1,kmax1]
       mnemto2:=mnemfrom2  in the box [imin2,imax2]*[jmin2,jmax2]*[kmin2,kmax2]
       mnemto3:=mnemfrom3  in the box [imin3,imax3]*[jmin3,jmax3]*[kmin3,kmax3]
       ...
```

COPYBOX

Copies a quantity from one input box to another input box. The keyword can be used only in a sequential (single-process) simulations.

COPYBOX syntax

```
-- in all sections except RUNSPEC and POST

COPYBOX
   mnemonic1    imin1_fr imax1_fr   jmin1_fr jmax1_fr   kmin1_fr kmax1_fr
                imin1_to imax1_to   jmin1_to jmax1_to   kmin1_to kmax1_to /
   mnemonic2    imin2_fr imax2_fr   jmin2_fr jmax2_fr   kmin2_fr kmax2_fr
                imin2_to imax2_to   jmin2_to jmax2_to   kmin2_to kmax2_to /
   mnemonic3    imin3_fr imax3_fr   jmin3_fr jmax3_fr   kmin3_fr kmax3_fr
                imin3_to imax3_to   jmin3_to jmax3_to   kmin3_to kmax3_to /
   ...
/

================================================================================

   mnemonic#         - mnemonic of the property that is copied.
   imin#_fr/imax#_fr - the boundaries of the input box along i-indexation axis,
                       from which the property is copied.
   jmin#_fr/jmax#_fr - the boundaries of the input box along j-indexation axis,
                       from which the property is copied.
   kmin#_fr/kmax#_fr - the boundaries of the input box along k-indexation axis,
                       from which the property is copied.
   imin#_to/imax#_to - the boundaries of the input box along i-indexation axis,
                       to which the property is copied.
   jmin#_to/jmax#_to - the boundaries of the input box along j-indexation axis,
                       to which the property is copied.
   kmin#_to/kmax#_to - the boundaries of the input box along k-indexation axis,
                       to which the property is copied.

================================================================================

   Note: The following conditions must be satidfied
             imax#_fr-imin#_fr=imax#_to-imin#_to
             jmax#_fr-jmin#_fr=jmax#_to-jmin#_to
             kmax#_fr-kmin#_fr=kmax#_to-kmin#_to

   Note: The keyword can be used only in a single-process run.
```

COPYNAM

Copies the value from one property to another property for all cells which character names (if assigned) belongs to character template.

COPYNAM syntax

```
-- in all sections except RUNSPEC and POST

COPYNAM
   mnemfrom1  mnemto1  template1 /
   mnemfrom2  mnemto2  template2 /
   mnemfrom3  mnemto3  template3 /
   ...
/

================================================================================

     mnemfrom# - mnemonic of the property from which the value is copied;
     mnemto#   - mnemonic of the property to which the value is copied;
     template# - character name template.

================================================================================

   The keyword results in the following:

       mnemto1:=mnemfrom1 for all cells which character name (if it
                                              is assigned) belong to template1;
       mnemto2:=mnemfrom2 for all cells which character name (if it
                                              is assigned) belong to template2;
       mnemto3:=mnemfrom3 for all cells which character name (if it
                                              is assigned) belong to template3.
       ...
```

COPYREG

Copies the value from one property to another property in the region.

COPYREG syntax

```
-- in all sections except RUNSPEC and POST

COPYREG
  mnemfrom1  mnemto1  region1  regionID1 /
  mnemfrom2  mnemto2  region2  regionID2 /
  mnemfrom3  mnemto3  region3  regionID3 /
  ...
/

================================================================================

     mnemfrom# - mnemonic of the property from which the value is copied;
     mnemto#   - mnemonic of the property to which the value is copied;
     region#   - mnemonic of the region in which the property is modified;
     regionID# - region number.

================================================================================

   The keyword results in the following:

       mnemto1:=mnemfrom1  in the region region1=regionID1
       mnemto2:=mnemfrom2  in the region region2=regionID2
       mnemto3:=mnemfrom3  in the region region3=regionID3
       ...
```

COPY2BND

Copies a quantity to the boundary grid blocks from associated parent grid blocks (see the BOUNDARY keyword).

COPY2BND syntax

```
-- in any section except RUNSPEC and POST

COPY2BND
   property1  region1  regionID1  /
   property2  region2  regionID2  /
   property3  region3  regionID3  /
   ...                            /
/

================================================================================

   property#   - the mnemonic of a property to be copied.
   region#     - the menmonic of a region which the procdure is applied to.
   regionID#   - ID of the region.
```

COUPLED

Enables the simulator coupling with third-party software. The associated output is defined by the keywords RPTGRIDC and RPTSUMC.

COUPLED syntax

```
-- in RUNSPEC or SCHEDULE section

COUPLED
   on-off two-way read-incl waitfile  gridfile  sumfile_out  sumfile_in
   include_in  sleeptime  /

================================================================================

    on-off      - ON enables the coupling option, OFF (default) disables the
                  option;
    two-way     - if YES then the two-way coupling is applied (the
                  sumfile_in file will be loaded after every timestep). If NO
                  (default option) then one-way coupling is applied;
    read-incl   - if YES then additional input keywords are loaded into
                  simulation from a user-provided include file. The syntax
                  of this include file is the same as that of the RUN-file.
                  The TSTEP, TIME, DATES and other keywords related to time
                  are not allowed in the include file. If NO (default option)
                  then include file is not read;
    waitfile    - name of the file for run control. If the first item in the
                  file is 0 the hydrodynamic simulation is continued (default
                  WAIT.TXT);
    gridfile    - name of the file for output from the GRID section (default
                  COUPLED.GRID.SUM);
    sumfile_out - name of the file for output from the SCHEDULE section
                  after every time step (default COUPLED.OUT.SUM);
    sumfile_in  - name of the file for input in the SCHEDULE section
                  after every time step (default COUPLED.IN.SUM);
    include_in  - the name of the include file (see option read-incl).
                  Default is COUPLED.INC;
    sleeptime   - time (milliseconds) between the waitfile checks
                  (default 1000).
```

CPVAR

Copies a particular quantity of to the auxiliary variable defined by AUXVAR keyword.

CPVAR syntax

```
-- in any section except RUNSPEC and POST

CPVAR
  varname1 DataType1 PropName1 Name1_1 Name2_1  ID1_1 ID2_1 ID3_1 /
  varname2 DataType2 PropName2 Name1_2 Name2_2  ID1_2 ID2_2 ID3_2 /
  varname3 DataType3 PropName3 Name1_3 Name2_3  ID1_3 ID2_3 ID3_3 /
   ...
/

================================================================================

   varname#    - auxiliary variable to which the property is copied.
   DataType#   - mnemonic of the data type of the property:
                 CELL - a cell quantity (the cell is specified by ID1_#=
                        CellName)).
                 BLOCK - a grid block quantity (the block is specified by
                 ID1_#=i, ID2_#=j, ID3_#=k, Name1_#=GridName, Name2_#=
                 ReservoirName).
                 SOURCE - a point source quantity (the source is specified by
                 Name1_#=SourceName).
                 WELL - a wellbore quantity (the well is specified by
                 Name1_#=SourceName).
                 FIELD - a field quantity (specific names are not needed).
                 FPCE, FIPCELL - an average of cumulative quantity for a
                 FIPNUM region ID1.
                 FPCO, FIPCONN - an average of cumulative quantity for a
                 boundary between two adjacenet FIPNUM regions ID1 and ID2.
   PropName#   - name of the property (e.g. PRES if DataType#=BLOCK).
   Name$_#     - a character name characterizing particular element of the
                 reservoir model.
   ID$_#       - an integer name characterizing particular element of the
                 reservoir model.

```

CPVARAUT

Automatically copies a particular quantity of to the auxiliary variable defined by AUXVAR keyword at the end of every time step.

CPVARAUT syntax

```
-- in INIT or SCHEDULE section

CPVARAUT
  varname1 DataType1 PropName1 Name1_1 Name2_1 ID1_1 ID2_1 ID3_1 /
  varname2 DataType2 PropName2 Name1_2 Name2_2 ID1_2 ID2_2 ID3_2 /
  varname3 DataType3 PropName3 Name1_3 Name2_3 ID1_3 ID2_3 ID3_3 /
   ...
/

================================================================================

   varname#    - auxiliary variable to which the property is copied.
   DataType#   - mnemonic of the data type of the property:
                 CELL - a cell quantity (the cell is specified by ID1_#=
                        CellName)).
                 BLOCK - a grid block quantity (the block is specified by
                 ID1_#=i, ID2_#=j, ID3_#=k, Name1_#=GridName, Name2_#=
                 ReservoirName).
                 SOURCE - a point source quantity (the source is specified by
                 Name1_#=SourceName).
                 WELL - a wellbore quantity (the well is specified by
                 Name1_#=SourceName).
                 FIELD - a field quantity (specific names are not needed).
   PropName#   - name of the property (e.g. PRES if DataType#=BLOCK).
   Name$_#     - a character name characterizing particular element of the
                 reservoir model.
   ID$_#       - an integer name characterizing particular element of the
                 reservoir model.

```

DATES

Defines the report dates and advances the simulation to further report dates. When the simulator encounters this keyword it proceeds the simulation further in time and produces the required outputs. See also the START keyword.

DATES syntax

```
-- in SCHEDULE section

DATES
   day1  month1  year1  time1  sec1 /
   day2  month2  year2  time2  sec2 /
   day3  month3  year3  time3  sec3 /
...
/

================================================================================

      day#   - calendar day   (1,2,3,...,31)
      month# - calendar month (JAN,FEB,MAR,APR,MAY,JUN,JUL,AUG,SEP,OCT,NOV,DEC)
      year#  - year (e.g., 2012)
      time#  - time of the day in HH:MM format (default 12:00)
      sec#   - seconds of float type (default 0.0)

      The dates must be in calendar order.
```

DENTAB

For additional phase loads the density on pressure and temperature. This keyword must be used within PVT-ENDPVT brackets. See also the AD-DPHASE keyword.

DENTAB syntax

```
-- within EOS-ENDEOS, or PVT-ENDPVT brackets, or in the PROPS section

DENTAB
        temp1  temp2  temp3  ...  /
 pres1  den11  den12  den13  ...  /
 pres2  den21  den22  den23  ...  /
 pres3  den31  den32  den33  ...  /
 ...    ...    ...    ...    ...  /
/

================================================================================

     temp#  -  temperature (degree Kelvin);
     pres$  -  pressure;
     den$#  -  density for a given pressure (pres$) and temperature (temp#).
```

DIFFCHEM

Enables modelling molecular diffusion by using the gradient of the chemical potentials. The diffusivity multipliers for every grid block can be specified by DIFFMX, DIFFMY, DIFFMZ and similar keywords. The coefficients of molecular diffusion can be specified using the DIFFAGAS and DIFFAOIL keyword.

DIFFCHEM syntax

```
-- in section RUNSPEC

DIFFCHEM
```

DIFFMI

Specifies the DIFFMI property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

DIFFMI syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

DIFFMI
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

DIFFMI
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

DIFFMI
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

DIFFMJ

Specifies the DIFFMJ property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

DIFFMJ syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

DIFFMJ
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

DIFFMJ
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

DIFFMJ
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

DIFFMK

Specifies the DIFFMK property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

DIFFMK syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

DIFFMK
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

DIFFMK
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

DIFFMK
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

DIFFMMF

Specifies the DIFFMMF property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

DIFFMMF syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

DIFFMMF
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

DIFFMMF
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

DIFFMMF
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

DIFFMX

Specifies the DIFFMX property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

DIFFMX syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

DIFFMX
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

DIFFMX
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

DIFFMX
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

DIFFMY

Specifies the DIFFMY property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

DIFFMY syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

DIFFMY
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

DIFFMY
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

DIFFMY
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

DIFFMZ

Specifies the DIFFMZ property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

DIFFMZ syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

DIFFMZ
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

DIFFMZ
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

DIFFMZ
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

DPNUM

Allows fine tuning of the connections within the matrix media in double permeability simulations. See also the DUALPERM keyword.

DPNUM syntax

```
-- within the MAKE-ENDMAKE brackets

DPNUM
   dpnum1  dpnum2  dpnum3  ... dpnum(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
               If dpnum=1 for both grid blocks corresponding to an interface
               between the blocks then connections both between fracture and
               matrix grid blocks are created.
               If dpnum=0 for any grid block corresponding to an interface
               between the blocks then connection only between fracture grid
               blocks are created.
               By default dpnum=0 if the DUALPERM keyword is not specified,
               and dpnum=1 it the DUALPERM keyword is specified in the
               SPECRUN section.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX. The data corresponding only to the matrix medium is
               relevant.
```

DRV

Defines the grid block extensions along axis R in Radial grids. This keyword should be specified immediately after keyword MAKE or keyword RTZBOUND.

DRV syntax

```
-- within MAKE-ENDMAKE brackets

DRV
  dr(1) dr(2) dr(3) ... dr(nr) /

================================================================================

    dr(#) - grid blocks extensions along axis R.
    nr    - number of grid block along axis R. nr is the 2nd argument of the
            keyword MAKE.
```

DTHETAV

Defines the grid block extensions along axis Theta in Radial grids. This keyword should be specified immediately after keyword MAKE or keyword RTZBOUND.

DTHETAV syntax

```
-- within MAKE-ENDMAKE brackets

DTHETAV
  dt(1) dt(2) dt(3) ... dt(nt) /

================================================================================

    dt(#) - grid blocks extensions along Theata axis in radial geometry [rad].
    nt    - number of grid block along Theta axis. nt is the 3rd argument of the
            keyword MAKE.
```

DUALPERM

Activates double permeability option. The DUALPORO keyword must be specified before DUALPERM keyword.

DUALPERM syntax

```
-- in section RUNSPEC

DUALPERM
```

DUALPORO

Activates double porosity option.

DUALPORO syntax

```
-- in section RUNSPEC

DUALPORO
```

DUCTILE

Enables the option for modelling plastic rock behaviour at elevated temperatures.

DUCTILE syntax

```
-- in RUNSPEC section

DUCTILE
```

DUCTTAB

Defines parameters of the ductile creep of rocks at elevated temperatures. See also the DUCTILE keyword. The data should be provided for every rock unit in the ROCK-ENDROCK brackets. If the data for a rock unit are not specified then the DUCTILE option is not applied to that unit.

DUCTTAB syntax

```
-- within ROCK-ENDROCK brackets

DUCTTAB
   Units     Temp1     Temp2     ...   TempN    /
             eta1      eta2      ...   etaN     /
   lambda1  -log(D)11 -log(D)12       -log(D)1N /
   lambda2  -log(D)21 -log(D)22       -log(D)2N /
   ...                                          /
   lambdaK  -log(D)K1 -log(D)K2       -log(D)KN /
/

================================================================================

    Units      - Temperature units. Possible values are "K" for degree Kelvin
                 (default value) and "C" for degree Celsius;
    Temp#      - Temperature values (must increase in the row);
    eta#       - The eta parameter values (must not decrease in the row and
                 include the value eta=0 and eta=1);
    lambda$    - The lambda paramter (i.e. pore-fluid factor). Must decrease
                 down the table and the highest value (i.e. lambda1) must be
                 equal to 1;
    -log(D)$#  - The logarithm of the parameter D (transmissibility
                 multiplier). Must not decrease in every row and every column;
                 -log(D)$# must be equal to 0 at lambda=1 and eta=0.
```

DVARCTRL

Enables the time step selection by using target variables changes. If the option is enabled then at the beginning of every timestep the simulator estimates variables changes over the next time step and chooses the next timestep length such that the variables changes are equal to the target quantities. See also the CFLCTRL keyword.

DVARCTRL syntax

```
-- in the RUNSPEC or SCHEDULE section

DVARCTRL
   OnOff   Scale /

================================================================================

     OnOff   - ON or OFF (default OFF). If ON then the time step control is
               enabled.
     Scale   - Multiplier for the target variable changes (default 1.0).
```

DXV

Defines the grid block extensions along axis X in Cartesian grids. This keyword should be specified immediately after keyword MAKE or keyword XYZBOUND

DXV syntax

```
-- within MAKE-ENDMAKE brackets

DXV
  dx(1) dx(2) dx(3) ... dx(nx) /

================================================================================

    dx(#) - grid blocks extensions along axis X.
    nx    - number of grid block along axis X. nx is the 2nd argument of the
            keyword MAKE.
```

DYNFRACK

Enables the dynamic fractures option. A synonym to this keyword is BRITTLE. This keyword enables the "nonequilibrium" model. The equilibrium model is switched on with the DYNFREQL keyword.

DYNFRACK syntax

```
-- in the RUNSPEC section

DYNFRACK
```

DYNFREQL

Enables the "equilibrium" dynamic fractures option. The "nonequilibrium" model is enabled with the DYNFRACK keyword.

DYNFREQL syntax

```
-- in the RUNSPEC section

DYNFREQL
```

DYNFROFF

Enables the dynamic fractures permeability changes. See also the DYNFRACK and DYNFRON. By default the changes are enabled if DYNFRACK was placed in the RUNSPEC section.

DYNFROFF syntax

```
-- in the SCHEDULE section

DYNFROFF
```

DYNFRON

Enables the dynamic fractures permeability changes. See also the DYNFRACK and DYNFROFF. By default the changes are enabled if DYNFRACK was placed in the RUNSPEC section.

DYNFRON syntax

```
-- in the SCHEDULE section

DYNFRON
```

DYNFRTAB

Defines parameters of the dynamic fractures model. See also the DYNFRACK keyword. The data should be provided for every rock unit in the ROCK-ENDROCK brackets. If the data for a rock unit are not specified then the DYNFRACK option is not applied to that unit. A synonym to this keyword is BRITTAB.

DYNFRTAB syntax

```
-- within ROCK-ENDROCK brackets

DYNFRTAB
    lambda1    F1    ximax1  /
    lambda2    F2    ximax2  /
    ...                      /
    lambdaN    FN    ximaxN  /
/

================================================================================

    lambda$    - The lambda paramter (i.e. pore-fluid factor). It must increase
                 down the table;
         F$    - The value of the fractures "productivity" for the corresponding
                 lambda;
      ximax$   - The maixmum fractures aperture.
```

DYV

Defines the grid block extensions along axis Y in Cartesian grids. This keyword should be specified immediately after keyword MAKE or keyword XYZBOUND

DYV syntax

```
-- within MAKE-ENDMAKE brackets

DYV
  dy(1) dy(2) dy(3) ... dy(ny) /

================================================================================

    dy(#) - grid blocks extensions along axis Y.
    ny    - number of grid block along axis Y. nx is the 3rd argument of the
            keyword MAKE.
```

DZMTRX

Specifies default vertical dimension of matrix blocks for gravity drainage calculations (see also GRAVDR and DUALPORO). This default value is used if vertical dimension is not defined using DZMTRXV keyword.

DZMTRX syntax

```
-- in the GRID section

DZMTRX
   dzmtrx /

================================================================================

   dzmtrx    -  the vertical dimension (height) of matrix blocks in gravity
                drainage calculation. This is a default value for the height
                that is used if dzmtrx cannot be calculated using other input
                data. By default, dzmtrx is not defined.
```

DZMTRXV

Specifies the DZMTRXV property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

DZMTRXV syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

DZMTRXV
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

DZMTRXV
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

DZMTRXV
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

DZV

Defines the grid block extensions along axis Z in Cartesian and Radial grids. This keyword should be specified immediately after keyword MAKE or keyword XYZBOUND (RTZBOUND)

DZV syntax

```
-- within MAKE-ENDMAKE brackets

DZV
  dz(1) dz(2) dz(3) ... dz(nz) /

================================================================================

   dz(#) - grid blocks extensions along axis Z.
   nz    - number of grid block along axis Z. nz is the 4th argument of the
           keyword MAKE.
```

DZWELL

Defines the parameters of straight lines used for wells visualization. This keywords affects only the output in the file .WELL.SUM.

DZWELL syntax

```
-- within MAKE-ENDMAKE brackets

DZWELL
   zmin  zmax /

================================================================================

     zmin - depth of the upper end of straight line visualizing a well
     zmax - depth of the bottom end of straight line visualizing a well
```

EDIT

Opens the EDIT section.

EDIT syntax

```
-- after GRID section

EDIT
```

EHYSTR

Changes parameters of the hysteresis model. Note that the hysteresis option is enabled by the HYSTER keyword.

EHYSTR syntax

```
-- in the PROPS section

EHYSTR
   curv_cap  model  curv_kil  mod_kil /

================================================================================

       curv_cap    - not used at present;
          model    - the hysteresis model ID.
                     Can accept:
                       2  - Killoughs model is used for the non-wetting phase,
                            and the drainage curve is used for the wetting
                            phase (default option);
                       3  - Killoughs model is used for the non-wetting phase,
                            and the imbibition curve is used for the wetting
                            phase;
                       4  - Killoughs model is used for both wetting and
                            non-wetting phases;
          curv_kil - the curvature parameter for the Killoughs model
                     (the default is 1.0);
          mod_kil  - the modification parameter for the Killoughs model
                     (the default is 0.1).

```

ENDBOX

Resets the current input box to default position.

ENDBOX syntax

```
-- in all sections except RUNSPEC and POST

ENDBOX
```

ENDFIN

Resumes the active grid to the initial grid encompassing the whole reservoir.

ENDFIN syntax

```
-- in every section except RUNSPEC and POST

ENDFIN
```

ENDSCALE

Enables the end-point scaling option.

ENDSCALE syntax

```
-- in RUNSPEC section

ENDSCALE
/
```

ENTHTAB

For additional phase loads the specific enthalpy on pressure and temperature. This keyword must be used within PVT-ENDPVT brackets. See also the ADDPHASE keyword.

ENTHTAB syntax

```
-- within EOS-ENDEOS, or PVT-ENDPVT brackets, or in the PROPS section

ENTHTAB
        temp1  temp2  temp3  ...  /
 pres1  enth11 enth12 enth13  ...  /
 pres2  enth21 enth22 enth23  ...  /
 pres3  enth31 enth32 enth33  ...  /
 ...    ...    ...    ...    ...  /
/

================================================================================

     temp#  -  temperature (degree Kelvin);
     pres$  -  pressure;
     enth$# -  specific enthalpy for a given pressure (pres$) and
               temperature (temp#).
```

EOS-ENDEOS

EOS-ENDEOS brackets confine a section of the RUN-file for input of thermophysical properties of reservoir fluid. These brackets should be used in the PROPS section the same number of times as the number of EOSNUM regions used in the simulation.

EOS-ENDEOS syntax

```
-- in section PROPS

EOS
  eosnum /

-- keywords for input of thermophysical properties of reservoir fluid

ENDEOS

================================================================================

     eosnum - EOSNUM region ID for which the properties are specified.
```

EOSNUM

Specifies the EOSNUM property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

EOSNUM syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

EOSNUM
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

EOSNUM
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

EOSNUM
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

EQL-ENDEQL

EQL-ENDEQL brackets confine a section of the RUN-file for input of initial equilibration data. These brackets should be used in the INIT section the same number of times as the number of EQLNUM regions used in the simulation.

EQL-ENDEQL syntax

```
-- in INIT section

EQL
  eqlnum /

-- keywords for equilibration data specification

ENDEQL

/
================================================================================

     eqlnum - EQLNUM region number for which the initial equilibration data are
              specified.
```

EQLNUM

Specifies the EQLNUM property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

EQLNUM syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

EQLNUM
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

EQLNUM
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

EQLNUM
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

EQLZCORN

Equate the corner point depths to a user defined value in the input box.

EQLZCORN syntax

```
-- within MAKE-ENDMAKE brackets

EQLZCORN
  value1  imin1 imax1 jmin1 jmax1 kmin1 kmax1 m1_1 m2_1 m3_1 m4_1 m5_1 m6_1 /
  value2  imin2 imax2 jmin2 jmax2 kmin2 kmax2 m1_2 m2_2 m3_2 m4_2 m5_2 m6_2 /
  value3  imin3 imax3 jmin3 jmax3 kmin3 kmax3 m1_3 m2_3 m3_3 m4_3 m5_3 m6_3 /
  ...
/

================================================================================

   value#      - the value to which ZCORN array is equated in the input box;
   imin#/imax# - the boundaries of the input box along i-indexation axis.
                 By default these values are equal to the arguments 1 and 2
                 of the keyword BOX.
   jmin#/jmax# - the boundaries of the input box along j-indexation axis.
                 By default these values are equal to the arguments 3 and 4
                 of the keyword BOX.
   kmin#/kmax# - the boundaries of the input box along k-indexation axis.
                 By default these values are equal to the arguments 5 and 6
                 of the keyword BOX.
   mi_#        - (i=1,...,6). The mode:
                 'I-' - the operation is also applied to the adjacent face of
                        the grid block connected to the considered block in
                        the input box in the negative direction of i-index
                        axis.
                 'I+' - the operation is also applied to the adjacent face of
                        the grid block connected to the considered block in
                        the input box in the positive direction of i-index
                        axis.
                 'J-' - the operation is also applied to the adjacent face of
                        the grid block connected to the considered block in
                        the input box in the negative direction of j-index
                        axis.
                 'J+' - the operation is also applied to the adjacent face of
                        the grid block connected to the considered block in
                        the input box in the positive direction of j-index
                        axis.
                 'K-' - the operation is also applied to the adjacent face of
                        the grid block connected to the considered block in
                        the input box in the negative direction of k-index
                        axis.
                 'K+' - the operation is also applied to the adjacent face of
                        the grid block connected to the considered block in
```

```
45                          the input box in the positive direction of k-index
46                          axis.
```

EQUALNAM

Assigns the value of a property for all cells which character names (if assigned) belongs to character template.

EQUALNAM syntax

```
-- in all sections except RUNSPEC and POST

EQUALNAM
   mnemonic1  value1  template1 /
   mnemonic2  value2  template2 /
   mnemonic3  value3  template3 /
   ...
/

================================================================================

   mnemonic#   - mnemonic of the property which is modified;
   value#      - value assigned to the property;
   template#   - character name template.

================================================================================

   The keyword results in the following:

       mnemonic1:=value1 for all cells which character name (if it
                                                 is assigned) belong to template1;
       mnemonic2:=value2 for all cells which character name (if it
                                                 is assigned) belong to template2;
       mnemonic3:=value3 for all cells which character name (if it
                                                 is assigned) belong to template3.
       ...
```

EQUALREG

Assigns the value of a property in a region. The region should be one of FLUXNUM, ROCKNUM, ACTNUM, etc.

EQUALREG syntax

```
-- in all sections except RUNSPEC and POST

EQUALREG
  mnemonic1  value1  region1  regionID1 /
  mnemonic2  value2  region2  regionID2 /
  mnemonic3  value3  region3  regionID3 /
  ...
/

=============================================================================

    mnemonic# - mnemonic of the property which is modified;
    value#    - value assigned to the property in the region;
    region#   - mnemonic of the region in which the property is modified;
    regionID# - region number.

=============================================================================

   The keyword results in the following:

       mnemonic1:=value1  in the region region1=regionID1
       mnemonic2:=value2  in the region region2=regionID2
       mnemonic3:=value3  in the region region3=regionID3
       ...
```

EQUALS

Assigns the value of a property in a box of cells.

EQUALS syntax

```
-- in all sections except RUNSPEC and POST

EQUALS
   mnemonic1  value1  imin1 imax1   jmin1 jmax1   kmin1 kmax1 /
   mnemonic2  value2  imin2 imax2   jmin2 jmax2   kmin2 kmax2 /
   mnemonic3  value3  imin3 imax3   jmin3 jmax3   kmin3 kmax3 /
   ...
/

================================================================================

   mnemonic#   - mnemonic of the property which is modified.
   value#      - value assigned to the property in the current input box.
   imin#/imax# - the boundaries of the input box along i-indexation axis.
                 By default these values are equal to the arguments 1 and 2
                 of the keyword BOX.
   jmin#/jmax# - the boundaries of the input box along j-indexation axis.
                 By default these values are equal to the arguments 3 and 4
                 of the keyword BOX.
   kmin#/kmax# - the boundaries of the input box along k-indexation axis.
                 By default these values are equal to the arguments 5 and 6
                 of the keyword BOX.

================================================================================

   The keyword results in the following:

       mnemonic1:=value1  in the box [imin1,imax1]*[jmin1,jmax1]*[kmin1,kmax1]
       mnemonic2:=value2  in the box [imin2,imax2]*[jmin2,jmax2]*[kmin2,kmax2]
       mnemonic3:=value3  in the box [imin3,imax3]*[jmin3,jmax3]*[kmin3,kmax3]
       ...
```

EV

Calculates an integral quantity in a box of grid blocks and saves it into auxiliary variable (see AUXVAR keyword).

EV syntax

```
-- in all sections except RUNSPEC and POST

EV
  varname1 property1 oper1 imin1 imax1 jmin1 jmax1 kmin1 kmax1 /
  varname2 property2 oper2 imin2 imax2 jmin2 jmax2 kmin2 kmax2 /
  varname3 property3 oper3 imin3 imax3 jmin3 jmax3 kmin3 kmax3 /
   ...
/

================================================================================

   varname#    - auxiliary variable.
   property#   - mnemonic of the property.
   oper#       - mnemonic of operation which is applied. The operation is one
                 of the following applied to the region
                 imin<=i<=imax, jmin<=j<=jmax, kmin<=k<=kmax:
                        AVERAGE - calculates the average by cell volume.
                        AVG - calculates the average by pore volume.
                        MEAN - calculates the average value by the number of
                               cases
                        MAX - calculates the maximum.
                        MIN - calculates the minimum.
                        NUMB - calculates number of cells (cases).
                        SUM - sums the quantity.
   imin#/imax# - the boundaries of the input box along i-indexation axis.
                 By default these values are equal to the arguments 1 and 2
                 of the keyword BOX.
   jmin#/kmax# - the boundaries of the input box along j-indexation axis.
                 By default these values are equal to the arguments 3 and 4
                 of the keyword BOX.
   kmin#/kmax# - the boundaries of the input box along k-indexation axis.
                 By default these values are equal to the arguments 5 and 6
                 of the keyword BOX.
```

EVAUTO

At the end of every timestep automatically calculates an integral quantity in a box of grid blocks and saves it into auxiliary variable (see AUXVAR keyword).

EVAUTO syntax

```
-- in the INIT or SCHEDULE section

EVAUTO
  varname1 property1 oper1 imin1 imax1 jmin1 jmax1 kmin1 kmax1 /
  varname2 property2 oper2 imin2 imax2 jmin2 jmax2 kmin2 kmax2 /
  varname3 property3 oper3 imin3 imax3 jmin3 jmax3 kmin3 kmax3 /
   ...
/

================================================================================

   varname#    - auxiliary variable.
   property#   - mnemonic of the property.
   oper#       - mnemonic of operation which is applied. The operation is one
                 of the following applied to the region
                 imin<=i<=imax, jmin<=j<=jmax, kmin<=k<=kmax:
                        AVERAGE - calculates the average by cell volume.
                        AVG - calculates the average by pore volume.
                        MEAN - calculates the average value by the number of
                               cases
                        MAX - calculates the maximum.
                        MIN - calculates the minimum.
                        NUMB - calculates number of cells (cases).
                        SUM - sums the quantity.
   imin#/imax# - the boundaries of the input box along i-indexation axis.
                 By default these values are equal to the arguments 1 and 2
                 of the keyword BOX.
   jmin#/kmax# - the boundaries of the input box along j-indexation axis.
                 By default these values are equal to the arguments 3 and 4
                 of the keyword BOX.
   kmin#/kmax# - the boundaries of the input box along k-indexation axis.
                 By default these values are equal to the arguments 5 and 6
                 of the keyword BOX.
```

EVAUTREG

At the end of every timestep automatically calculates an integral quantity for selected region and saves it into auxiliary variable (see AUXVAR keyword).

EVAUTREG syntax

```
-- in the INIT or SCHEDULE section

EVAUTREG
  varname1 property1 region1  regionID1  oper1 /
  varname2 property2 region2  regionID2  oper2 /
  varname3 property3 region3  regionID3  oper3 /
   ...
/

================================================================================

   varname#    - auxiliary variable.
   property#   - mnemonic of the property.
   region#     - mnemonic of the region in which the property is modified;
   regionID#   - region number;
   oper#       - mnemonic of operation which is applied. The operation is one
                 of the following applied to the region region#=regionID#:
                        AVERAGE - calculates the average by cell volume.
                        AVG - calculates the average by pore volume.
                        MEAN - calculates the average value by the number of
                               cases
                        MAX - calculates the maximum.
                        MIN - calculates the minimum.
                        NUMB - calculates number of cells (cases).
                        SUM - sums the quantity.

```

EVNAM

Calculates an integral quantity for selected named elements and saves it into auxiliary variable (see AUXVAR keyword).

EVNAM syntax

```
-- in all sections except RUNSPEC and POST

EVNAM
  varname1 property1 template1  oper1 /
  varname2 property2 template2  oper2 /
  varname3 property3 template3  oper3 /
   ...
/

================================================================================

   varname#    - auxiliary variable.
   property#   - mnemonic of the property.
   template#   - character name template.
   oper#       - mnemonic of operation which is applied. The operation is one
                 of the following applied to the named elements belonging
                 to the template#:
                        AVERAGE - calculates the average by cell volume.
                        AVG - calculates the average by pore volume.
                        MEAN - calculates the average value by the number of
                               cases
                        MAX - calculates the maximum.
                        MIN - calculates the minimum.
                        NUMB - calculates number of cells (cases).
                        SUM - sums the quantity.
```

EVREG

Calculates an integral quantity for selected region and saves it into auxiliary variable (see AUXVAR keyword).

EVREG syntax

```
-- in all sections except RUNSPEC and POST

EVREG
  varname1 property1 region1  regionID1  oper1 /
  varname2 property2 region2  regionID2  oper2 /
  varname3 property3 region3  regionID3  oper3 /
   ...
/

================================================================================

   varname#    - auxiliary variable.
   property#   - mnemonic of the property.
   region#     - mnemonic of the region in which the property is modified;
   regionID#   - region number;
   oper#       - mnemonic of operation which is applied. The operation is one
                 of the following applied to the region region#=regionID#:
                        AVERAGE - calculates the average by cell volume.
                        AVG - calculates the average by pore volume.
                        MEAN - calculates the average value by the number of
                               cases
                        MAX - calculates the maximum.
                        MIN - calculates the minimum.
                        NUMB - calculates number of cells (cases).
                        SUM - sums the quantity.

```

EXPORT

Exports a property distribution in a formatted file. The export is provided for the current input box (see the keyword BOX). Grid blocks are ordered that i-index is cycling the fastest following by the j- and k-indexes.

EXPORT syntax

```
-- in all sections except RUNSPEC and POST

EXPORT
  mnemonic defvalue  dir  filename  ncol  elf format append /

================================================================================

   mnemonic  - mnemonic of the property to be exported;
   defvalue  - default value used for inactive/not existing grid blocks;
   dir       - directory in which the output file (with default name) is saved;
   filename  - output file name. If specified the previous argument is override;
   ncol      - number of columns in the out file (by default equal to the
               number of grid blocks along X axis in current input box;
   elf       - 'empty line flag'. If 'T' (default value) an empty line is
               output when the k-index is increased. The 'F' flag diables
               this option;
   format    - enables to change the output format (the default value
               recommended);
   append    - if F (default value) the file is cleared (rewind) before
               saving the data;
             - if T the exported data are appended to the file.
```

FAST

This option may facilitate a faster simulation.

FAST syntax

```
-- in RUNSPEC section

FAST
```

FAULTS

Defines the fault faces. See also the MULTFLT keyword.

FAULTS syntax

```
-- within MAKE-ENDMAKE brackets

FAULTS
   name1 imin1 imax1 jmin1 jmax1 kmin1 kmax1 face1 /
   name2 imin2 imax2 jmin2 jmax2 kmin2 kmax2 face2 /
   name3 imin3 imax3 jmin3 jmax3 kmin3 kmax3 face3 /
   ...
/

================================================================================

   name#       - a character ID of the fault;
   imin#-imax# - the boundaries of the input box along i-indexation axis.
                 By default these values are equal to '1' and the 2nd
                 argument of the keyword MAKE, respectively;
   jmin#-jmax# - the boundaries of the input box along j-indexation axis.
                 By default these values are equal to '1' and the 3rd
                 argument of the keyword MAKE, respectively;
   kmin#-kmax# - the boundaries of the input box along k-indexation axis.
                 By default these values are equal to '1' and the 4th
                 argument of the keyword MAKE, respectively;
   face#       - falut face tag. Must be one of
                 'I-','X-' - fault face coincides with the grid block face in
                             the negative direction of the i-index coordinate
                             line;
                 'I+','X+' - fault face coincides with the grid block face in
                             the positive direction of the i-index coordinate
                             line;
                 'J-','Y-' - fault face coincides with the grid block face in
                             the negative direction of the j-index coordinate
                             line;
                 'J+','Y+' - fault face coincides with the grid block face in
                             the positive direction of the j-index coordinate
                             line;
                             line;
                 'K-','Z-' - fault face coincides with the grid block face in
                             the negative direction of the k-index coordinate
                             line;
                 'K+','Z+' - fault face coincides with the grid block face in
                             the positive direction of the k-index coordinate
                             line
                             (in the input box).
```

FIELDSEP

Changes the field separator conditions. By default the field separator works as 1-stage separation to standard conditions (see STCOND). Note, that currently the separators are enabled for use only with the COMPS module.

FIELDSEP syntax

```
-- in the SCHEDULE section

FIELDSEP
   stage1 tempc1 pres1 liqstg1  gasstg1  2*nu1  eosnum1 /
   stage2 tempc2 pres2 liqstg2  gasstg2  2*nu2  eosnum2 /
   stage3 tempc3 pres3 liqstg3  gasstg3  2*nu3  eosnum3 /
...                                                    /
/

================================================================================

    stage#     - the separator stage index (i.e. 1,2,3,4,5, or 6). Default is
                 1.
    tempc#     - the stage temperature (relative). Default is the standard
                 temperature.
    pres#      - the stage pressure. Default is the standard pressure.
    liqstg#    - the index of the destination stage for liquid. Default is 0 -
                 means that liquid flows to the next stage. For the last stage
                 0 means 'to the stock tank'.
    gasstg#    - the index of the destination stage for gas. Default is 0,
                 which means 'add to separator gas'.
    nu#        - not used at present.
    eosnum#    - equation of state number (default 1).
```

FIPNUM

Specifies the FIPNUM property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

FIPNUM syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

             -- if the LOADING switch is in IJK position --

FIPNUM
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

             -- if the LOADING switch is in CELLID position --

FIPNUM
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

FIPNUM
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

FLUXNUM

Specifies the FLUXNUM property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

FLUXNUM syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

FLUXNUM
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

FLUXNUM
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

FLUXNUM
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

FUNCNAME

Defines the user function ID loaded by the succeeding FUNCSPEC keyword. The function can be used by the keywords OPERATE, OPERAREG or OPERANAM if the operation is USERFUNC.

FUNCNAME syntax

```
-- in all sections except RUNSPEC and POST

FUNCNAME
  funcid /

================================================================================

  funcid - a user function id (must be 1,2,3,...,100; default 1). The function
           id affect the data loaded by the FUNCSPEC keyword.
```

FUNCSPEC

Defines in a tabular form the a function. The function ID is defined by the preceding FUNCNAME keyword. The function can be used by the keywords OPERATE, OPERAREG or OPERANAM if the operation is USERFUNC.

FUNCSPEC syntax

```
-- in all sections except RUNSPEC and POST

FUNCSPEC
  arg1  value1 /
  arg2  value2 /
  arg3  value3 /
  ...
/

================================================================================

  arg#   - the argument;
  value# - the function value.
```

GDFILE

Loads the EGRID file (the simulation gird in a standard binary format).

GDFILE syntax

```
-- within brackets MAKE-ENDMAKE

GDFILE
   path_to_file /

================================================================================

    path_to_file     - path to the grid file of EGRID format.

```

GEOSNUM

Specifies equation of state used for calculations of the well group parameters. By default each group uses the first equation of state defined within the EOS-ENDEOS brackets.

GEOSNUM syntax

```
-- in the SCHEDULE section

GEOSNUM
    template1 eosnum1 /
    template2 eosnum2 /
    template3 eosnum3 /
...                   /
/

================================================================================

    template#  - well group template (8-byte character).
    eosnum#    - equation of state region ID.
```

GETPHASE

Allows to define additional mnemonics for the properties of phases (PHASES option). This keyword defines cells form which the parameters are copied in the INIT section to define the phases.

GETPHASE syntax

```
-- in PROPS section

GETPHASE
   name1 charname1 cellid1 nu1 pres1 /
   name2 charname2 cellid2 nu2 pres2 /
   name3 charname3 cellid3 nu3 pres3 /
   ...
/

================================================================================

   name#     - phase name (4-byte character);
   charname# - character name of a cell in which the phase properties will
               be given (8-byte character);
   cellid#   - id of a cell in which the phase properties will be given;
   nu#       - not used at present;
   pres#     - this arguments allows to change the pressure when defining the
               properties of the phase.
```

GETWELLS

Read wells defined in the SCHEDULE section.

GETWELLS syntax

```
-- within MAKE-ENDMAKE brackets

GETWELLS
```

GO2DATE

Defines a target report date and advances the simulation further reporting every specified time interval. See also the GO2STEP and GO2TIME keywords.

GO2DATE syntax

```
-- in SCHEDULE section

GO2DATE
   day  month  year  time  sec  rep_step /

================================================================================

    day      - calendar day   (1,2,3,...,31)
    month    - calendar month (JAN,FEB,MAR,APR,MAY,JUN,JUL,AUG,SEP,OCT,NOV,DEC)
    year     - year (e.g., 2012)
    time     - time of the day in HH:MM format (default 12:00)
    sec      - number of seconds of float format (default 0.0)
    rep_step - target stepping of the report time

    The simulation data will be reported at
        cur_time+1*rep_step
        cur_time+2*rep_step
        cur_time+3*rep_step
        ...
        until the specified date is reached.
        The last report step can be less than rep_step in order to match the
        date. Here, cur_time is the simulation time at which GO2DATE is
        encounted.
```

GO2STEP

Defines a target report time and advances the simulation further reporting every specified time interval. See also the GO2DATE and GO2TIME keywords.

GO2STEP syntax

```
-- in SCHEDULE section

GO2STEP
   step  rep_step /

================================================================================

        step - specifies one of the next report times = cur_time+step
    rep_step - target stepping of the report time

    The simulation data will be reported at
        cur_time+1*rep_step
        cur_time+2*rep_step
        cur_time+3*rep_step
        ...
        until the specified time=cur_time+step is reached.
        The last report step can be less than rep_step in order to match the
        required time. Here, cur_time is the simulation time at which GO2STEP
        isencounted.
```

GO2TIME

Defines a target report time and advances the simulation further reporting every specified time interval. See also the GO2DATE and GO2STEP keywords.

GO2TIME syntax

```
-- in SCHEDULE section

GO2TIME
   time  rep_step /

================================================================================

        time - next target time
    rep_step - target stepping of the report time

    The simulation data will be reported at
        cur_time+1*rep_step
        cur_time+2*rep_step
        cur_time+3*rep_step
        ...
        until the specified time is reached.
        The last report step can be less than rep_step in order to match the
        required time. Here, cur_time is the simulation time at which GO2TIME
        isencounted.
```

GRAVDR

Activates gravity drainage option. The DUALPORO keyword must be specified before GRAVDR keyword.

GRAVDR syntax

```
-- in section RUNSPEC

GRAVDR
```

GRAVIMET

Enables the gravity changes calculation and sets the computation mode. The observation points for the gravity changes can be specified by the OBSPOINT keyword. The RPTOBS invokes the gravity changes reporting to the summary files.

GRAVIMET syntax

```
-- in the RUNSPEC section

GRAVIMET
   CompMode   Scale   DirX DirY   RotationX  RotationY  Resolution
   Symmetry  SymPoint(1)  SymPoint(2)  SymPoint(3)
             SymNormal(1) SymNormal(2) SymNormal(3) /

================================================================================

    CompMode    - Computation mode. Possible values are
                  1. 3D - the simulation is 3D. No symmetry assumed (default).
                  2. PLANE - A plane simulation e.g. in Oxz axes.
                  3. AXISYM - An axisymmetric study.
    Scale       - The gravity change is multiplied by the Scale (default 1.0).
                  This value is useful in the case of PLANE or AXISYM modes.
                  For example, if the domain thickness along Oy axis in the
                  PLANE mode is 100 meters, then specify 0.01 for the scale.
                  Othewise if 1/10th of the full circle is simulated in the
                  AXISYM study then set Scale=10.
                  ----------------------------------------------------------
    DirX/DirY   - The extrusion direction (X,Y) for the PLANE mode.
                  Default = (0,1). The vertical component of the direction is 0.
                  ----------------------------------------------------------
    RotationX/  - The X and Y coordinates on the vertical axis of rotation in
    RotationY     the AXISYM mode. Default=(0,0).
    Resolution  - An integer specifying the number of segement in which the
                  full circle is discretized for calculating the gravity
                  change in the AXISYM mode (default 12).
                  ----------------------------------------------------------
    Symmetry    - Enables accounting for the reflectional symmetry. This option
                  is useful if a half of the reservois (e.g. only at x>0) is
                  simulated but the other half (at x<0) must be accounted for
                  in the observation data. The Symmetry switch must be either
                  T (the option is on) or F (the option is off). The default
                  value is F. The plane of reflection is specified by the
                  following data items.
    SymPoint(i) - The X, Y, and Z coordinates of a point on the plane of
                  symmetry. Default is (0,0,0).
```

```
39    SymNormal(i) - The X, Y, and Z components of a vector perpendicular to the
40                   plane of symmetry.
```

GRAVITY

Redefines the gravity acceleration.

GRAVITY syntax

```
-- in RUNSPEC section

GRAVITY
   newgrav /

================================================================================

      newgrav - a new gravity acceleration value
```

GRDDISP

Enables the ground displacement (e.g. the surface uplift or subsidence) calculation and sets the computation mode. The observation points for the displacement can be specified by the OBSPOINT keyword. The RPTOBS invokes the ground displacement reporting to the summary files. The mechanical properties of rock are specified for each ROCKNUM region by the GRDPARAM keyword. The ground displacement is estimated using the Mindlin's solution and assuming flat surface of Earth and homogeneous mechanical properties of rock.

GRDDISP syntax

```
-- in the RUNSPEC section

GRDDISP
   CompMode   Scale   DirX DirY   RotationX  RotationY  Resolution
   Symmetry  SymPoint(1)  SymPoint(2)  SymPoint(3)
             SymNormal(1) SymNormal(2) SymNormal(3)
             SurfDepth /

================================================================================

    CompMode    - Computation mode. Possible values are
                  1. 3D - the simulation is 3D. No symmetry assumed (default).
                  2. PLANE - A plane simulation e.g. in Oxz axes.
                  3. AXISYM - An axisymmetric study.
    Scale       - The displacement is multiplied by the Scale (default 1.0).
                  This value is useful in the case of PLANE or AXISYM modes.
                  For example, if the domain thickness along Oy axis in the
                  PLANE mode is 100 meters, then specify 0.01 for the scale.
                  Othewise if 1/10th of the full circle is simulated in the
                  AXISYM study then set Scale=10.
                  ------------------------------------------------------------
    DirX/DirY   - The extrusion direction (X,Y) for the PLANE mode.
                  Default = (0,1). The vertical component of the direction is 0.
                  ------------------------------------------------------------
    RotationX/  - The X and Y coordinates on the vertical axis of rotation in
    RotationY     the AXISYM mode. Default=(0,0).
    Resolution  - An integer specifying the number of segement in which the
                  full circle is discretized for calculating the ground
                  displacement in the AXISYM mode (default 12).
                  ------------------------------------------------------------
    Symmetry    - Enables accounting for the reflectional symmetry. This option
                  is useful if a half of the reservois (e.g. only at x>0) is
```

```
                simulated but the other half (at x<0) must be accounted for
                in the observation data. The Symmetry switch must be either
                T (the option is on) or F (the option is off). The default
                value is F. The plane of reflection is specified by the
                following data items.
  SymPoint(i) - The X, Y, and Z coordinates of a point on the plane of
                symmetry. Default is (0,0,0).
 SymNormal(i) - The X, Y, and Z components of a vector perpendicular to the
                plane of symmetry.
                ---------------------------------------------------------
  SurfDepth   - Depth of the top surface for the Mindlin's solution. The
                default is 0.
```

GRDPARAM

Defines the mechanical properties of rocks for estimating the ground displacement (see GRDDISP keyword).

GRDPARAM syntax

```
-- within the ROCK-ENDROCK brackets

GRDPARAM
   mu  K  Ks  alphas /

================================================================================

    mu     - the shear modulus (GPA in METRIC units).
    K      - the drained bulk modulus (GPA in METRIC).
    Ks     - the bulk modulus of the solid (rock) phase (GPA in METRIC).
    alphas - the thermal expansion coefficient (1/K in METRIC).

```

GRID

Opens the GRID section.

GRID syntax

```
-- after section RUNSPEC

GRID
```

GRIDUNIT

Defines the units of the grid coordinates.

GRIDUNIT syntax

```
-- within MAKE-ENDMAKE brackets

GRIDUNIT
  units  arg2 /

================================================================================

   units  - The units used in the COORD and ZCORN keywords. Should be positioned
            before COORD and ZCORN. Possible values:
               METERS - default if DEFUNITS units are used;
               METRES - default if METRIC units are used;
               FEET   - default if FIELD units are used.
   arg2   - if GRID (default value) then coordinates in:
                COORD, SRCSPECG
                are given relative to the grid coordinate system;
            if MAP then coordinates in the specified keywords are given relative
                to the map.
```

GRUPINJE

Specifies injection controls for a well group. The individual controls and limits for wells are specified by the WELLINJE keyword.

GRUPINJE syntax

```
-- in SCHEDULE section

GRUPINJE
   name1 targ1 volrate1 massrate1 dimflag1 bal1 /
   name2 targ2 volrate2 massrate2 dimflag2 bal2 /
   name3 targ3 volrate3 massrate3 dimflag3 bal3 /
   ...
/

================================================================================

   name#     - group name or group name template;
   targ#     - well operational target. Available values: MASS - mass rate,
               RATE - volumetric rate at stock tank conditions;
   volrate#  - volumetric rate;
   massrate# - mass rate;
   dimflag#  - if LIQ (default) then the volrate# and vp# are in liquid rate
               units, if GAS then volrate# and vp# are in gas rate units;
   bal#      - this switch controls how the flow rate is balanced between the
               children of the group. Possible values
               POT - the flow rate is balanced using the well potential
               injection rates corresponding to the targ# (default option);
               THP - the tubing head pressure is kept constant for all wells
               under the group control. If this option is chosen every well
               under the group control must have a VFPINJ table associated
               with it.
```

GSEPCOND

Associates well groups with separators. The separators must be defined by the SEPCOND keyword.

GSEPCOND syntax

```
-- in the SCHEDULE section

GSEPCOND
    template1 sepname1 /
    template2 sepname2 /
    template3 sepname3 /
...                    /
/

================================================================================

    template#  - well group template (8-byte character).
    sepname#   - the name of the separator which will act for the well or well
                 group.
```

HCFLUID

Enables heat conduction in reservoir fluid.

HCFLUID syntax

```
-- in section RUNSPEC

HCFLUID
```

HCONDCFI

Specifies the HCONDCFI property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

HCONDCFI syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

HCONDCFI
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

HCONDCFI
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

HCONDCFI
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

HCONDCFJ

Specifies the HCONDCFJ property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

HCONDCFJ syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

HCONDCFJ
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

HCONDCFJ
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

HCONDCFJ
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

HCONDCFK

Specifies the HCONDCFK property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

HCONDCFK syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

             -- if the LOADING switch is in IJK position --

HCONDCFK
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

HCONDCFK
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

HCONDCFK
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

HCONDCFX

Specifies the HCONDCFX property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

HCONDCFX syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

HCONDCFX
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

HCONDCFX
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

HCONDCFX
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

HCONDCFY

Specifies the HCONDCFY property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

HCONDCFY syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

HCONDCFY
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

HCONDCFY
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

HCONDCFY
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

HCONDCFZ

Specifies the HCONDCFZ property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

HCONDCFZ syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

HCONDCFZ
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

HCONDCFZ
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

HCONDCFZ
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

HCROCK

Enables heat conduction in rocks.

HCROCK syntax

```
-- in section RUNSPEC

HCROCK
```

HIFIN

Defines the ratios of the grid block sizes along i-index direction. The keyword must be located immediately after the CARFIN keyword defining the local grid refinement.

HIFIN syntax

```
-- wihin MAKE-ENDMAKE brackets

HIFIN
   di1 di2 di3 ... diN /

================================================================================

    di# - the ratio of the grid block size along i-index direction;
    N   - number of grid blocks in local grid refinement along i-index
          direction (defined ny the 8th argument of the keyword CARFIN).
```

HJFIN

Defines the ratios of the grid block sizes along j-index direction. The keyword must be located immediately after the CARFIN keyword defining the local grid refinement.

HJFIN syntax

```
-- wihin MAKE-ENDMAKE brackets

HJFIN
   dj1 dj2 dj3 ... djN /

================================================================================

    dj# - the ratio of the grid block size along j-index direction;
    N   - number of grid blocks in local grid refinement along j-index
          direction (defined ny the 9th argument of the keyword CARFIN).
```

HKFIN

Defines the ratios of the grid block sizes along k-index direction. The keyword must be located immediately after the CARFIN keyword defining the local grid refinement.

HKFIN syntax

```
-- wihin MAKE-ENDMAKE brackets

HKFIN
   dk1 dk2 dk3 ... dkN /

================================================================================

    dk# - the ratio of the grid block size along k-index direction;
    N   - number of grid blocks in local grid refinement along k-index
          direction (defined ny the 10th argument of the keyword CARFIN).
```

HOSTINFO

Allows saving of the information about host computer for the license file. See also the LICENSE keyword.

HOSTINFO syntax

```
-- before RUNSPEC keyword

HOSTINFO
   filename /

================================================================================

    filename - name of the output file with information about the host.
               Default value: HostInfo.txt.
```

HUGE

Use this option when running ¿1 M grid blocks simulation or complicated simulations.

HUGE syntax

```
-- in RUNSPEC section

HUGE
```

HXFIN

Defines the ratios of the grid block sizes along X axis direction. The keyword must be located immediately after the CARFIN keyword defining the local grid refinement. This is synonym for the HIFIN keyword.

HXFIN syntax

```
-- wihin MAKE-ENDMAKE brackets

HXFIN
   dx1 dx2 dx3 ... dxN /

================================================================================

    dx# - the ratio of the grid block size along X axis direction;
    N   - number of grid blocks in local grid refinement along X axis
          direction (defined ny the 8th argument of the keyword CARFIN).
```

HYFIN

Defines the ratios of the grid block sizes along Y axis direction. The keyword must be located immediately after the CARFIN keyword defining the local grid refinement. This is synonym for the HJFIN keyword.

HYFIN syntax

```
-- wihin MAKE-ENDMAKE brackets

HYFIN
   dy1 dy2 dy3 ... dyN /

================================================================================

    dy# - the ratio of the grid block size along Y axis direction;
    N   - number of grid blocks in local grid refinement along Y axis
          direction (defined ny the 9th argument of the keyword CARFIN).
```

HYSTER

Enables the hysteresis option. The hysteresis model parameters can be altered by the EHYSTR keyword.

HYSTER syntax

```
-- in RUNSPEC section

HYSTER
```

HZFIN

Defines the ratios of the grid block sizes along Z axis direction. The keyword must be located immediately after the CARFIN keyword defining the local grid refinement. This is synonym for the HKFIN keyword.

HZFIN syntax

```
-- wihin MAKE-ENDMAKE brackets

HZFIN
   dz1 dz2 dz3 ... dzN /

================================================================================

    dz# - the ratio of the grid block size along Z axis direction;
    N   - number of grid blocks in local grid refinement along Z axis
          direction (defined ny the 10th argument of the keyword CARFIN).
```

IDTOMPI

Copies the CELLID property in the MPINUM property.

IDTOMPI syntax

```
-- within MAKE-ENDMAKE brackets

IDTOMPI
   kx ky kz /

================================================================================

    kx/ky/kz  -- the size of blocks, each get the same MPINUM.
                 (Default: 0/0/0)
```

ILUTDROP

Defines the ilut method drop-off threshold.

ILUTDROP syntax

```
-- in section SCHEDULE

ILUTDROP
   ilutdrop /

================================================================================

      ilutdrop - ilut drop threshold.
```

ILUTFILL

Defines the ilut method fill-in factor.

ILUTFILL syntax

```
-- in section SCHEDULE

ILUTFILL
   ilutfill /

================================================================================

      ilutfill - ilut fill factor.
```

IMBNUM

Specifies the IMBNUM property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

IMBNUM syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

IMBNUM
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

IMBNUM
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

IMBNUM
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

INACTIVE

Enables inactive cells loading in the simulator kernel. This keyword must be encompassed by the MAKE-ENDMAKE brackets.

INACTIVE syntax

```
within MAKE-ENDMAKE brackets

INACTIVE
```

INCONUM

Specifies the INCONUM property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

INCONUM syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

INCONUM
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

INCONUM
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

INCONUM
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

INIT

Opens the INIT section.

INIT syntax

```
-- after section PROPS

INIT
```

INITNAM

Invokes the primary variables initialization procedure for all cells with assigned character name.

INITNAM syntax

```
-- in SCHEDULE section

INITNAM
  name1 name2 name3 ... /

================================================================================

   name# - character name or name template.
```

ISOTHERM

Switches the program to isothermal modeling. This options is not available with the BINMIXT module.

ISOTHERM syntax

```
-- in RUNSPEC section

ISOTHERM
```

LICENSE

Loads the license file into the simulation. See also the HOSTINFO keyword.

LICENSE syntax

```
-- before RUNSPEC keyword

LICENSE
   licensefile /

================================================================================

    licensefile - name of the license file.
```

LOADING

Defines the arrays loading method. By default the arrays are loaded on block by block basis (IJK). For example, of the loading syntax see the keyword ACTNUM

LOADING syntax

```
-- in all section except POST

LOADING
   mode /

================================================================================

    mode = IJK    - load arrays on block by block basis
         = CELLID - load arrays by numerical IDs (e.g., CELLID)
```

LOADGRD

Loads a GRD file to define a 2D function. This keyword can be used as an alternative to the UF2DSPEC keyword.

LOADGRD syntax

```
-- in all sections except RUNSPEC and POST

LOADGRD
  filename /

================================================================================

  filename - name of the file.
```

LOADSUM

Loads a summary file.

LOADSUM syntax

```
-- in INIT section

LOADSUM
   filename   mnemonic  mnemmin  mnemmax/

================================================================================

   filename - summary file name;
   mnemonic - mnemonic name;
   mnemmin  - minimum mnemonic value;
   mnemmax  - maximum mnemonic value.

   The simulator loads parameters only to those cells in which
         mnemmin <= mnemonic <= mnemmax.
```

LTOSIGMA

Coefficients for sigma-factor calculation in double porosity simulations.

LTOSIGMA syntax

```
-- in the GRID section

LTOSIGMA
   fx fy fz notused mode /

================================================================================

   fx,fy,fz      -  coefficient for sigma parameter calculation using LX,LY,LZ
                    data (spacing between fractures);
   notused       -  not used at present;
   mode          -  calculation mode. Two options XONLY and ALL are available.
                    By default mode=XONLY.
```

LX

Specifies the LX property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

LX syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

LX
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

LX
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

LX
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

LY

Specifies the LY property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

LY syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

LY
   value1  value2  value3  ... value(nbox) /

/////////////////////////////////////////////////////////////////////////////////

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

LY
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

/////////////////////////////////////////////////////////////////////////////////

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

LY
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

LZ

Specifies the LZ property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

LZ syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

LZ
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

LZ
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

LZ
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

MAKE-ENDMAKE

MAKE-ENDMAKE brackets create a computational domain. The opening MAKE bracket has associated parameters describing the grid type and the number of grid blocks. The closing ENDMAKE bracket does not have associated data.

MAKE-ENDMAKE syntax

```
-- in GRID section

MAKE
  gridtype  ni nj nk  cpgname  mumpillr /

-- other keywords

ENDMAKE

================================================================================

    gridtype = CART   - Cartesian Grid
             = RADIAL - Radial Grid
             = CORNER - Corner-Point grid

          ni - number of grid blocks along i-indexation axis
          nj - number of grid blocks along j-indexation axis
          nk - number of grid blocks along k-indexation axis
     cpgname - corner point grid name
    numpillr - number of point used to define every pillar line (default 2).
               This parameter influences the syntax of COORD.
```

MAPAXES

Defines grid axes to map axes transformation. The units of the input arguments are defines by the keyword MAPUNITS. Note, that by default the grid Y-axis has opposite direction to the map Y-axis. The output in ParaView is in map axes.

MAPAXES syntax

```
-- within MAKE-ENDMAKE brackets

MAPAXES
  x1 y1 x2 y2 x3 y3 /

================================================================================

   x1/y1  - coordinates of a point on the Y axis relative to the map;
   x2/y2  - coordinates of the grid origin relative to the map;
   x3/y3  - coordinates of a point on the X axis relative to the map.

Default MAPAXES:

MAPAXES
  0 -1 0 0 1 0 / metres
```

MAPPER

Calculates a map of a quantity (e.g. an average permeability) in a given grid box. The calculation can be performed in an automated regime by placing AUTO ahead of MAPPER.

MAPPER syntax

```
-- in any section except RUNSPEC and POST

MAPPER
  Map1  Array1  Oper1 Var1 kflag1  imin1 imax1  jmin1 jmax1  kmin1 kmax1 /
  Map2  Array2  Oper2 Var2 kflag2  imin2 imax2  jmin2 jmax2  kmin2 kmax2 /
  Map3  Array3  Oper3 Var3 kflag3  imin3 imax3  jmin3 jmax3  kmin3 kmax3 /
  ...                                                                    /
/

================================================================================

              Map# - Name of the array where the map is saved.
            Array# - Name of the array from that the map is created.
             Oper# - Should be one of the following:
                         AVERAGE - calculates the average by cell volume.
                         AVG - calculates the average by pore volume.
                         MEAN - calculates the average value by the number of
                                cases
                         MAX - calculates the maximum.
                         MIN - calculates the minimum.
                         NUMB - calculates number of cells (cases).
                         SUM - sums the quantity.
              Var# - A scalar variable defined in AUXVAR can be placed here.
                     If it is placed, then all Map# quantities are summed over
                     the box and assigned to the variable (is empty by
                     default).
            kflag# - If true, then Map# values are assigned only to the grid
                     blocks at kmin#<=k<=kmax#. Default is false, i.e. the
                     Map# values are assigned to all grid blocks.
       imin#/imax# - the boundaries of the input box along i-indexation axis.
                     By default these values are equal to the arguments 1 and 2
                     of the keyword BOX.
       jmin#/kmax# - the boundaries of the input box along j-indexation axis.
                     By default these values are equal to the arguments 3 and 4
                     of the keyword BOX.
       kmin#/kmax# - the boundaries of the input box along k-indexation axis.
                     By default these values are equal to the arguments 5 and 6
                     of the keyword BOX.
```

MAPUNITS

Defines the units for the keyword MAPAXES arguments.

MAPUNITS syntax

```
-- within MAKE-ENDMAKE brackets

MAPUNITS
  units /

================================================================================

   units  - The units used in the MAPAXES keyword. Should be positioned before
            MAPAXES. Possible values:
               METERS - default if DEFUNITS units are used;
               METRES - default if METRIC units are used;
               FEET   - default if FIELD units are used.
```

MAXVALUE

Applies a maximum limit to a property in a box of cells.

MAXVALUE syntax

```
-- in all sections except RUNSPEC and POST

MAXVALUE
   mnemonic1  value1  imin1 imax1   jmin1 jmax1   kmin1 kmax1 /
   mnemonic2  value2  imin2 imax2   jmin2 jmax2   kmin2 kmax2 /
   mnemonic3  value3  imin3 imax3   jmin3 jmax3   kmin3 kmax3 /
   ...
/

================================================================================

   mnemonic#   - mnemonic of the property which is modified.
   value#      - maximum limit applied to the property in the current input box.
   imin#/imax# - the boundaries of the input box along i-indexation axis.
                 By default these values are equal to the arguments 1 and 2
                 of the keyword BOX.
   jmin#/jmax# - the boundaries of the input box along j-indexation axis.
                 By default these values are equal to the arguments 3 and 4
                 of the keyword BOX.
   kmin#/kmax# - the boundaries of the input box along k-indexation axis.
                 By default these values are equal to the arguments 5 and 6
                 of the keyword BOX.

================================================================================

   The keyword results in the following:

       mnemonic1:=min(mnemonic1,value1)
                          in the box [imin1,imax1]*[jmin1,jmax1]*[kmin1,kmax1]
       mnemonic2:=min(mnemonic2,value2)
                          in the box [imin2,imax2]*[jmin2,jmax2]*[kmin2,kmax2]
       mnemonic3:=min(mnemonic3,value3)
                          in the box [imin3,imax3]*[jmin3,jmax3]*[kmin3,kmax3]
       ...
```

MAXVANAM

Applies a maximum limit to a property for all cells which character names (if assigned) belongs to character template.

MAXVANAM syntax

```
-- in all sections except RUNSPEC and POST

MAXVANAM
   mnemonic1  value1  template1 /
   mnemonic2  value2  template2 /
   mnemonic3  value3  template3 /
   ...
/

================================================================================

   mnemonic#   - mnemonic of the property which is modified;
   value#      - maximum limit applied to the property;
   template#   - character name template.

================================================================================

   The keyword results in the following:

       mnemonic1:=min(mnemonic1,value1) for all cells which character name (if it
                                          is assigned) belong to template1;
       mnemonic2:=min(mnemonic2,value2) for all cells which character name (if it
                                          is assigned) belong to template2;
       mnemonic3:=min(mnemonic3,value3) for all cells which character name (if it
                                          is assigned) belong to template3.
       ...
```

MAXVAREG

Applies a maximum limit to a property in a region.

MAXVAREG syntax

```
-- in all sections except RUNSPEC and POST

MAXVAREG
  mnemonic1  value1  region1  regionID1 /
  mnemonic2  value2  region2  regionID2 /
  mnemonic3  value3  region3  regionID3 /
  ...
/

=============================================================================

     mnemonic# - mnemonic of the property which is modified;
     value#    - maximum limit applied to the property in the region;
     region#   - mnemonic of the region in which the property is modified;
     regionID# - region number.

=============================================================================

   The keyword results in the following:

       mnemonic1:=min(mnemonic1,value1)  in the region region1=regionID1
       mnemonic2:=min(mnemonic2,value2)  in the region region2=regionID2
       mnemonic3:=min(mnemonic3,value3)  in the region region3=regionID3
       ...
```

MDISPL

Specifies the longitudinal dispersivity as a function of the phase saturation and velocity for a particular rock type.

MDISPL syntax

```
-- within the ROCK-ENDROCK brackets

MDISPL
  pname  sat1    sat2    sat3    ...  /
  vel1   disp11  disp12  disp13  ...  /
  vel2   disp21  disp22  disp23  ...  /
  vel3   disp31  disp32  disp33  ...  /
  ...    ...     ...     ...     ...  /
/

================================================================================

    pname   - phase name, e.g.
                 BLACKOIL:  OIL, WAT, GAS
                 GASSTORE:  LIQ, GAS, SOL
                 T2EOS1:    WAT, VAP
                 SIMPLMOD, BINMIXT:  FLD
    sat#    - the phase saturation;
    vel$    - the phase true velocity u/phi/sat, where u - is the Darcy's
              velocity, phi is the porosity and sat is the phase
              saturation;
    disp$#  - dispersivity.
```

MDISPT

Specifies the transverse dispersivity as a function of the phase saturation and velocity for a particular rock type.

MDISPT syntax

```
-- within the ROCK-ENDROCK brackets

MDISPT
  pname  sat1    sat2    sat3    ...  /
  vel1   disp11  disp12  disp13  ...  /
  vel2   disp21  disp22  disp23  ...  /
  vel3   disp31  disp32  disp33  ...  /
  ...    ...     ...     ...     ...  /
/

================================================================================

    pname   - phase name, e.g.
                  BLACKOIL:  OIL, WAT, GAS
                  GASSTORE:  LIQ, GAS, SOL
                  T2EOS1:    WAT, VAP
                  SIMPLMOD, BINMIXT:  FLD
    sat#    - the phase saturation;
    vel$    - the phase true velocity u/phi/sat, where u - is the Darcy's
              velocity, phi is the porosity and sat is the phase
              saturation;
    disp$#  - dispersivity.
```

MECHDISP

Enables mechanical dispersion. The longitudinal and transverse dispersivities should be specified for every rock type with the MDISPL and MDISPT keywords. The not-specified dispersivities are assumed zero.

MECHDISP syntax

```
-- in section RUNSPEC

MECHDISP
```

MINC

Activates multiple interacting continua option in double porosity simulations. The DUALPORO keyword must be specified before this keyword. The input data specified by the MINC keyword can be alternatively specified with the NMATRIX and NMATOPTS keywords.

MINC syntax

```
-- in section RUNSPEC

MINC
   mincmax geometry volfrac(1) volfrac(2) ... volfrac(mincmax) /

================================================================================

    mincmax    - maximum number of matrix media when using multiple interacting
                 continuum option (MINC). This parameter should be greater than
                 1;
    geometry   - LINEAR - linear geomerty for nested matrix grid blocks (1D);
                 RADIAL - radial geometry for nested matrix grid blocks (2D);
                 SPHERICAL - spherical geometry for nested matrix grid blocks
                 (3D);
    volfrac(i) - volume fraction of the i-th matrix grid block when moving
                 from fracture media inside matrix media. These numbers should
                 be greater than 0. The simulator normalizes these input
                 numbers making its sum equal to 1.
```

MINCSIZE

Specifies the number of matrix media per grid block in double porosity simulations with activated MINC option.

MINCSIZE syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

             -- if the LOADING switch is in IJK position --

MINCSIZE
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

MINCSIZE
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

MINCSIZE
   filetype filename /

```

```
43 ================================================================================
44
45         This option enables the property loading from an include file
46
47         filetype = ASCII  - formatted include file option
48                  = BINARY - binary include file
49
50 >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

MINVALUE

Applies a minimum limit to a property in a box of cells.

MINVALUE syntax

```
-- in all sections except RUNSPEC and POST

MINVALUE
   mnemonic1  value1  imin1 imax1   jmin1 jmax1   kmin1 kmax1 /
   mnemonic2  value2  imin2 imax2   jmin2 jmax2   kmin2 kmax2 /
   mnemonic3  value3  imin3 imax3   jmin3 jmax3   kmin3 kmax3 /
   ...
/

================================================================================

   mnemonic#   - mnemonic of the property which is modified.
   value#      - minimum limit applied to the property in the current input box.
   imin#/imax# - the boundaries of the input box along i-indexation axis.
                 By default these values are equal to the arguments 1 and 2
                 of the keyword BOX.
   jmin#/jmax# - the boundaries of the input box along j-indexation axis.
                 By default these values are equal to the arguments 3 and 4
                 of the keyword BOX.
   kmin#/kmax# - the boundaries of the input box along k-indexation axis.
                 By default these values are equal to the arguments 5 and 6
                 of the keyword BOX.

================================================================================

   The keyword results in the following:

       mnemonic1:=max(mnemonic1,value1)
                          in the box [imin1,imax1]*[jmin1,jmax1]*[kmin1,kmax1]
       mnemonic2:=max(mnemonic2,value2)
                          in the box [imin2,imax2]*[jmin2,jmax2]*[kmin2,kmax2]
       mnemonic3:=max(mnemonic3,value3)
                          in the box [imin3,imax3]*[jmin3,jmax3]*[kmin3,kmax3]
       ...
```

MINVANAM

Applies a minimum limit to a property for all cells which character names (if assigned) belongs to character template.

MINVANAM syntax

```
-- in all sections except RUNSPEC and POST

MINVANAM
   mnemonic1  value1  template1 /
   mnemonic2  value2  template2 /
   mnemonic3  value3  template3 /
   ...
/

================================================================================

   mnemonic#   - mnemonic of the property which is modified;
   value#      - minimum limit applied to the property;
   template#   - character name template.

================================================================================

   The keyword results in the following:

       mnemonic1:=max(mnemonic1,value1) for all cells which character name (if it
                                                 is assigned) belong to template1;
       mnemonic2:=max(mnemonic2,value2) for all cells which character name (if it
                                                 is assigned) belong to template2;
       mnemonic3:=max(mnemonic3,value3) for all cells which character name (if it
                                                 is assigned) belong to template3.
       ...
```

MINVAREG

Applies a minimum limit to a property in a region.

MINVAREG syntax

```
-- in all sections except RUNSPEC and POST

MINVAREG
  mnemonic1  value1  region1  regionID1 /
  mnemonic2  value2  region2  regionID2 /
  mnemonic3  value3  region3  regionID3 /
  ...
/

================================================================================

     mnemonic# - mnemonic of the property which is modified;
     value#    - minimum limit applied to the property in the region;
     region#   - mnemonic of the region in which the property is modified;
     regionID# - region number.

================================================================================

   The keyword results in the following:

       mnemonic1:=max(mnemonic1,value1)  in the region region1=regionID1
       mnemonic2:=max(mnemonic2,value2)  in the region region2=regionID2
       mnemonic3:=max(mnemonic3,value3)  in the region region3=regionID3
       ...
```

MNEMPARA

Changes parameters of mnemonics

MNEMPARA syntax

```
-- in all sections except RUNSPEC

MNEMPARA
  mnemonic1  format1 /
  mnemonic2  format2 /
  mnemonic3  format3 /
  ...
/

================================================================================

     mnemonic#   - name of the mnemonic;
     format#     - output format in formatted output.

                   Possible values of format#:

                   1PEw.d (or Ew.d) - transfers real values with E exponent.
                   Can be applyied only to the Real data types.
                   w-d must be greater than 6 (Example: 1PE12.5).

                   Fw.d - transfers real values with no exponents.
                   Can be applyied only to the Real data types.
                   w-d must be greater than 3 (Example: F9.6).

                   Iw - transfers decimal integer values.
                   Can be applyied only to the Integer data types.
                   (Example: I8).

                   Aw - transfers characters.
                   Can be applyied only to the Character data types.
                   (Example: A8).

                   Here,
                   w is the field width,
                   d is the number of digits to the right of the decimal point.
```

MPINUM

Specifies the MPINUM property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

MPINUM syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

MPINUM
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

MPINUM
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

MPINUM
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

MULTFLT

Defines the transmissibility multipliers for the faults. See also the FAULTS keyword.

MULTFLT syntax

```
-- within MAKE-ENDMAKE brackets

MULTFLT
  name1  mult1 /
  name2  mult2 /
  name3  mult3 /
  ...
/

================================================================================

    name# - character ID of the fault;
    mult# - transmissibility multiplier across the fault.
```

MULTINAM

Multiplies a property by a constant for all cells which character names (if assigned) belongs to character template.

MULTINAM syntax

```
-- in all sections except RUNSPEC and POST

MULTINAM
   mnemonic1  value1  template1 /
   mnemonic2  value2  template2 /
   mnemonic3  value3  template3 /
   ...
/

================================================================================

   mnemonic#   - mnemonic of the property which is modified;
   value#      - value by which the property is multiplied;
   template#   - character name template.

================================================================================

   The keyword results in the following:

       mnemonic1:=mnemonic1*value1 for all cells which character name (if it
                                                 is assigned) belong to template1;
       mnemonic2:=mnemonic2*value2 for all cells which character name (if it
                                                 is assigned) belong to template2;
       mnemonic3:=mnemonic3*value3 for all cells which character name (if it
                                                 is assigned) belong to template3.
       ...
```

MULTIREG

Multiplies a property by a constant in a region.

MULTIREG syntax

```
-- in all sections except RUNSPEC and POST

MULTIREG
  mnemonic1  value1  region1  regionID1 /
  mnemonic2  value2  region2  regionID2 /
  mnemonic3  value3  region3  regionID3 /
  ...
/

================================================================================

     mnemonic# - mnemonic of the property which is modified;
     value#    - value by ehich the property is multiplied in the region;
     region#   - mnemonic of the region in which the property is modified;
     regionID# - region number.

================================================================================

   The keyword results in the following:

       mnemonic1:=mnemonic1*value1  in the region region1=regionID1
       mnemonic2:=mnemonic2*value2  in the region region2=regionID2
       mnemonic3:=mnemonic3*value3  in the region region3=regionID3
       ...
```

MULTIPLY

Multiplies a property by a constant in a box of cells.

MULTIPLY syntax

```
-- in all sections except RUNSPEC and POST

MULTIPLY
   mnemonic1  value1  imin1 imax1   jmin1 jmax1   kmin1 kmax1 /
   mnemonic2  value2  imin2 imax2   jmin2 jmax2   kmin2 kmax2 /
   mnemonic3  value3  imin3 imax3   jmin3 jmax3   kmin3 kmax3 /
   ...
/

================================================================================

   mnemonic#   - mnemonic of the property which is modified.
   value#      - value by which the property is multiplied in the current
                 input box.
   imin#/imax# - the boundaries of the input box along i-indexation axis.
                 By default these values are equal to the arguments 1 and 2
                 of the keyword BOX.
   jmin#/jmax# - the boundaries of the input box along j-indexation axis.
                 By default these values are equal to the arguments 3 and 4
                 of the keyword BOX.
   kmin#/kmax# - the boundaries of the input box along k-indexation axis.
                 By default these values are equal to the arguments 5 and 6
                 of the keyword BOX.

================================================================================

   The keyword results in the following:

       mnemonic1:=mnemonic1*value1
                          in the box [imin1,imax1]*[jmin1,jmax1]*[kmin1,kmax1]
       mnemonic2:=mnemonic2*value2
                          in the box [imin2,imax2]*[jmin2,jmax2]*[kmin2,kmax2]
       mnemonic3:=mnemonic3*value3
                          in the box [imin3,imax3]*[jmin3,jmax3]*[kmin3,kmax3]
       ...
```

MULTPV

Specifies the MULTPV property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

MULTPV syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

MULTPV
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

MULTPV
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

MULTPV
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

MULTX

Specifies the MULTX property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

MULTX syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

MULTX
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

MULTX
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

MULTX
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

MULTX-

Specifies the MULTX- property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

MULTX- syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

MULTX-
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

MULTX-
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

MULTX-
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

MULTY

Specifies the MULTY property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

MULTY syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

MULTY
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

MULTY
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

MULTY
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

MULTY-

Specifies the MULTY- property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

MULTY- syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

MULTY-
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

MULTY-
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

MULTY-
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

MULTZ

Specifies the MULTZ property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

MULTZ syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

MULTZ
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

MULTZ
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

MULTZ
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

MULTZ-

Specifies the MULTZ- property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

MULTZ- syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

MULTZ-
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

MULTZ-
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

MULTZ-
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

MUSEGWEL

Enables multisegmented wells. The well model (an averaged model or multisegmented) for every well is specified by the WELSPECS keyword.

MUSEGWEL syntax

```
-- in section RUNSPEC

MUSEGWEL
```

NEWTON

Defines some controls for the Newton method.

NEWTON syntax

```
-- in SCHEDULE section

NEWTON
   itmin   itqc   itmax /

================================================================================

      itmin - the minimum number of the Newton iteration at every time step;
      itqc  - the number of itertations for which the quadratic convergence is
              expected;
      itmax - the maximum number of the Newton iteration at every time step.
```

NIFIN

Defines the number of local grid blocks in every parent grid block of an LGR in i-index direction. The keyword must be located immediately after the CARFIN keyword defining the local grid refinement (LGR).

NIFIN syntax

```
-- wihin MAKE-ENDMAKE brackets

NIFIN
   ni1 ni2 ni3 ... niN /

================================================================================

    ni# - the number of local grid blocks in every parent grid block of an LGR
          in i-index direction;
    N   - number of grid blocks in local grid refinement along i-index
          direction (defined ny the 8th argument of the keyword CARFIN).
```

NJFIN

Defines the number of local grid blocks in every parent grid block of an LGR in j-index direction. The keyword must be located immediately after the CARFIN keyword defining the local grid refinement (LGR).

NJFIN syntax

```
-- wihin MAKE-ENDMAKE brackets

NJFIN
   nj1 nj2 nj3 ... njN /

/
------------------------------------------------------------------------------

    nj# - the number of local grid blocks in every parent grid block of an LGR
          in j-index direction;
    N   - number of grid blocks in local grid refinement along j-index
          direction (defined ny the 9th argument of the keyword CARFIN).
```

NKFIN

Defines the number of local grid blocks in every parent grid block of an LGR in k-index direction. The keyword must be located immediately after the CARFIN keyword defining the local grid refinement (LGR).

NKFIN syntax

```
-- wihin MAKE-ENDMAKE brackets

NKFIN
   nk1 nk2 nk3 ... nkN /

================================================================================

    nk# - the number of local grid blocks in every parent grid block of an LGR
          in k-index direction;
    N   - number of grid blocks in local grid refinement along k-index
          direction (defined ny the 10th argument of the keyword CARFIN).
```

NMATOPTS

Defines parameters of multiple interacting continua. The NMATRIX and DUALPORO keywords must be specified before this keyword. See also the MINC keyword as an alternative.

NMATOPTS syntax

```
-- in section RUNSPEC

NMATOPTS
   geometry not_used volfrac(1) volfrac(2) ... volfrac(mincmax) /

================================================================================

    geometry   - LINEAR - linear geomerty for nested matrix grid blocks (1D);
                 RADIAL - radial geometry for nested matrix grid blocks (2D);
                 SPHERICAL - spherical geometry for nested matrix grid blocks
                 (3D);
    not_used   - not used at present;
    volfrac(i) - volume fraction of the i-th matrix grid block when moving
                 from fracture media inside matrix media. These numbers should
                 be greater than 0. The simulator normalizes these input
                 numbers making its sum equal to 1.
```

NMATRIX

Activates multiple interacting continua option in double porosity simulations. The DUALPORO keyword must be specified before this keyword. Parameters of the MINC data can be specified with the NMATOPTS keyword. See also the MINC keyword as an alternative.

NMATRIX syntax

```
-- in section RUNSPEC

NMATRIX
   mincmax /

================================================================================

    mincmax    - maximum number of matrix media when using multiple interacting
                 continuum option (MINC). This parameter should be greater than
                 1.
```

NOAUTO

Disables automated operations loading.

NOAUTO syntax

```
-- in the INIT or SCHEDULE section of the H-program
-- within SCHED-ENDSCHED or OPTIM-ENDOPTIM bractkets of the U-program

NOAUTO
```

NOAUTOPA

Disables automatic grid decomposition by Metis when ENDMAKE bracket is encountered.

NOAUTOPA syntax

```
-- within MAKE-ENDMAKE brackets

NOAUTOPA
```

NOGRAV

Switches off calculation of the densities at the interfaces for modeling gravity effects.

NOGRAV syntax

```
-- in RUNSPEC section

NOGRAV
```

NOGRID

Disables GRID file output from the GRID section. See also the SAVEGRID keyword.

NOGRID syntax

```
in GRID section

NOGRID
```

NOMVS

Disables MVS file output from the current MAKE-ENDMAKE brackets. See also the SAVEMVS keyword.

NOMVS syntax

```
-- within MAKE-ENDMAKE brackets

NOMVS
```

NOMVSW

Disables MVS file for wells output from the current MAKE-ENDMAKE brackets. See also the SAVEMVSW keyword.

NOMVSW syntax

```
-- within MAKE-ENDMAKE brackets

NOMVSW
```

NOSIM

Disables simulation progress. The program runs only input data checking mode.

NOSIM syntax

```
-- in RUNSPEC or SCHEDULE section

NOSIM
```

NTG

Specifies the NTG property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

NTG syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

NTG
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

NTG
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

NTG
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

NUPCOL

Controls calculation of the well tagets in the group tree.

NUPCOL syntax

```
-- in the SCHEDULE section

NUPCOL
  nupcol  nupcol2 /

================================================================================

    nupcol  - number of first non-linear iterations over which the group tree
              is rebalanced (by default nupcol=2).
    nupcol2 - number of first non-linear iterations over which the well
              potentials (source potentials) are updated (by default
              nupcol2=0, i.e. the potentials are updated only at the beginning
              of each time step.
```

NXFIN

Defines the number of local grid blocks in every parent grid block of an LGR in X axis direction. The keyword must be located immediately after the CARFIN keyword defining the local grid refinement (LGR). This is synonym for the NIFIN keyword.

NXFIN syntax

```
-- wihin MAKE-ENDMAKE brackets

NXFIN
   nx1 nx2 nx3 ... nxN /

================================================================================

    nx# - the number of local grid blocks in every parent grid block of an LGR
          in X axis direction;
    N   - number of grid blocks in local grid refinement along X axis
          direction (defined ny the 8th argument of the keyword CARFIN).
```

NYFIN

Defines the number of local grid blocks in every parent grid block of an LGR in Y axis direction. The keyword must be located immediately after the CARFIN keyword defining the local grid refinement (LGR). This is synonym for the NJFIN keyword.

NYFIN syntax

```
-- wihin MAKE-ENDMAKE brackets

NYFIN
   ny1 ny2 ny3 ... nyN /

================================================================================

    ny# - the number of local grid blocks in every parent grid block of an LGR
          in Y axis direction;
    N   - number of grid blocks in local grid refinement along Y axis
          direction (defined ny the 9th argument of the keyword CARFIN).
```

NZFIN

Defines the number of local grid blocks in every parent grid block of an LGR in Z axis direction. The keyword must be located immediately after the CARFIN keyword defining the local grid refinement (LGR). This is synonym for the NKFIN keyword.

NZFIN syntax

```
-- wihin MAKE-ENDMAKE brackets

NZFIN
   nz1 nz2 nz3 ... nzN /

================================================================================

    nz# - the number of local grid blocks in every parent grid block of an LGR
          in Z axis direction;
    N   - number of grid blocks in local grid refinement along Z axis
          direction (defined ny the 10th argument of the keyword CARFIN).
```

OBJBLOC

Adds a grid block parameter for a given report time to the objective function.

OBJBLOC syntax

```
-- in POST section

OBJBLOC
   i1  j1  k1  gridname1  resname1 weight1 property1 target1 repframe1  /
   i2  j2  k2  gridname2  resname2 weight2 property2 target2 repframe2  /
   i3  j3  k3  gridname3  resname3 weight3 property3 target3 repframe3  /
   ...                                                                  /
/

================================================================================

      i#        - i-index of the grid block;
      j#        - j-index of the grid block;
      k#        - k-index of the grid block;
      gridname# - grid name (e.g., defined by the CARFIN keyword);
      resname#  - reservoir name;
      weight#   - weight of the residual when it is added to the
                  objective function;
      property# - a reported property for the specified well;
      target#   - target value of the property;
      repframe# - report time frame for which the property is reported.
```

OBJEWM

Reports the number of errors, warnings and messages to the OBJ-file.

OBJEWM syntax

```
-- in POST section

OBJEWM
```

OBJFLD

Adds a field parameter for a given report time to the objective function.

OBJFLD syntax

```
-- in POST section

OBJFLD
   weight1 property1 target1 repframe1  /
   weight2 property2 target2 repframe2  /
   weight3 property3 target3 repframe3  /
   ...                                     /
/

================================================================================

      weight#   - weight of the residual when it is added to the
                  objective function;
      property# - a reported field property;
      target#   - target value of the property;
      repframe# - report time frame for which the property is reported.
```

OBJFPCE

Adds a fluid-in-place region parameter for a given report time to the objective function.

OBJFPCE syntax

```
-- in POST section

OBJFPCE
   fipnum1 weight1 property1 target1 repframe1  /
   fipnum2 weight2 property2 target2 repframe2  /
   fipnum3 weight3 property3 target3 repframe3  /
   ...                                          /
/

================================================================================

      fipnum#   - fluid-in-place region number;
      weight#   - weight of the residual when it is added to the
                  objective function;
      property# - a reported property for the specified fluid-in-place region;
      target#   - target value of the property;
      repframe# - report time frame for which the property is reported.
```

OBJFPCO

Adds a parameter defined on the boundary between two fluid-in-place regions for a given report time to the objective function.

OBJFPCO syntax

```
-- in POST section

OBJFPCO
   fipnuma1  fipnumb1 weight1 property1 target1 repframe1  /
   fipnuma2  fipnumb2 weight2 property2 target2 repframe2  /
   fipnuma3  fipnumb3 weight3 property3 target3 repframe3  /
   ...                                                     /
/

================================================================================

     fipnuma#   - two fluid-in-place region numbers. The flow rate is reported
      -fipnumb#   in the direction from fipnuma# to fipnumb#.
      weight#   - weight of the residual when it is added to the
                  objective function;
      property# - a reported property for the specified fluid-in-place regions;
      target#   - target value of the property;
      repframe# - report time frame for which the property is reported.
```

OBJOBS

Adds a parameter of the observation point for a given report time to the objective function.

OBJOBS syntax

```
-- in POST section

OBJOBS
   obspoint1 weight1 property1 target1 repframe1  /
   obspoint2 weight2 property2 target2 repframe2  /
   obspoint3 weight3 property3 target3 repframe3  /
   ...                                           /
/

================================================================================

      obspoint# - name of the observation point;
      weight#   - weight of the residual when it is added to the
                  objective function;
      property# - a reported property for the specified source;
      target#   - target value of the property;
      repframe# - report time frame for which the property is reported.
```

OBJPARAM

Parameters of the objective function.

OBJPARAM syntax

```
-- in POST section

OBJPARAM
   n /

================================================================================

   n   - the power coefficient in the objective function.
```

OBJSRC

Adds a point source/sink parameter for a given report time to the objective function.

OBJSRC syntax

```
-- in POST section

OBJSRC
   srcname1 weight1 property1 target1 repframe1  /
   srcname2 weight2 property2 target2 repframe2  /
   srcname3 weight3 property3 target3 repframe3  /
   ...                                           /
/

================================================================================

       srcname# -  point source/sink name;
      weight#   - weight of the residual when it is added to the
                  objective function;
      property# - a reported property for the specified source;
      target#   - target value of the property;
      repframe# - report time frame for which the property is reported.
```

OBJTBLOC

Adds a grid block parameter values over a time interval to the objective function. The target values for every report time are given by the interpolation through the user defined function. Compare to OBJXBLOC.

OBJTBLOC syntax

```
-- in POST section

OBJTBLOC
   i1  j1  k1  gridname1  resname1 weight1 propertyX1 propertyY1 UserFunc1
                                                    FirstFrame1 LastFrame1 /
   i2  j2  k2  gridname2  resname2 weight2 propertyX2 propertyY2 UserFunc2
                                                    FirstFrame2 LastFrame2 /
   i3  j3  k3  gridname3  resname3 weight3 propertyX3 propertyY3 UserFunc3
                                                    FirstFrame3 LastFrame3 /
   ...                                                                     /
/

================================================================================

      i#        - i-index of the grid block;
      j#        - j-index of the grid block;
      k#        - k-index of the grid block;
      gridname# - grid name (e.g., defined by the CARFIN keyword);
      resname#  - reservoir name;
      weight#   - weight of the residual when it is added to the
                  objective function;
     propertyX# - the property of the UserFunc# argument;
     propertyY# - the property of the UserFunc# value;
      UserFunc# - the user function ID previously specified with FUNCSPEC
                  keyword. The simulation to be calibrated to that function;
    FirstFrame# - first report frame of the fitting interval;
     LastFrame# - last report frame of the fitting interval.
```

OBJTFLD

Adds a field parameter values over a time interval to the objective function. The target values for every report time are given by the interpolation through the user defined function.

OBJTFLD syntax

```
-- in POST section

OBJTFLD
   weight1 propertyX1 propertyY1 UserFunc1 FirstFrame1 LastFrame1 /
   weight2 propertyX2 propertyY2 UserFunc2 FirstFrame2 LastFrame2 /
   weight3 propertyX3 propertyY3 UserFunc3 FirstFrame3 LastFrame3 /
   ...                                                           /
/

================================================================================

      weight#   - weight of the residual when it is added to the
                  objective function;
     propertyX# - the property of the UserFunc# argument;
     propertyY# - the property of the UserFunc# value;
      UserFunc# - the user function ID previously specified with FUNCSPEC
                  keyword. The simulation to be calibrated to that function;
    FirstFrame# - first report frame of the fitting interval;
     LastFrame# - last report frame of the fitting interval.
```

OBJTFPCE

Adds a fluid-in-place region parameter values over a time interval to the objective function. The target values for every report time are given by the interpolation through the user defined function.

OBJTFPCE syntax

```
-- in POST section

OBJTFPCE
   fipnum1 weight1 propertyX1 propertyY1 UserFunc1 FirstFrame1 LastFrame1 /
   fipnum2 weight2 propertyX2 propertyY2 UserFunc2 FirstFrame2 LastFrame2 /
   fipnum3 weight3 propertyX3 propertyY3 UserFunc3 FirstFrame3 LastFrame3 /
   ...                                                                    /
/

================================================================================

      fipnum#   - fluid-in-place region number;
      weight#   - weight of the residual when it is added to the
                  objective function;
     propertyX# - the property of the UserFunc# argument;
     propertyY# - the property of the UserFunc# value;
      UserFunc# - the user function ID previously specified with FUNCSPEC
                  keyword. The simulation to be calibrated to that function;
    FirstFrame# - first report frame of the fitting interval;
     LastFrame# - last report frame of the fitting interval.
```

OBJTFPCO

Adds a parameter defined on the boundary between two fluid-in-place regions over a time interval to the objective function. The target values for every report time are given by the interpolation through the user defined function.

OBJTFPCO syntax

```
-- in POST section

OBJTFPCO
   fipnuma1  fipnumb1 weight1 propertyX1 propertyY1 UserFunc1
                                                    FirstFrame1 LastFrame1 /
   fipnuma2  fipnumb2 weight2 propertyX2 propertyY2 UserFunc2
                                                    FirstFrame2 LastFrame2 /
   fipnuma3  fipnumb3 weight3 propertyX3 propertyY3 UserFunc3
                                                    FirstFrame3 LastFrame3 /
   ...                                                                     /
/

================================================================================

     fipnuma#   - two fluid-in-place region numbers. The flow rate is reported
      -fipnumb#   in the direction from fipnuma# to fipnumb#.
      weight#   - weight of the residual when it is added to the
                  objective function;
     propertyX# - the property of the UserFunc# argument;
     propertyY# - the property of the UserFunc# value;
      UserFunc# - the user function ID previously specified with FUNCSPEC
                  keyword. The simulation to be calibrated to that function;
    FirstFrame# - first report frame of the fitting interval;
     LastFrame# - last report frame of the fitting interval.
```

OBJTOBS

Adds a parameter of the observation point over a time interval to the objective function. The target values for every report time are given by the interpolation through the user defined function.

OBJTOBS syntax

```
-- in POST section

OBJTOBS
   obspoint1 weight1 propertyX1 propertyY1 UserFunc1 FirstFrame1 LastFrame1 /
   obspoint2 weight2 propertyX2 propertyY2 UserFunc2 FirstFrame2 LastFrame2 /
   obspoint3 weight3 propertyX3 propertyY3 UserFunc3 FirstFrame3 LastFrame3 /
   ...                                                                      /
/

================================================================================

      obspoint# - name of the observation point;
      weight#   - weight of the residual when it is added to the
                  objective function;
     propertyX# - the property of the UserFunc# argument;
     propertyY# - the property of the UserFunc# value;
      UserFunc# - the user function ID previously specified with FUNCSPEC
                  keyword. The simulation to be calibrated to that function;
    FirstFrame# - first report frame of the fitting interval;
     LastFrame# - last report frame of the fitting interval.
```

OBJTSRC

Adds a point source/sink parameter values over a time interval to the objective function. The target values for every report time are given by the interpolation through the user defined function.

OBJTSRC syntax

```
-- in POST section

OBJTSRC
   srcname1 weight1 propertyX1 propertyY1 UserFunc1 FirstFrame1 LastFrame1 /
   srcname2 weight2 propertyX2 propertyY2 UserFunc2 FirstFrame2 LastFrame2 /
   srcname3 weight3 propertyX3 propertyY3 UserFunc3 FirstFrame3 LastFrame3 /
   ...                                                                     /
/

============================================================================

      srcname#  - point source/sink name;
      weight#   - weight of the residual when it is added to the
                  objective function;
     propertyX# - the property of the UserFunc# argument;
     propertyY# - the property of the UserFunc# value;
      UserFunc# - the user function ID previously specified with FUNCSPEC
                  keyword. The simulation to be calibrated to that function;
    FirstFrame# - first report frame of the fitting interval;
     LastFrame# - last report frame of the fitting interval.
```

OBJTVAR

Adds an auxiliary variable over a time interval to the objective function. The target values for every report time are given by the interpolation through the user defined function.

OBJTVAR syntax

```
-- in POST section

OBJTVAR
   weight1 propertyX1 propertyY1 UserFunc1 FirstFrame1 LastFrame1 /
   weight2 propertyX2 propertyY2 UserFunc2 FirstFrame2 LastFrame2 /
   weight3 propertyX3 propertyY3 UserFunc3 FirstFrame3 LastFrame3 /
   ...                                                           /
/

================================================================================

      weight#   - weight of the residual when it is added to the
                  objective function;
     propertyX# - the property of the UserFunc# argument;
     propertyY# - the property of the UserFunc# value;
      UserFunc# - the user function ID previously specified with FUNCSPEC
                  keyword. The simulation to be calibrated to that function;
    FirstFrame# - first report frame of the fitting interval;
     LastFrame# - last report frame of the fitting interval.
```

OBJTWELL

Adds a well parameter values over a time interval to the objective function. The target values for every report time are given by the interpolation through the user defined function.

OBJTWELL syntax

```
-- in POST section

OBJTWELL
   wellname1 weight1 propertyX1 propertyY1 UserFunc1 FirstFrame1 LastFrame1 /
   wellname2 weight2 propertyX2 propertyY2 UserFunc2 FirstFrame2 LastFrame2 /
   wellname3 weight3 propertyX3 propertyY3 UserFunc3 FirstFrame3 LastFrame3 /
   ...                                                                      /
/

================================================================================

      wellname# - well name;
      weight#   - weight of the residual when it is added to the
                  objective function;
     propertyX# - the property of the UserFunc# argument;
     propertyY# - the property of the UserFunc# value;
      UserFunc# - the user function ID previously specified with FUNCSPEC
                  keyword. The simulation to be calibrated to that function;
    FirstFrame# - first report frame of the fitting interval;
     LastFrame# - last report frame of the fitting interval.
```

OBJVAR

Adds an auxiliary variable at a given report time to the objective function.

OBJVAR syntax

```
-- in POST section

OBJVAR
   weight1 varname1 target1 repframe1  /
   weight2 varname2 target2 repframe2  /
   weight3 varname3 target3 repframe3  /
   ...                                          /
/

================================================================================

      weight#   - weight of the residual when it is added to the
                  objective function;
      varname#  - name of auxiliary variable;
      target#   - target value of the property;
      repframe# - report time frame for which the property is reported.
```

OBJWELL

Adds a well parameter for a given report time to the objective function.

OBJWELL syntax

```
-- in POST section

OBJWELL
   wellname1 weight1 property1 target1 repframe1  /
   wellname2 weight2 property2 target2 repframe2  /
   wellname3 weight3 property3 target3 repframe3  /
   ...                                          /
/

================================================================================

      wellname# - well name;
      weight#   - weight of the residual when it is added to the
                  objective function;
      property# - a reported property for the specified well;
      target#   - target value of the property;
      repframe# - report time frame for which the property is reported.
```

OBJXBLOC

Adds a grid block parameter values over a time interval to the objective function. The target values for every report time are given by the specified data points whereas the simulated values are given by interpolation of the calculated parameters at the reported times. Compare to OBJTBLOC.

OBJXBLOC syntax

```
-- in POST section

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
OBJXBLOC
   i  j  k  gridname  resname weight PropertyX PropertyY PropXMin PropXMax /
   filetype filename /
/

--------------------------------------------------------------------------------

The file "filename" contains the following records:
   ValueX1  ValueY1 /
   ValueX2  ValueY2 /
   ValueX3  ValueY3 /
   ...
/
>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
OBJXBLOC
   i  j  k  gridname  resname weight PropertyX PropertyY PropXMin PropXMax /
   ValueX1  ValueY1 /
   ValueX2  ValueY2 /
   ValueX3  ValueY3 /
   ...
/
>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

================================================================================

      i         - i-index of the grid block;
      j         - j-index of the grid block;
      k         - k-index of the grid block;
      gridname  - grid name (e.g., defined by the CARFIN keyword);
      resname   - reservoir name;
      weight    - weight of the residual when it is added to the
                  objective function;
     PropertyX  - the menmonic of the ValueX property;
```

```
    PropertyY  - the menmonic of the ValueY property;
    PropXMin   - minimum value of PropertyX. The fitting is applied
                 to the data in the interval PropXMin<=PropertyX<=PropXMax.
                 Default values:
                 if PropertyX=TIME    then PropXMin=sim. start time in days
                 if PropertyX=TIMEY   then PropXMin=sim. start time in years
                 if PropertyX=TIMEKY  then PropXMin=sim. start time in ky
                 if PropertyX=TIMESEC then PropXMin=sim. start time in sec
                 Otherwise PropXMin=NaN
    PropXMax   - maximum value of PropertyX. The fitting is applied
                 to the data in the interval PropXMin<=PropertyX<=PropXMax.
                 Default values:
                 if PropertyX=TIME    then PropXMax=sim. end time in days
                 if PropertyX=TIMEY   then PropXMax=sim. end time in years
                 if PropertyX=TIMEKY  then PropXMax=sim. end time in ky
                 if PropertyX=TIMESEC then PropXMax=sim. end time in sec
                 Otherwise PropXMax=NaN
    ValueX#    - the value of the propertyX to be fitted to;
    ValueY#    - the value of the propertyY to be fitted to;
    filetype   = ASCII  - formatted include file option (this option
                 recommended);
               = BINARY - binary include file;
    filename   - name of the include file with the (ValueX,ValueY) data
                 points.
```

OBSPOINT

Defines the observation point for e.g. reporting the gravity changes (see the GRAVIMET keyword) and the ground displacement (see also the OBSSPECM keyword).

OBSPOINT syntax

```
-- in the GRID section

OBSPOINT
   name1  xmap1  ymap1  depth1 /
   name2  xmap2  ymap2  depth2 /
   name3  xmap3  ymap2  depth3 /
   ...                         /
/


================================================================================

    name#   -  name (id) of the observation point (8-byte character).
    xmap#   -  x-coordinate of the observation point in the map axes.
    ymap#   -  y-coordinate of the observation point in the map axes.
    depth#  -  depth of the observation point.
```

OBSSPECM

Defines a network of observation points for e.g. reporting the gravity changes (see the GRAVIMET keyword) and the ground displacement (See also the OBSPOINT keyword). The map of observations for a particular time can be reported using the POSTOBSM keyword.

OBSSPECM syntax

```
-- in the GRID section

OBSSPECM
   template1  srt1   nx1 xmin1 xmax1   ny1 ymin1 ymax1   nz1 zmin1 zmax1 /
   template2  srt2   nx2 xmin2 xmax2   ny2 ymin2 ymax2   nz2 zmin2 zmax2 /
   template3  srt3   nx3 xmin3 xmax3   ny3 ymin3 ymax3   nz3 zmin3 zmax3 /
   ...                                                                   /
/


================================================================================

     template# - the template for the observation point ID.
          srt# - initial integer to add to the template (default: srt#=1)
                 The point IDs are created as follows
                 do k=kmin,kmax; do j=jmin,jmax; do i=imin,imax
                   observation_point_ID=template+srt
                   srt=srt+1
                 enddo;enddo;enddo.
           nx# - number of the observation points along axis X.
   xmin#/xmax# - The boundaries of the region along axis X for placing the
                 network of observation points.
           ny# - number of the observation points along axis Y.
   ymin#/ymax# - The boundaries of the region along axis Y for placing the
                 network of observation points.
           nz# - number of the observation points along axis Z.
   zmin#/zmax# - The boundaries of the region along axis Z for placing the
                 network of observation points.
```

OPAUTNAM

Adds operation for properties of all cells, which character names (if assigned) belong to character template, to the list of automated operations. The list can be cleaned with the OPAUTPAR keyword. The automated operations in the sequence they are uploaded are applied at every report time before the summary file is saved.

OPAUTNAM syntax

```
-- in the INIT or SCHEDULE section

OPAUTNAM
  mnemonic1  mdep1   template1  oper1  par1_1 par2_1 par3_1 /
  mnemonic2  mdep2   template2  oper2  par1_2 par2_2 par3_2 /
  mnemonic3  mdep3   template3  oper3  par1_3 par2_3 par3_3 /
  ...
/

================================================================================

     mnemonic# - mnemonic of the property which is modified (M);
     mdep#     - mnemonic of the dependent property (X);
     template# - character name template.
     oper#     - mnemonic of operation which is applied (see the list);
     par1_#    - dependent scalar value;
     par2_#    - dependent scalar value;
     par3_#    - dependent scalar value.

================================================================================

   The keyword results in the following at every report time before the summary
   report:

 mnem1:=Oper1(mnem1,mdep1,par1_1,par2_1,par3_1) for all cells which character
                               name (if it is assigned) belong to template1;
 mnem2:=Oper2(mnem2,mdep2,par1_2,par2_2,par3_2) for all cells which character
                               name (if it is assigned) belong to template2;
 mnem3:=Oper3(mnem3,mdep3,par1_3,par2_3,par3_3) for all cells which character
                               name (if it is assigned) belong to template3;
 ...
```

List of available operations		
ADD	-	$M := M + par_1$
ADDARR	-	$M := M + X$

List of available operations		
ADDX	-	$M := X + par_1$
COPY	-	$M := X$
COS	-	$M := M + par_1 * \cos(par_2 * X + par_3)$
COSSPEC	-	$M := M + par_1 * \cos(2 * \pi * X/par_2 + par_3)$
CRUSTPER	-	$M := min(10^{par_1 - par_2 * \log_{10}(X/10^3)}, par_3)$
CRUSTPOR	-	$M := max(min(10^{par_1 + par_2 * (X/10^3) + par_3 * (X/10^3)^2}, 1.0), 0.001)$
DISTURB	-	$M := M + par_1 + rand(-1, 1) * par_2$, where $rand(-1, 1)$ is a function generating random numbers in the interval $(-1, 1)$
DIVIARR	-	$M := M/X$
DIVIDE	-	$M := M/par_1$
EQUALS	-	$M := par_1$
EXP	-	$M := M + par_1 \exp(par_2 X) + par_3$
EXP2	-	$M := M + par_3 \exp\left(-\frac{(X-par_2)^2}{par_1^2}\right)$
IFGT	-	$M := X$ if $X > 0$
IFLT	-	$M := X$ if $X < 0$
INV	-	$M := 1/X$
LOGE	-	$M := \ln(X)$
LOG10	-	$M := \log_{10}(X)$
MAXARR	-	$M := max(M, X)$
MAXLIM	-	$M := min\{X, par_1\}$
MAXVALUE	-	$M := min\{M, par_1\}$
MINARR	-	$M := min(M, X)$
MINLIM	-	$M := max\{X, par_1\}$
MINVALUE	-	$M := max\{M, par_1\}$
MOD	-	$M := mod(X, par_1)$; the remainder when X is divided by par_1
MULTA	-	$M := par_1 + X * par_2$
MULTARR	-	$M := M * X$
MULTP	-	$M := par_1 * X^{par_2}$
MULTX	-	$M := X * par_1$
MULTIPLY	-	$M := M * par_1$
NORMDIST	-	$M := M + \frac{par_3}{2\pi par_1^2} \exp\left(-\frac{(X-par_2)^2}{2par_1^2}\right)$
POLY	-	$M := M + par_1 * X^{par_2}$

List of available operations		
POW	-	$M := M^{par_1 * X + par_2}$
SCALE	-	$M := par_1 * X + par_2 * (M - par_1)$
SETINT	-	$M := par_1$ if $par_2 \leq X \leq par_3$
SETEXT	-	$M := par_1$ if $X \leq par_2$ or $X \geq par_3$
SIGN	-	$M := par_1 * sign(X)$
SIN	-	$M := M + par_1 * \sin(par_2 * X + par_3)$
SINSPEC	-	$M := M + par_1 * \sin(2 * \pi * X / par_2 + par_3)$
SLOG	-	$M := 10^{par_1 + X * par_2}$
UF2D	-	$M := UF2D_{par_1}(M, X)$, where $UF2D_{par_1}$ is a function supplied by user
USERFUNC	-	$M := UserFunc_{par_1}(X)$, where $UserFunc_{par_1}$ is a function supplied by user

OPAUTO

Adds operation for properties in a box of cells to the list of automated operations. The list can be cleaned with the OPAUTPAR keyword. The automated operations in the sequence they are uploaded are applied at every report time before the summary file is saved.

OPAUTO syntax

```
-- in the INIT or SCHEDULE section

OPAUTO
  mnem1 mdep1 oper1 par1_1 par2_1 par3_1  imin1 imax1 jmin1 jmax1 kmin1 kmax1 /
  mnem2 mdep2 oper2 par1_2 par2_2 par3_2  imin2 imax2 jmin2 jmax2 kmin2 kmax2 /
  mnem3 mdep3 oper3 par1_3 par2_3 par3_3  imin3 imax3 jmin3 jmax3 kmin3 kmax3 /
   ...
/

================================================================================

   mnem#       - mnemonic of the property which is modified (M).
   mdep#       - mnemonic of the dependent property (X).
   oper#       - mnemonic of operation which is applied (see the list).
   par1_#      - dependent scalar value.
   par2_#      - dependent scalar value.
   par3_#      - dependent scalar value.
   imin#/imax# - the boundaries of the input box along i-indexation axis.
                 By default these values are equal to the arguments 1 and 2
                 of the keyword BOX.
   jmin#/kmax# - the boundaries of the input box along j-indexation axis.
                 By default these values are equal to the arguments 3 and 4
                 of the keyword BOX.
   kmin#/kmax# - the boundaries of the input box along k-indexation axis.
                 By default these values are equal to the arguments 5 and 6
                 of the keyword BOX.

================================================================================

   The keyword results in the following at every report time before the summary
   report:

       mnem1:=Oper1(mnem1,mdep1,par1_1,par2_1,par3_1)
                          in the box [imin1,imax1]*[jmin1,jmax1]*[kmin1,kmax1]
       mnem2:=Oper2(mnem2,mdep2,par1_2,par2_2,par3_2)
                          in the box [imin2,imax2]*[jmin2,jmax2]*[kmin2,kmax2]
       mnem3:=Oper3(mnem3,mdep3,par1_3,par2_3,par3_3)
                          in the box [imin3,imax3]*[jmin3,jmax3]*[kmin3,kmax3]
```

39 ...

List of available operations		
ADD	-	$M := M + par_1$
ADDARR	-	$M := M + X$
ADDX	-	$M := X + par_1$
COPY	-	$M := X$
COS	-	$M := M + par_1 * \cos(par_2 * X + par_3)$
COSSPEC	-	$M := M + par_1 * \cos(2 * \pi * X/par_2 + par_3)$
CRUSTPER	-	$M := min(10^{par_1 - par_2 * \log_{10}(X/10^3)}, par_3)$
CRUSTPOR	-	$M := max(min(10^{par_1 + par_2 * (X/10^3) + par_3 * (X/10^3)^2}, 1.0), 0.001)$
DISTURB	-	$M := M + par_1 + rand(-1, 1) * par_2$, where $rand(-1, 1)$ is a function generating random numbers in the interval $(-1, 1)$
DIVIARR	-	$M := M/X$
DIVIDE	-	$M := M/par_1$
EQUALS	-	$M := par_1$
EXP	-	$M := M + par_1 \exp(par_2 X) + par_3$
EXP2	-	$M := M + par_3 \exp\left(-\frac{(X - par_2)^2}{par_1^2}\right)$
IFGT	-	$M := X$ if $X > 0$
IFLT	-	$M := X$ if $X < 0$
INV	-	$M := 1/X$
LOGE	-	$M := \ln(X)$
LOG10	-	$M := \log_{10}(X)$
MAXARR	-	$M := max(M, X)$
MAXLIM	-	$M := min\{X, par_1\}$
MAXVALUE	-	$M := min\{M, par_1\}$
MINARR	-	$M := min(M, X)$
MINLIM	-	$M := max\{X, par_1\}$
MINVALUE	-	$M := max\{M, par_1\}$
MOD	-	$M := mod(X, par_1)$; the remainder when X is divided by par_1
MULTA	-	$M := par_1 + X * par_2$
MULTARR	-	$M := M * X$
MULTP	-	$M := par_1 * X^{par_2}$
MULTX	-	$M := X * par_1$

List of available operations		
MULTIPLY	-	$M := M * par_1$
NORMDIST	-	$M := M + \frac{par_3}{2\pi par_1^2} \exp\left(-\frac{(X-par_2)^2}{2par_1^2}\right)$
POLY	-	$M := M + par_1 * X^{par_2}$
POW	-	$M := M^{par_1 * X + par_2}$
SCALE	-	$M := par_1 * X + par_2 * (M - par_1)$
SETINT	-	$M := par_1$ if $par_2 \leq X \leq par_3$
SETEXT	-	$M := par_1$ if $X \leq par_2$ or $X \geq par_3$
SIGN	-	$M := par_1 * sign(X)$
SIN	-	$M := M + par_1 * \sin(par_2 * X + par_3)$
SINSPEC	-	$M := M + par_1 * \sin(2 * \pi * X / par_2 + par_3)$
SLOG	-	$M := 10^{par_1 + X * par_2}$
UF2D	-	$M := UF2D_{par_1}(M, X)$, where $UF2D_{par_1}$ is a function supplied by user
USERFUNC	-	$M := UserFunc_{par_1}(X)$, where $UserFunc_{par_1}$ is a function supplied by user

OPAUTPAR

Controls application of the automated operations with arrays. See the OPAUTO, OPAUTREG and OPAUTNAM keywords. The automated operations in the sequence they are uploaded are applied at every report time before the summary file is saved.

OPAUTPAR syntax

```
-- in the INIT or SCHEDULE section

OPAUTPAR
  par1 par2 par3 ... /

================================================================================

   par#       - parameter for automated operations with arrays.
                The possible values are:
                  - ON    - the automated operations are enabled;
                  - OFF   - the automated operations are disabled;
                  - FREE  - free the list of automated operations.


```

OPAUTREG

Adds operation for properties in a region to the list of automated operations. The list can be cleaned with the OPAUTPAR keyword. The automated operations in the sequence they are uploaded are applied at every report time before the summary file is saved.

OPAUTREG syntax

```
-- in the INIT and SCHEDULE sections

OPAUTREG
  mnemonic1  mdep1   region1  regionID1  oper1  par1_1 par2_1 par3_1 /
  mnemonic2  mdep2   region2  regionID2  oper2  par1_2 par2_2 par3_2 /
  mnemonic3  mdep3   region3  regionID3  oper3  par1_3 par2_3 par3_3 /
  ...
/

================================================================================

     mnemonic# - mnemonic of the property which is modified (M);
     mdep#     - mnemonic of the dependent property (X);
     region#   - mnemonic of the region in which the property is modified;
     regionID# - region number;
     oper#     - mnemonic of operation which is applied (see the list);
     par1_#    - dependent scalar value;
     par2_#    - dependent scalar value;
     par3_#    - dependent scalar value.

================================================================================

   The keyword results in the following at every report time before the summary
   report:

 mnem1:=Oper1(mnem1,mdep1,par1_1,par2_1,par3_1) in the region region1=regionID1
 mnem2:=Oper2(mnem2,mdep2,par1_2,par2_2,par3_2) in the region region2=regionID2
 mnem3:=Oper3(mnem3,mdep3,par1_3,par2_3,par3_3) in the region region3=regionID3
 ...
```

List of available operations		
ADD	-	$M := M + par_1$
ADDARR	-	$M := M + X$
ADDX	-	$M := X + par_1$
COPY	-	$M := X$
COS	-	$M := M + par_1 * \cos(par_2 * X + par_3)$

List of available operations		
COSSPEC	-	$M := M + par_1 * \cos(2 * \pi * X/par_2 + par_3)$
CRUSTPER	-	$M := min(10^{par_1 - par_2 * \log_{10}(X/10^3)}, par_3)$
CRUSTPOR	-	$M := max(min(10^{par_1 + par_2 * (X/10^3) + par_3 * (X/10^3)^2}, 1.0), 0.001)$
DISTURB	-	$M := M + par_1 + rand(-1, 1) * par_2$, where $rand(-1, 1)$ is a function generating random numbers in the interval $(-1, 1)$
DIVIARR	-	$M := M/X$
DIVIDE	-	$M := M/par_1$
EQUALS	-	$M := par_1$
EXP	-	$M := M + par_1 \exp(par_2 X) + par_3$
EXP2	-	$M := M + par_3 \exp\left(-\frac{(X-par_2)^2}{par_1^2}\right)$
IFGT	-	$M := X$ if $X > 0$
IFLT	-	$M := X$ if $X < 0$
INV	-	$M := 1/X$
LOGE	-	$M := \ln(X)$
LOG10	-	$M := \log_{10}(X)$
MAXARR	-	$M := max(M, X)$
MAXLIM	-	$M := min\{X, par_1\}$
MAXVALUE	-	$M := min\{M, par_1\}$
MINARR	-	$M := min(M, X)$
MINLIM	-	$M := max\{X, par_1\}$
MINVALUE	-	$M := max\{M, par_1\}$
MOD	-	$M := mod(X, par_1)$; the remainder when X is divided by par_1
MULTA	-	$M := par_1 + X * par_2$
MULTARR	-	$M := M * X$
MULTP	-	$M := par_1 * X^{par_2}$
MULTX	-	$M := X * par_1$
MULTIPLY	-	$M := M * par_1$
NORMDIST	-	$M := M + \frac{par_3}{2\pi par_1^2} \exp\left(-\frac{(X-par_2)^2}{2par_1^2}\right)$
POLY	-	$M := M + par_1 * X^{par_2}$
POW	-	$M := M^{par_1 * X + par_2}$
SCALE	-	$M := par_1 * X + par_2 * (M - par_1)$
SETINT	-	$M := par_1$ if $par_2 \leq X \leq par_3$

List of available operations		
SETEXT	-	$M := par_1$ if $X \leq par_2$ or $X \geq par_3$
SIGN	-	$M := par_1 * sign(X)$
SIN	-	$M := M + par_1 * \sin(par_2 * X + par_3)$
SINSPEC	-	$M := M + par_1 * \sin(2 * \pi * X/par_2 + par_3)$
SLOG	-	$M := 10^{par_1 + X * par_2}$
UF2D	-	$M := UF2D_{par_1}(M, X)$, where $UF2D_{par_1}$ is a function supplied by user
USERFUNC	-	$M := UserFunc_{par_1}(X)$, where $UserFunc_{par_1}$ is a function supplied by user

OPERANAM

Applies an operation to a property for all cells which character names (if assigned) belongs to character template.

OPERANAM syntax

```
-- in all sections except RUNSPEC and POST

OPERANAM
  mnemonic1  mdep1   template1  oper1  par1_1 par2_1 par3_1 /
  mnemonic2  mdep2   template2  oper2  par1_2 par2_2 par3_2 /
  mnemonic3  mdep3   template3  oper3  par1_3 par2_3 par3_3 /
  ...
/

================================================================================

     mnemonic# - mnemonic of the property which is modified (M);
     mdep#     - mnemonic of the dependent property (X);
     template# - character name template.
     oper#     - mnemonic of operation which is applied (see the list);
     par1_#    - dependent scalar value;
     par2_#    - dependent scalar value;
     par3_#    - dependent scalar value.

================================================================================

   The keyword results in the following:

 mnem1:=Oper1(mnem1,mdep1,par1_1,par2_1,par3_1) for all cells which character
                              name (if it is assigned) belong to template1;
 mnem2:=Oper2(mnem2,mdep2,par1_2,par2_2,par3_2) for all cells which character
                              name (if it is assigned) belong to template2;
 mnem3:=Oper3(mnem3,mdep3,par1_3,par2_3,par3_3) for all cells which character
                              name (if it is assigned) belong to template3;
 ...
```

List of available operations		
ADD	-	$M := M + par_1$
ADDARR	-	$M := M + X$
ADDX	-	$M := X + par_1$
COPY	-	$M := X$
COS	-	$M := M + par_1 * \cos(par_2 * X + par_3)$
COSSPEC	-	$M := M + par_1 * \cos(2 * \pi * X/par_2 + par_3)$

List of available operations		
CRUSTPER	-	$M := min(10^{par_1 - par_2 * \log_{10}(X/10^3)}, par_3)$
CRUSTPOR	-	$M := max(min(10^{par_1 + par_2 * (X/10^3) + par_3 * (X/10^3)^2}, 1.0), 0.001)$
DISTURB	-	$M := M + par_1 + rand(-1, 1) * par_2$, where $rand(-1, 1)$ is a function generating random numbers in the interval $(-1, 1)$
DIVIARR	-	$M := M/X$
DIVIDE	-	$M := M/par_1$
EQUALS	-	$M := par_1$
EXP	-	$M := M + par_1 \exp(par_2 X) + par_3$
EXP2	-	$M := M + par_3 \exp\left(-\frac{(X-par_2)^2}{par_1^2}\right)$
IFGT	-	$M := X$ if $X > 0$
IFLT	-	$M := X$ if $X < 0$
INV	-	$M := 1/X$
LOGE	-	$M := \ln(X)$
LOG10	-	$M := \log_{10}(X)$
MAXARR	-	$M := max(M, X)$
MAXLIM	-	$M := min\{X, par_1\}$
MAXVALUE	-	$M := min\{M, par_1\}$
MINARR	-	$M := min(M, X)$
MINLIM	-	$M := max\{X, par_1\}$
MINVALUE	-	$M := max\{M, par_1\}$
MOD	-	$M := mod(X, par_1)$; the remainder when X is divided by par_1
MULTA	-	$M := par_1 + X * par_2$
MULTARR	-	$M := M * X$
MULTP	-	$M := par_1 * X^{par_2}$
MULTX	-	$M := X * par_1$
MULTIPLY	-	$M := M * par_1$
NORMDIST	-	$M := M + \frac{par_3}{2\pi par_1^2} \exp\left(-\frac{(X-par_2)^2}{2par_1^2}\right)$
POLY	-	$M := M + par_1 * X^{par_2}$
POW	-	$M := M^{par_1 * X + par_2}$
SCALE	-	$M := par_1 * X + par_2 * (M - par_1)$
SETINT	-	$M := par_1$ if $par_2 \leq X \leq par_3$
SETEXT	-	$M := par_1$ if $X \leq par_2$ or $X \geq par_3$

List of available operations		
SIGN	-	$M := par_1 * sign(X)$
SIN	-	$M := M + par_1 * \sin(par_2 * X + par_3)$
SINSPEC	-	$M := M + par_1 * \sin(2 * \pi * X / par_2 + par_3)$
SLOG	-	$M := 10^{par_1 + X * par_2}$
UF2D	-	$M := UF2D_{par_1}(M, X)$, where $UF2D_{par_1}$ is a function supplied by user
USERFUNC	-	$M := UserFunc_{par_1}(X)$, where $UserFunc_{par_1}$ is a function supplied by user

OPERAREG

Applies an operation to a property in a region.

OPERAREG syntax

```
-- in all sections except RUNSPEC and POST

OPERAREG
  mnemonic1  mdep1   region1  regionID1  oper1  par1_1 par2_1 par3_1 /
  mnemonic2  mdep2   region2  regionID2  oper2  par1_2 par2_2 par3_2 /
  mnemonic3  mdep3   region3  regionID3  oper3  par1_3 par2_3 par3_3 /
  ...
/

================================================================================

     mnemonic# - mnemonic of the property which is modified (M);
     mdep#     - mnemonic of the dependent property (X);
     region#   - mnemonic of the region in which the property is modified;
     regionID# - region number;
     oper#     - mnemonic of operation which is applied (see the list);
     par1_#    - dependent scalar value;
     par2_#    - dependent scalar value;
     par3_#    - dependent scalar value.

================================================================================

   The keyword results in the following:

 mnem1:=Oper1(mnem1,mdep1,par1_1,par2_1,par3_1) in the region region1=regionID1
 mnem2:=Oper2(mnem2,mdep2,par1_2,par2_2,par3_2) in the region region2=regionID2
 mnem3:=Oper3(mnem3,mdep3,par1_3,par2_3,par3_3) in the region region3=regionID3
 ...
```

List of available operations		
ADD	-	$M := M + par_1$
ADDARR	-	$M := M + X$
ADDX	-	$M := X + par_1$
COPY	-	$M := X$
COS	-	$M := M + par_1 * \cos(par_2 * X + par_3)$
COSSPEC	-	$M := M + par_1 * \cos(2 * \pi * X/par_2 + par_3)$
CRUSTPER	-	$M := min(10^{par_1 - par_2 * \log_{10}(X/10^3)}, par_3)$
CRUSTPOR	-	$M := max(min(10^{par_1 + par_2 * (X/10^3) + par_3 * (X/10^3)^2}, 1.0), 0.001)$

List of available operations		
DISTURB	-	$M := M + par_1 + rand(-1,1) * par_2$, where $rand(-1,1)$ is a function generating random numbers in the interval $(-1,1)$
DIVIARR	-	$M := M/X$
DIVIDE	-	$M := M/par_1$
EQUALS	-	$M := par_1$
EXP	-	$M := M + par_1 \exp(par_2 X) + par_3$
EXP2	-	$M := M + par_3 \exp\left(-\frac{(X-par_2)^2}{par_1^2}\right)$
IFGT	-	$M := X$ if $X > 0$
IFLT	-	$M := X$ if $X < 0$
INV	-	$M := 1/X$
LOGE	-	$M := \ln(X)$
LOG10	-	$M := \log_{10}(X)$
MAXARR	-	$M := max(M, X)$
MAXLIM	-	$M := min\{X, par_1\}$
MAXVALUE	-	$M := min\{M, par_1\}$
MINARR	-	$M := min(M, X)$
MINLIM	-	$M := max\{X, par_1\}$
MINVALUE	-	$M := max\{M, par_1\}$
MOD	-	$M := mod(X, par_1)$; the remainder when X is divided by par_1
MULTA	-	$M := par_1 + X * par_2$
MULTARR	-	$M := M * X$
MULTP	-	$M := par_1 * X^{par_2}$
MULTX	-	$M := X * par_1$
MULTIPLY	-	$M := M * par_1$
NORMDIST	-	$M := M + \frac{par_3}{2\pi par_1^2} \exp\left(-\frac{(X-par_2)^2}{2par_1^2}\right)$
POLY	-	$M := M + par_1 * X^{par_2}$
POW	-	$M := M^{par_1 * X + par_2}$
SCALE	-	$M := par_1 * X + par_2 * (M - par_1)$
SETINT	-	$M := par_1$ if $par_2 \le X \le par_3$
SETEXT	-	$M := par_1$ if $X \le par_2$ or $X \ge par_3$
SIGN	-	$M := par_1 * sign(X)$
SIN	-	$M := M + par_1 * \sin(par_2 * X + par_3)$

List of available operations		
SINSPEC	-	$M := M + par_1 * \sin(2 * \pi * X/par_2 + par_3)$
SLOG	-	$M := 10^{par_1 + X * par_2}$
UF2D	-	$M := UF2D_{par_1}(M, X)$, where $UF2D_{par_1}$ is a function supplied by user
USERFUNC	-	$M := UserFunc_{par_1}(X)$, where $UserFunc_{par_1}$ is a function supplied by user

OPERATE

Applies an operation to a property in a box of cells.

OPERATE syntax

```
-- in all sections except RUNSPEC and POST

OPERATE
  mnem1 mdep1 oper1 par1_1 par2_1 par3_1  imin1 imax1 jmin1 jmax1 kmin1 kmax1 /
  mnem2 mdep2 oper2 par1_2 par2_2 par3_2  imin2 imax2 jmin2 jmax2 kmin2 kmax2 /
  mnem3 mdep3 oper3 par1_3 par2_3 par3_3  imin3 imax3 jmin3 jmax3 kmin3 kmax3 /
   ...
/

================================================================================

   mnem#       - mnemonic of the property which is modified (M).
   mdep#       - mnemonic of the dependent property (X).
   oper#       - mnemonic of operation which is applied (see the list).
   par1_#      - dependent scalar value.
   par2_#      - dependent scalar value.
   par3_#      - dependent scalar value.
   imin#/imax# - the boundaries of the input box along i-indexation axis.
                 By default these values are equal to the arguments 1 and 2
                 of the keyword BOX.
   jmin#/kmax# - the boundaries of the input box along j-indexation axis.
                 By default these values are equal to the arguments 3 and 4
                 of the keyword BOX.
   kmin#/kmax# - the boundaries of the input box along k-indexation axis.
                 By default these values are equal to the arguments 5 and 6
                 of the keyword BOX.

================================================================================

   The keyword results in the following:

       mnem1:=Oper1(mnem1,mdep1,par1_1,par2_1,par3_1)
                          in the box [imin1,imax1]*[jmin1,jmax1]*[kmin1,kmax1]
       mnem2:=Oper2(mnem2,mdep2,par1_2,par2_2,par3_2)
                          in the box [imin2,imax2]*[jmin2,jmax2]*[kmin2,kmax2]
       mnem3:=Oper3(mnem3,mdep3,par1_3,par2_3,par3_3)
                          in the box [imin3,imax3]*[jmin3,jmax3]*[kmin3,kmax3]
       ...
```

List of available operations		
ADD	-	$M := M + par_1$
ADDARR	-	$M := M + X$

List of available operations		
ADDX	-	$M := X + par_1$
COPY	-	$M := X$
COS	-	$M := M + par_1 * \cos(par_2 * X + par_3)$
COSSPEC	-	$M := M + par_1 * \cos(2 * \pi * X/par_2 + par_3)$
CRUSTPER	-	$M := min(10^{par_1 - par_2 * \log_{10}(X/10^3)}, par_3)$
CRUSTPOR	-	$M := max(min(10^{par_1 + par_2 * (X/10^3) + par_3 * (X/10^3)^2}, 1.0), 0.001)$
DISTURB	-	$M := M + par_1 + rand(-1, 1) * par_2$, where $rand(-1, 1)$ is a function generating random numbers in the interval $(-1, 1)$
DIVIARR	-	$M := M/X$
DIVIDE	-	$M := M/par_1$
EQUALS	-	$M := par_1$
EXP	-	$M := M + par_1 \exp(par_2 X) + par_3$
EXP2	-	$M := M + par_3 \exp\left(-\frac{(X-par_2)^2}{par_1^2}\right)$
IFGT	-	$M := X$ if $X > 0$
IFLT	-	$M := X$ if $X < 0$
INV	-	$M := 1/X$
LOGE	-	$M := \ln(X)$
LOG10	-	$M := \log_{10}(X)$
MAXARR	-	$M := max(M, X)$
MAXLIM	-	$M := min\{X, par_1\}$
MAXVALUE	-	$M := min\{M, par_1\}$
MINARR	-	$M := min(M, X)$
MINLIM	-	$M := max\{X, par_1\}$
MINVALUE	-	$M := max\{M, par_1\}$
MOD	-	$M := mod(X, par_1)$; the remainder when X is divided by par_1
MULTA	-	$M := par_1 + X * par_2$
MULTARR	-	$M := M * X$
MULTP	-	$M := par_1 * X^{par_2}$
MULTX	-	$M := X * par_1$
MULTIPLY	-	$M := M * par_1$
NORMDIST	-	$M := M + \frac{par_3}{2\pi par_1^2} \exp\left(-\frac{(X-par_2)^2}{2par_1^2}\right)$
POLY	-	$M := M + par_1 * X^{par_2}$

List of available operations		
POW	-	$M := M^{par_1 * X + par_2}$
SCALE	-	$M := par_1 * X + par_2 * (M - par_1)$
SETINT	-	$M := par_1$ if $par_2 \leq X \leq par_3$
SETEXT	-	$M := par_1$ if $X \leq par_2$ or $X \geq par_3$
SIGN	-	$M := par_1 * sign(X)$
SIN	-	$M := M + par_1 * \sin(par_2 * X + par_3)$
SINSPEC	-	$M := M + par_1 * \sin(2 * \pi * X / par_2 + par_3)$
SLOG	-	$M := 10^{par_1 + X * par_2}$
UF2D	-	$M := UF2D_{par_1}(M, X)$, where $UF2D_{par_1}$ is a function supplied by user
USERFUNC	-	$M := UserFunc_{par_1}(X)$, where $UserFunc_{par_1}$ is a function supplied by user

OPTIONS

Sets the simulator options.

OPTIONS syntax

```
-- in all sections

OPTIONS
  opt1 opt2 opt3 ... /

================================================================================

     opt# - if 1 the corresponding option is enabled, if 0 the option is
            disabled.
```

OPVAR

Applies instantaneous operations to the auxiliary variables defined by AUXVAR keyword.

OPVAR syntax

```
-- in any section except RUNSPEC

OPVAR
  varmod1  vardep1  oper1 par1_1 par2_1 par3_1 /
  varmod2  vardep2  oper2 par1_2 par2_2 par3_2 /
  varmod3  vardep3  oper3 par1_3 par2_3 par3_3 /
   ...
/

================================================================================

   varmod#     - auxiliary variable which is modified (M).
   vardep#     - mnemonic of the dependent variable (X).
   oper#       - mnemonic of operation which is applied (see the list).
   par1_#      - dependent scalar value.
   par2_#      - dependent scalar value.
   par3_#      - dependent scalar value.

================================================================================

   The keyword results in the following instantaneous operations:

       varmod1:=Oper1(varmod1,vardep1,par1_1,par2_1,par3_1)
       varmod2:=Oper2(varmod2,vardep2,par1_2,par2_2,par3_2)
       varmod3:=Oper3(varmod3,vardep3,par1_3,par2_3,par3_3)
       ...
```

List of available operations		
ADD	-	$M := M + par_1$
ADDARR	-	$M := M + X$
ADDX	-	$M := X + par_1$
COPY	-	$M := X$
COS	-	$M := M + par_1 * \cos(par_2 * X + par_3)$
COSSPEC	-	$M := M + par_1 * \cos(2 * \pi * X/par_2 + par_3)$
CRUSTPER	-	$M := min(10^{par_1 - par_2 * \log_{10}(X/10^3)}, par_3)$
CRUSTPOR	-	$M := max(min(10^{par_1 + par_2 * (X/10^3) + par_3 * (X/10^3)^2}, 1.0), 0.001)$
DISTURB	-	$M := M + par_1 + rand(-1, 1) * par_2$, where $rand(-1, 1)$ is a function generating random numbers in the interval $(-1, 1)$

List of available operations		
DIVIARR	-	$M := M/X$
DIVIDE	-	$M := M/par_1$
EQUALS	-	$M := par_1$
EXP	-	$M := M + par_1 \exp(par_2 X) + par_3$
EXP2	-	$M := M + par_3 \exp\left(-\frac{(X-par_2)^2}{par_1^2}\right)$
IFGT	-	$M := X$ if $X > 0$
IFLT	-	$M := X$ if $X < 0$
INV	-	$M := 1/X$
LOGE	-	$M := \ln(X)$
LOG10	-	$M := \log_{10}(X)$
MAXARR	-	$M := max(M, X)$
MAXLIM	-	$M := min\{X, par_1\}$
MAXVALUE	-	$M := min\{M, par_1\}$
MINARR	-	$M := min(M, X)$
MINLIM	-	$M := max\{X, par_1\}$
MINVALUE	-	$M := max\{M, par_1\}$
MOD	-	$M := mod(X, par_1)$; the remainder when X is divided by par_1
MULTA	-	$M := par_1 + X * par_2$
MULTARR	-	$M := M * X$
MULTP	-	$M := par_1 * X^{par_2}$
MULTX	-	$M := X * par_1$
MULTIPLY	-	$M := M * par_1$
NORMDIST	-	$M := M + \frac{par_3}{2\pi par_1^2} \exp\left(-\frac{(X-par_2)^2}{2par_1^2}\right)$
POLY	-	$M := M + par_1 * X^{par_2}$
POW	-	$M := M^{par_1 * X + par_2}$
SCALE	-	$M := par_1 * X + par_2 * (M - par_1)$
SETINT	-	$M := par_1$ if $par_2 \leq X \leq par_3$
SETEXT	-	$M := par_1$ if $X \leq par_2$ or $X \geq par_3$
SIGN	-	$M := par_1 * sign(X)$
SIN	-	$M := M + par_1 * \sin(par_2 * X + par_3)$
SINSPEC	-	$M := M + par_1 * \sin(2 * \pi * X/par_2 + par_3)$
SLOG	-	$M := 10^{par_1 + X * par_2}$

List of available operations		
UF2D	-	$M := UF2D_{par_1}(M, X)$, where $UF2D_{par_1}$ is a function supplied by user
USERFUNC	-	$M := UserFunc_{par_1}(X)$, where $UserFunc_{par_1}$ is a function supplied by user

OPVARAUT

Performs operations with the auxiliary variables defined by AUXVAR keyword. The operations defined by OPAUTO, OPAUTREG, OPAUTNAM, OPVARAUT, CPVARAUT, and similar keyword for automated operations are applied in the sequence they appear in the RUN-file. The list of operations can be erased by OPAUTPAR keyword.

OPVARAUT syntax

```
-- in INIT or SCHEDULE section

OPVARAUT
  varmod1  vardep1  oper1 par1_1 par2_1 par3_1 /
  varmod2  vardep2  oper2 par1_2 par2_2 par3_2 /
  varmod3  vardep3  oper3 par1_3 par2_3 par3_3 /
   ...
/

================================================================================

   varmod#     - auxiliary variable which is modified (M).
   vardep#     - mnemonic of the dependent variable (X).
   oper#       - mnemonic of operation which is applied (see the list).
   par1_#      - dependent scalar value.
   par2_#      - dependent scalar value.
   par3_#      - dependent scalar value.

================================================================================

   The keyword results in the following operations:

       varmod1:=Oper1(varmod1,vardep1,par1_1,par2_1,par3_1)
       varmod2:=Oper2(varmod2,vardep2,par1_2,par2_2,par3_2)
       varmod3:=Oper3(varmod3,vardep3,par1_3,par2_3,par3_3)
       ...
   automatically applied after every time step.
```

List of available operations		
ADD	-	$M := M + par_1$
ADDARR	-	$M := M + X$
ADDX	-	$M := X + par_1$
COPY	-	$M := X$
COS	-	$M := M + par_1 * \cos(par_2 * X + par_3)$

List of available operations		
COSSPEC	-	$M := M + par_1 * \cos(2 * \pi * X/par_2 + par_3)$
CRUSTPER	-	$M := min(10^{par_1 - par_2 * \log_{10}(X/10^3)}, par_3)$
CRUSTPOR	-	$M := max(min(10^{par_1 + par_2 * (X/10^3) + par_3 * (X/10^3)^2}, 1.0), 0.001)$
DISTURB	-	$M := M + par_1 + rand(-1, 1) * par_2$, where $rand(-1, 1)$ is a function generating random numbers in the interval $(-1, 1)$
DIVIARR	-	$M := M/X$
DIVIDE	-	$M := M/par_1$
EQUALS	-	$M := par_1$
EXP	-	$M := M + par_1 \exp(par_2 X) + par_3$
EXP2	-	$M := M + par_3 \exp\left(-\frac{(X-par_2)^2}{par_1^2}\right)$
IFGT	-	$M := X$ if $X > 0$
IFLT	-	$M := X$ if $X < 0$
INV	-	$M := 1/X$
LOGE	-	$M := \ln(X)$
LOG10	-	$M := \log_{10}(X)$
MAXARR	-	$M := max(M, X)$
MAXLIM	-	$M := min\{X, par_1\}$
MAXVALUE	-	$M := min\{M, par_1\}$
MINARR	-	$M := min(M, X)$
MINLIM	-	$M := max\{X, par_1\}$
MINVALUE	-	$M := max\{M, par_1\}$
MOD	-	$M := mod(X, par_1)$; the remainder when X is divided by par_1
MULTA	-	$M := par_1 + X * par_2$
MULTARR	-	$M := M * X$
MULTP	-	$M := par_1 * X^{par_2}$
MULTX	-	$M := X * par_1$
MULTIPLY	-	$M := M * par_1$
NORMDIST	-	$M := M + \frac{par_3}{2\pi par_1^2} \exp\left(-\frac{(X-par_2)^2}{2par_1^2}\right)$
POLY	-	$M := M + par_1 * X^{par_2}$
POW	-	$M := M^{par_1 * X + par_2}$
SCALE	-	$M := par_1 * X + par_2 * (M - par_1)$
SETINT	-	$M := par_1$ if $par_2 \leq X \leq par_3$

List of available operations		
SETEXT	-	$M := par_1$ if $X \leq par_2$ or $X \geq par_3$
SIGN	-	$M := par_1 * sign(X)$
SIN	-	$M := M + par_1 * \sin(par_2 * X + par_3)$
SINSPEC	-	$M := M + par_1 * \sin(2 * \pi * X/par_2 + par_3)$
SLOG	-	$M := 10^{par_1 + X * par_2}$
UF2D	-	$M := UF2D_{par_1}(M, X)$, where $UF2D_{par_1}$ is a function supplied by user
USERFUNC	-	$M := UserFunc_{par_1}(X)$, where $UserFunc_{par_1}$ is a function supplied by user

PARTI

Defines the grid partition parameters along i-index direction. This keyword must be positioned immediately after the PARTIT keyword.

PARTI syntax

```
-- within MAKE-ENDMAKE brackets

PARTI
   pi1 pi2 pi3 ... piN /

================================================================================

   pi# - the number of grid blocks along i-index direction belonging to the
         grid partition region;
   N   - the number of grid partition regions along i-index direction (defined
         by the 1st argument of the PARTIT keyword).
```

PARTIT

Applies a user specified grid partition of a Cartesian type. See also the PARTI, PARTJ and PARTK keywords.

PARTIT syntax

```
-- within MAKE-ENDMAKE brackets

PARTIT
   ni  nj  nk  nsrt /

================================================================================

    ni   - number of the grid partition regions along i-index direction;
    nj   - number of the grid partition regions along j-index direction;
    nk   - number of the grid partition regions along k-index direction;
    nsrt - the initial number from which the partition regions are numbered.
           The MPINUM property is assigned to the regions with i-index cycling
           the fastest following by the j- and k-indexes.

```

PARTJ

Defines the grid partition parameters along j-index direction. This keyword must be positioned immediately after the PARTIT keyword.

PARTJ syntax

```
-- within MAKE-ENDMAKE brackets

PARTJ
   pj1 pj2 pj3 ... pjN /

================================================================================

   pj# - the number of grid blocks along j-index direction belonging to the
         grid partition region;
   N   - the number of grid partition regions along j-index direction (defined
         by the 2nd argument of the PARTIT keyword).
```

PARTK

Defines the grid partition parameters along k-index direction. This keyword must be positioned immediately after the PARTIT keyword.

PARTK syntax

```
-- within MAKE-ENDMAKE brackets

PARTK
   pk1 pk2 pk3 ... pkN /

================================================================================

   pk# - the number of grid blocks along k-index direction belonging to the
         grid partition region;
   N   - the number of grid partition regions along k-index direction (defined
         by the 3rd argument of the PARTIT keyword).
```

PBLKROCK

Specifies the PBLKROCK property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

PBLKROCK syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

PBLKROCK
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

PBLKROCK
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

PBLKROCK
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

PERMI

Specifies the PERMI property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

PERMI syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

PERMI
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

PERMI
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

PERMI
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

PERMIFR

Specifies the PERMIFR property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

PERMIFR syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

PERMIFR
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

PERMIFR
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

PERMIFR
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

PERMJ

Specifies the PERMJ property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

PERMJ syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

PERMJ
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

PERMJ
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

PERMJ
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

PERMJFR

Specifies the PERMJFR property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

PERMJFR syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

PERMJFR
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

PERMJFR
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

PERMJFR
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

PERMK

Specifies the PERMK property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

PERMK syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

             -- if the LOADING switch is in IJK position --

PERMK
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

PERMK
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

PERMK
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

PERMKFR

Specifies the PERMKFR property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

PERMKFR syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

PERMKFR
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

PERMKFR
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

PERMKFR
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

PERMX

Specifies the PERMX property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

PERMX syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

PERMX
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

PERMX
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

PERMX
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

PERMXFR

Specifies the PERMXFR property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

PERMXFR syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

             -- if the LOADING switch is in IJK position --

PERMXFR
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

PERMXFR
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

PERMXFR
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

PERMY

Specifies the PERMY property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

PERMY syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

PERMY
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

PERMY
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

PERMY
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

PERMYFR

Specifies the PERMYFR property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

PERMYFR syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

PERMYFR
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

PERMYFR
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

PERMYFR
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

PERMZ

Specifies the PERMZ property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

PERMZ syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

PERMZ
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

PERMZ
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

PERMZ
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

PERMZFR

Specifies the PERMZFR property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

PERMZFR syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

PERMZFR
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

PERMZFR
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

PERMZFR
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

PINCH

Defines the pinch-out layers parameters and options.

PINCH syntax

```
in GRID section

PINCH
  dz /

================================================================================

    dz - pinch-out treshold thickness.
```

POLYLINE

Defines a polyline that can be used in the arithmetic operations with POLYGON or POLYLINE (see ARITH and section 1.5). The syntax of that keyword is influenced by the POLYOPT keyword.

POLYLINE syntax

```
-- in any section except RUNSPEC or POST

POLYLINE
  x1  y1  z1 /
  x2  y2  z2 /
  x3  y3  z3 /
  ...
/

================================================================================

    x#, y#, z#    - the coourdinates of a point on the polyline in 3D space.
```

POLYOPT

Changes the number of columns required in POLYLINE.

POLYOPT syntax

```
-- in any section except RUNSPEC or POST

POLYOPT
  ncol  /

================================================================================

     ncol    - the number of columns readed by the POLYLINE keyword. Must be an
               integer between 1 and 3. Default value: 3.
```

PORO

Specifies the PORO property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

PORO syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

PORO
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

PORO
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

PORO
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

POROWAVE

Enables the option for modelling the porosity waves.

POROWAVE syntax

```
-- in RUNSPEC section

POROWAVE
```

POST

Opens the POST section.

POST syntax

```
-- after section SCHEDULE

POST
```

POSTBLK

Outputs a consolidated file with the parameters in the grid blocks specified with i-j-k indexes.

POSTBLK syntax

```
-- in POST section

POSTBLK
   i1  j1  k1  gridname1  resname1 MinFrame1 MaxFrame1 /
   i2  j2  k2  gridname2  resname2 MinFrame2 MaxFrame2 /
   i3  j3  k3  gridname3  resname3 MinFrame3 MaxFrame3 /
   ...
/

================================================================================

   i#        - i-index of the grid block;
   j#        - j-index of the grid block;
   k#        - k-index of the grid block;
   gridname# - grid name (e.g., defined by the CARFIN keyword);
   resname#  - reservoir name;
 MinFrame#/
   MaxFrame# - range of the report frames for which the output is requested.
```

POSTBLK2

Outputs a consolidated file with the parameters in the grid blocks specified with intervals of i-j-k indexes.

POSTBLK2 syntax

```
-- in POST section

POSTBLK2
   imin1 imax1  jmin1 jmax1  kmin1 kmax1  gridname1  resname1
                                                MinFrame1 MaxFrame1 /
   imin2 imax2  jmin2 jmax2  kmin2 kmax2  gridname2  resname2
                                                MinFrame2 MaxFrame2 /
   imin3 imax3  jmin3 jmax3  kmin3 kmax3  gridname3  resname3
                                                MinFrame3 MaxFrame3 /
   ...
/

================================================================================

   imin#-imax#  - i-index interval of the grid blocks;
   jmin#-jmax#  - j-index interval of the grid blocks;
   kmin#-kmax#  - k-index interval of the grid blocks;
   gridname# - grid name (e.g., defined by the CARFIN keyword);
   resname#  - reservoir name.
```

POSTBCON

Outputs in a consolidated file the parameters for a connection between two grid blocks during the simulated period. The keyword should be followed by a table, where every row defines the output for a particular connection defined by ijk indexes of the grid blocks. Usually, the default parameters should be used for the 4-5 and 9-11 arguments. The properties saved in every output file are defined by the RPTPOST. The output file type can be changed using the POSTSPEC keyword.

POSTBCON syntax

```
-- in POST section

POSTBCON
 ia1 ja1 kb1 gridnamea1 resnamea1  ib1 jb1 kb1 gridnameb1 resnameb1  filename1 /
 ia2 ja2 kb2 gridnamea2 resnamea2  ib2 jb2 kb2 gridnameb2 resnameb2  filename2 /
 ia3 ja3 kb3 gridnamea3 resnamea3  ib3 jb3 kb3 gridnameb3 resnameb3  filename3 /
   ...
/

================================================================================

   ia#        - i-index of the first grid block;
   ja#        - j-index of the first grid block;
   ka#        - k-index of the first grid block;
   gridnamea# - grid name for the first block (e.g., defined by the
                CARFIN keyword);
   resnamea#  - reservoir name for the second grid block;
   ib#        - i-index of the second grid block;
   jb#        - j-index of the second grid block;
   kb#        - k-index of the second grid block;
   gridnameb# - grid name for the second block (e.g., defined by the
                CARFIN keyword);
   resnameb#  - reservoir name for the second grid block;
   filename#  - output file name (if not specified the program uses default
                naming convention).
```

POSTBLOC

Outputs in a consolidated file the parameters in grid blocks during the simulated period. The keyword should be followed by a table, where every row defines the output for a particular grid block. Usually, the default parameters should be used for the 4-6 arguments. The properties saved in every output file are defined by the RPTPOST. The output file type can be changed using the POSTSPEC keyword.

POSTBLOC syntax

```
-- in POST section

POSTBLOC
   i1  j1  k1  gridname1  resname1  filename1 MinFrame1 MaxFrame1 /
   i2  j2  k2  gridname2  resname2  filename2 MinFrame2 MaxFrame2 /
   i3  j3  k3  gridname3  resname3  filename3 MinFrame3 MaxFrame3 /
   ...
/

================================================================================

   i#        - i-index of the grid block;
   j#        - j-index of the grid block;
   k#        - k-index of the grid block;
   gridname# - grid name (e.g., defined by the CARFIN keyword);
   resname#  - reservoir name;
   filename# - output file name (if not specified the program uses default
               naming convention);
 MinFrame#/
   MaxFrame# - range of the report frames for which the output is requested.
```

POSTCELL

Outputs in a consolidated file the parameters in a cell (grid block, stock tank, pipesegment (wellsegment), etc.) during the simulated period. The keyword should be followed by a table, where every row defines the output for a particular cell. The cell name can be detemined from formatted output. The properties saved in every output file are defined by the RPTPOST. The output file type can be changed using the POSTSPEC keyword.

POSTCELL syntax

```
-- in POST section

POSTCELL
   cellname1  filename1 MinFrame1 MaxFrame1 /
   cellname2  filename2 MinFrame2 MaxFrame2 /
   cellname3  filename3 MinFrame3 MaxFrame3 /
   ...
/

================================================================================

   cellname#  - name of the cell (the integer index of cell);
   filename#  - output file name (if not specified the program uses default
                    naming convention);
 MinFrame#/
   MaxFrame# - range of the report frames for which the output is requested.
```

POSTFLD

Outputs in a consolidated file the field parameters during the simulated period. See also the RPTFIELD keyword.

POSTFLD syntax

```
-- in POST section

POSTFLD
  filename MinFrame MaxFrame /

================================================================================

       filename  - output file name (if not specified the program uses default
                   naming convention).
     MinFrame/
       MaxFrame - range of the report frames for which the output is requested.
```

POSTFPCE

Outputs in a consolidated file the parameters for fluid-in-place regions (FIPNUM) during the simulated period. The keyword should be followed by a table, where every row defines the output for a particular fluid-in-place region. Usually, the default parameters should be used for the 2nd argument. The properties saved in every output file are defined by the RPTPOST. The output file type can be changed using the POSTSPEC keyword.

POSTFPCE syntax

```
-- in POST section

POSTFPCE
  fipnum1  filename1 MinFrame1 MaxFrame1 /
  fipnum2  filename2 MinFrame2 MaxFrame2 /
  fipnum3  filename3 MinFrame3 MaxFrame3 /
  ...
/

================================================================================

       fipnume#  - the fluid-in-place region number for which the output is
                   required;
       filename# - output file name (if not specified the program uses default
                   naming convention).
     MinFrame#/
       MaxFrame# - range of the report frames for which the output is requested.
```

POSTFPCO

Outputs in a consolidated file the parameters between two fluid-in-place regions (FIPNUM) during the simulated period. The keyword should be followed by a table, where every row defines the output for two fluid-in-place regions. Usually, the default parameters should be used for the 3rd argument. The properties saved in every output file are defined by the RPTPOST. The output file type can be changed using the POSTSPEC keyword.

POSTFPCO syntax

```
-- in POST section

POSTFPCO
  fipnuma1  fipnumb1  filename1 MinFrame1 MaxFrame1 /
  fipnuma2  fipnumb2  filename2 MinFrame2 MaxFrame2 /
  fipnuma3  fipnumb3  filename3 MinFrame3 MaxFrame3 /
  ...
/

================================================================================

       fipnuma#   - two fluid-in-place region numbers for which the output is
        -fipnumb#   required. The flow rate is reported in the direction
                    from fipnuma# to fipnumb#.
       filename#  - output file name (if not specified the program uses default
                    naming convention).
     MinFrame#/
       MaxFrame# - range of the report frames for which the output is requested.
```

POSTGRUP

Outputs in a consolidated file the parameters for well groups during the simulated period. The keyword should be followed by a table, where every row defines the output for a particular group. Usually, the default parameters should be used for the 2nd argument. The properties saved in every output file are defined by the RPTPOST. The output file type can be changed using the POSTSPEC keyword.

POSTGRUP syntax

```
-- in POST section

POSTGRUP
  groupname1  filename1 MinFrame1 MaxFrame1 /
  groupname2  filename2 MinFrame2 MaxFrame2 /
  groupname3  filename3 MinFrame3 MaxFrame3 /
  ...
/

================================================================================

      groupname# - the well group (or point source group) for which the output
                   is required;
       filename# - output file name (if not specified the program uses default
                   naming convention).
     MinFrame#/
       MaxFrame# - range of the report frames for which the output is requested.
```

POSTOBS

Outputs in a consolidated file the parameters in observation points during the simulated period. See also the RPTOBS keyword.

POSTOBS syntax

```
-- in POST section

POSTOBS
  obsname1  filename1 MinFrame1 MaxFrame1 /
  obsname2  filename2 MinFrame2 MaxFrame2 /
  obsname3  filename3 MinFrame2 MaxFrame3 /
  ...
/

================================================================================

       obsname#  - the observation point name for which the output is required;
       filename# - output file name (if not specified the program uses default
                   naming convention).
     MinFrame#/
       MaxFrame# - range of the report frames for which the output is requested.

```

POSTOBSM

Outputs a consolidated file with the parameters in a network of observation points. The network is defined by the template to the point IDs. The report frame and the output file are specified by the POSTSPEC keyword. This keyword is useful for reporting a map of gravity change or ground displacement by using a network of observation points specified with the OBSSPECM keyword.

POSTOBSM syntax

```
-- in POST section

POSTOBSM
  template1 /
  template2 /
  template3 /
  ...       /
/

================================================================================

   template#  - the template of observation point ID.
```

POSTSPEC

Specifies the parameters of the output from the POST section.

POSTSPEC syntax

```
-- in POST section

POSTSPEC
   filetype  filename  filenum  /

================================================================================

    filetype - if CSV (default option) then the files saved from the POST
               section are in comma-separated-values format. If SUM then the
               files are saved in MUFITS summary file format;
    filename - the output file name for the keywords in POST section which
               do not defined the name (e.g., the POSTBLK keyword);
    filenum  - the number of the summary file which is processed by several
               keywords in the POST section (e.g., the POSTBLK keyword).
```

POSTSRC

Outputs in a consolidated file the parameters for sources/sinks during the simulated period. The keyword should be followed by a table, where every row defines the output for a particular source/sink. Usually, the default parameters should be used for the 2nd argument. The properties saved in every output file are defined by the RPTPOST. The output file type can be changed using the POSTSPEC keyword.

POSTSRC syntax

```
-- in POST section

POSTSRC
  srcname1  filename1 MinFrame1 MaxFrame1 /
  srcname2  filename2 MinFrame2 MaxFrame2 /
  srcname3  filename3 MinFrame3 MaxFrame3 /
  ...
/

================================================================================

       srcname#  - the point source/sink name for which the output is required;
       filename# - output file name (if not specified the program uses default
                   naming convention).
     MinFrame#/
       MaxFrame# - range of the report frames for which the output is requested.
```

POSTVAR

Outputs in a consolidated file the scalar variables during the simulated period. See also the RPTVAR keyword. The auxiliary scalar variables are defined by the AUXVAR keyword.

POSTVAR syntax

```
-- in POST section

POSTVAR
  varname1  filename1 MinFrame1 MaxFrame1 /
  varname2  filename2 MinFrame2 MaxFrame2 /
  varname3  filename3 MinFrame3 MaxFrame3 /
  ...
/

================================================================================

       varname#  - the variable for which the output is required;
       filename# - output file name (if not specified the program uses default
                   naming convention).
     MinFrame#/
       MaxFrame# - range of the report frames for which the output is requested.
```

POSTWELL

Outputs in a consolidated file the parameters for wells during the simulated period. The keyword should be followed by a table, where every row defines the output for a particular well. Usually, the default parameters should be used for the 2nd argument. The properties saved in every output file are defined by the RPTPOST. The output file type can be changed using the POSTSPEC keyword.

POSTWELL syntax

```
-- in POST section

POSTWELL
  wellname1  filename1 MinFrame1 MaxFrame1 /
  wellname2  filename2 MinFrame2 MaxFrame2 /
  wellname3  filename3 MinFrame3 MaxFrame3 /
  ...
/

================================================================================

       wellname# - the wellname for which the output is required;
       filename# - output file name (if not specified the program uses default
                   naming convention).
    MinFrame#/
       MaxFrame# - range of the report frames for which the output is requested.
```

PRES

Specifies the PRES property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

PRES syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

PRES
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

PRES
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

PRES
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

PRESFAIL

Specifies the PRESFAIL property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

PRESFAIL syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

PRESFAIL
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

PRESFAIL
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

PRESFAIL
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

PRESLITH

Specifies the PRESLITH property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

PRESLITH syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

PRESLITH
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

PRESLITH
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

PRESLITH
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

PRESSURE

Specifies the PRESSURE property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

PRESSURE syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

PRESSURE
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

PRESSURE
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

PRESSURE
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

PRINTVAR

Reports the current value of the auxiliary variable (see AUXVAR) to the LOG-file

PRINTVAR syntax

```
-- in any section

PRINTVAR
  varname1 varname2 varname3 ... /
/

================================================================================

   varname#    - name of the variable to be reported.

```

.

PROPS

Opens the PROPS section.

PROPS syntax

```
-- after section GRID

PROPS
```

PUMPHEAT

Defines conductive heat flux through the pumping device. The synonym is SRCHEAT.

PUMPHEAT syntax

```
-- in SCHEDULE section

PUMPHEAT
   name1  HeatFlux1 /
   name2  HeatFlux2 /
   name3  HeatFlux3 /
   ...
/

================================================================================

     name# - name of the pumping device (8-byte character) associated with the
             the point source;
 HeatFlux# - conductive heat flux through the pumping device/point source.
```

PUMPINJE

Defines parameters of pumping device. The flow rate is controlled by the pressure on the outlet. The synonym is SRCINJE.

PUMPINJE syntax

```
-- in SCHEDULE section

PUMPINJE
  name1 targ1 injtype1 plim1 volrate1 massrate1 vp1 dp1 /
  name2 targ2 injtype2 plim2 volrate2 massrate2 vp2 dp2 /
  name3 targ3 injtype3 plim3 volrate3 massrate3 vp3 dp3 /
  ...
/

================================================================================

   name#     - pump name (8-byte character);
   targ#     - pump operational target. Available values: MASS - mass rate,
               RATEIN - volumetric rate on inlet, RATEOUT - volumetric rate on
               outlet;
   injtype#  - fluid used for operational control (default value recomended);
   plim#     - maximum pressure at the pump outlet;
   volrate#  - volumetric rate;
   massrate# - mass rate;
   vp#       - maximum volumetrix rate of the pumping device;
   dp#       - the pressure gap for control function.
```

PUMPOPEN

Changes the mode of the pumping device. The default mode to be specified is OPEN. The synonym is SRCOPEN.

PUMPOPEN syntax

```
-- in SCHEDULE section

PUMPOPEN
   name1  mode1 /
   name2  mode2 /
   name3  mode3 /
   ...
/

================================================================================

     name# - name of the pumping device (8-byte character);
     mode# - the pump mode. Available values are: OPEN (default value) - the
             pump is open, SHUT - the pump is shut.
```

PUMPPARA

Defines parameters of pumping device.

PUMPPARA syntax

```
-- in SCHEDULE section

PUMPPARA
  name1 mode1 targ1 dir1 tran1 dpmin1 dpmax1 /
  name2 mode2 targ2 dir2 tran2 dpmin2 dpmax2 /
  name3 mode3 targ3 dir3 tran3 dpmin3 dpmax3 /
  ...
/

================================================================================

   name#     - pump name (8-byte character);
   mode#     - the pump mode. Available values are: OPEN (default value) - the
               pump is open, SHUT - the pump is shut, FREE - the flux is
               calculated using a user defined transmissibility (not tested).
   targ#     - pump operational target. Available values: MASS - mass rate,
               RATEIN - volumetric rate on inlet, RATEOUT - volumetric rate on
               outlet;
   dir#      - direction of the pumping device. Available values: 1 - the device
               is pumping fluid from the 1st to the 2nd cell in the connection
               (see CONNID), -1 - the device is pumping fluid from the 2nd to
               the 1st cell in the connetion);
   tran#     - transimissibility;
   dpmin#    - pressure gap for control function for the inlet of the pump;
   dpmax#    - pressure gap for control function for the outlet of the pump.
```

PUMPPROD

Defines parameters of pumping device. The flow rate is controlled by the pressure on the inlet. The synonym is SRCPROD.

PUMPPROD syntax

```
-- in SCHEDULE section

PUMPPROD
  name1 targ1 injtype1 plim1 volrate1 massrate1 vp1 dp1 /
  name2 targ2 injtype2 plim2 volrate2 massrate2 vp2 dp2 /
  name3 targ3 injtype3 plim3 volrate3 massrate3 vp3 dp3 /
  ...
/

================================================================================

   name#     - pump name (8-byte character);
   targ#     - pump operational target. Available values: MASS - mass rate,
               RATEIN - volumetric rate on inlet, RATEOUT - volumetric rate on
               outlet;
   injtype#  - fluid used for operational control (default value recomended);
   plim#     - minimum pressure at the pump inlet;
   volrate#  - volumetric rate;
   massrate# - mass rate;
   vp#       - maximum volumetrix rate of the pumping device;
   dp#       - the pressure gap for control function.
```

PUMPSHUT

Changes the mode of the pumping device. The default mode to be specified is SHUT. The synonym is SRCSHUT.

PUMPSHUT syntax

```
-- in SCHEDULE section

PUMPSHUT
   name1  mode1 /
   name2  mode2 /
   name3  mode3 /
   ...
/

================================================================================

     name# - name of the pumping device (8-byte character);
     mode# - the pump mode. Available values are: OPEN - the pump is open,
             SHUT (default value) - the pump is shut.
```

PV2TRMLT

Set the dependence of the transmissibility multiplier on the pore volume multiplier.

PV2TRMLT syntax

```
-- within the ROCK-ENDROCK brackets

PV2TRMLT
   porvmult1    tranmult1  /
   porvmult2    tranmult2  /
   porvmult3    tranmult3  /
   ...
/

================================================================================

   porvmult# - the pore volume multiplier;
   tranmult# - the transmissibility multiplier.

================================================================================

   NOTE:   The table must begin with zero values in both columns.
```

PVT-ENDPVT

PVT-ENDPVT brackets confine a section of the RUN-file for input of thermophysical properties of additional phases defined by the ADDPHASE keyword. These brackets should be used in the PROPS section the same number of times as the number of PVTNUM regions used in the simulation.

PVT-ENDPVT syntax

```
-- in section PROPS

PVT
  pvtnum phasename /

-- keywords for input of thermophysical properties of additional phases

ENDPVT

================================================================================

     pvtnum    - PVTNUM region ID for which the properties are specified;
     phasename - the name of the additional phase (see ADDPHASE).
```

PVTNUM

Specifies the PVTNUM property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

PVTNUM syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

PVTNUM
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

PVTNUM
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

PVTNUM
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

PWVMODEL

Allows switching between different models of the porosity waves.

PWVMODEL syntax

```
-- within the ROCK-ENDROCK brackets

PWVMODEL
   ModelName /

================================================================================

   ModelName  - Name of the model.
                Possible options:
                1. MODEL0 - the latest version of the pororsity waves model.
                   This is the default option.
                1. MODEL1 - the old model.
```

PWVPEMLT

Defines the multiplier for modelling the plastic behaviour of rock.

PWVPEMLT syntax

```
-- within the ROCK-ENDROCK brackets

PWVPEMLT
   Peff1    M1  Mgr1 /
   Peff2    M2  Mgr2 /
   Peff3    M3  Mgr3 /
   ...
/

================================================================================

   Peff#     - the effective pressure, i.e. PRESLITH-PRES=PRESEFF;
   M1#       - the multiplier M for small porosity (is used for modelling the
                decompaction weakening);
   Mgr#      - the multiplier for large porosity (the quantitiy M_gr
               constraining the growth rate of the volume multiplier).

================================================================================

   NOTE:   If the table is not specified for a ROCK region, then it is assumed
           that M=1d0, Mgr=1e-10.
```

PWVROCK

Defines the viscosity of rock and the bulk modulus of rock dependence on temperature.

PWVROCK syntax

```
-- within the ROCK-ENDROCK brackets

PWVROCK
   temp1   etas1     G1 /
   temp2   etas2     G2 /
   temp3   etas3     G3 /
   ...
/

================================================================================

   temp#     - temperature in absolute units, i.e. [K] or [R]
   etas#     - the shear viscosity of rock;
   G#        - the elastic shear modulus.

================================================================================

   NOTE:   1. Any omitted value in the column 2 and 3 is treated as +Infinity.
           2. Constant parameters can be specified by entering a table with a
               single row.
```

RANDSEED

Initializes the random numbers generator.

RANDSEED syntax

```
-- in sections RNSPEC, GRID, INIT, SCHEDULE, in U-PROGRAM

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

RANDSEED
   seed1  seed2   /

================================================================================

    seed# - an integer used for intialization of the random numbers generator.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

-- to set the seed internaly (using time) the following command should be used.

RANDSEED
   INTERNAL   /

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

REFINE

Selects a grid to be active.

REFINE syntax

```
-- in every section except RUNSPEC and POST

REFINE
   gridname  resname /

================================================================================

    gridname - grid name (8-byte character);
    resname  - the name of reservoir in which the grid is defined.
```

REGALL

Enables application of arithmetic operations via ADDREG, COPYREG, EQUALREG, MAXVAREG, MINVAREG, MULTIREG, OPERAREG keyword to additional cells, e.g., created using the BOUNDARY keyword, associated with the parent grid created by the corresponding MAKE keyword. This cells usually do not belong to the parent grid box. The dual keyword REGBOX disables this option.

REGALL syntax

```
-- in all sections except POST

REGALL
```

REGBOX

Disables application of arithmetic operations via ADDREG, COPYREG, EQUALREG, MAXVAREG, MINVAREG, MULTIREG, OPERAREG keyword to additional cells, e.g., created using the BOUNDARY keyword, associated with the parent grid created by the corresponding MAKE keyword. This cells usually do not belong to the parent grid box. The dual keyword REGALL enables this option.

REGBOX syntax

```
-- in all sections except POST

REGBOX
```

REPORTS

Defines the reports outputted in the LOG-file.

REPORTS syntax

```
-- in SCHEDULE section

REPORT
  repmnem1 repmnem2 repmnem3 ... /

================================================================================

     repmnem# - a mnemmonic name of a report in LOG file, e.g.
                   CONV    - convergence reports;
                   MATBAL  - material balance reports;
                   LINSOL  - linear solver report;
                   NOTHING - clears the report list.
```

RESET

Resets (clears) the vector of primary variables in a box of grid blocks. The primary variables can be initialized back with the UPDATE keyword using the secondary variables.

RESET syntax

```
-- in the INIT or SCHEDULE section

RESET
   imin1 imax1   jmin1 jmax1   kmin1 kmax1 /
   imin2 imax2   jmin2 jmax2   kmin2 kmax2 /
   imin3 imax3   jmin3 jmax3   kmin3 kmax3 /
   ...
/

================================================================================

   imin#/imax# - the boundaries of the input box along i-indexation axis.
                 By default these values are equal to the arguments 1 and 2
                 of the keyword BOX.
   jmin#/jmax# - the boundaries of the input box along j-indexation axis.
                 By default these values are equal to the arguments 3 and 4
                 of the keyword BOX.
   kmin#/kmax# - the boundaries of the input box along k-indexation axis.
                 By default these values are equal to the arguments 5 and 6
                 of the keyword BOX.
```

RESETNAM

Resets (clears) the vector of primary variables in all cells which character names (if assigned) belongs to character template. The primary variables can be initialized back with the UPDATNAM keyword using the secondary variables.

RESETNAM syntax

```
-- in the INIT or SCHEDULE section

RESETNAM
   template1 /
   template2 /
   template3 /
   ...
/

================================================================================

   template#   - character name template.
```

RESTART

Requests the simulation restart. The simulation will restart from time defined by either specified file number or the CONTINUE keyword if its encountered earlier in SCHEDULE.

RESTART syntax

```
-- in the RUNSPEC section

RESTART
    report_file   timestep /

================================================================================

   report_file   - number of the report file from which restarting is requested.
      timestep   - next timestep after restarting.
```

ROCK-ENDROCK

ROCK-ENDROCK brackets confine a section of the RUN-file for input of thermophysical properties of reservoir rocks. These brackets should be used in the PROPS section the same number of times as the number of ROCKNUM regions used in the simulation.

ROCK-ENDROCK syntax

```
-- in section PROPS

ROCK
   rocknum /

-- keywords for input of thermophysical properties of reservoir rock

ENDROCK

================================================================================

     rocknum - ROCKNUM region ID for which the properties are specified.
```

ROCKCOMP

Enables the rock compaction option.

ROCKCOMP syntax

```
-- in RUNSPEC section

ROCKCOMP
```

ROCKDENS

Defines the rock grain density dependence of pressure and temperature.

ROCKDENS syntax

```
-- within the ROCK-ENDROCK brackets

ROCKDENS
   den0  Pref Tref   compr  heatexp /

================================================================================

   den0     - the reference density (default 2000 kg/m3);
   Pref     - the reference pressure (default 1 atm);
   Tref     - the reference temperature (default 300 K);
   compr    - the rock compressibility (default 0);
   heatexp  - the heat expansion coefficient for rock (default 0).


================================================================================

   NOTE:   The rock density is calculated as
           denrock=den0*(1+compr*(P-Pref)-heatexp*(T-Tref))
```

ROCKDH

Defines the rock grain density (in METRIC units kg/m^3) and heat capacity (in METRIC units kJ/kg/K). This keyword must be used within ROCK-ENDROCK brackets.

ROCKDH syntax

```
-- within ROCK-ENDROCK brackets

ROCKDH
   dens  heatcap /

================================================================================

       dens    - rock density
       heatcap - rock heat capacity
```

ROCKECL

Defines the rock density and compressibility. The keyword must be used only in isothermal modelling. This keyword must be used within ROCK-ENDROCK brackets. Note, that the rock compaction option is enabled by the ROCKCOMP keyword.

ROCKECL syntax

```
-- within ROCK-ENDROCK brackets

ROCKECL
   pref  compr /

================================================================================

      pref   - the reference pressure;
      compr  - the rock compressibility.
```

ROCKMULT

Enables the ROCKMULT option.

ROCKMULT syntax

```
-- in RUNSPEC section

ROCKMULT
```

ROCKNUM

Specifies the ROCKNUM property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ROCKNUM syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ROCKNUM
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ROCKNUM
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ROCKNUM
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ROCKTAB

Defines the rock compaction data in tabular form. The keyword must be used only in isothermal modeling. This keyword must be used within ROCK-ENDROCK brackets. Note, that the rock compaction option is enabled by the ROCKCOMP keyword.

ROCKTAB syntax

```
-- within ROCK-ENDROCK brackets

ROCKTAB
   pres1   pvmult1  tranmult1  /
   pres2   pvmult2  tranmult2  /
   pres3   pvmult3  tranmult3  /
   ...
/

================================================================================

      pres#     - pressure (must increase down the column);
      pvmult#   - pore volume multiplier;
      tranmult# - transmissibility multiplier.
```

RPTFIELD

Defines the output for field (field injection/production rates or totals).

RPTFIELD syntax

```
-- in INIT section

RPTFIELD
   mnemonic1  mnemonic2  mnemonic3  ...  /

================================================================================

      mnemonic# - is the mnemonic of a property saved in the files *.0000.SUM,
                  *.0001.SUM, *.0002.SUM, etc for fluid-in-place regions.
                  If one of the mnemonics is ASCII then the formatted file is
                  saved. Mnemonic NOTHING clears the report list.
```

RPTFIP

Defines the output for fluid in place regions.

RPTFIP syntax

```
-- in INIT or SCHEDULE section

RPTFIP
   mnemonic1  mnemonic2  mnemonic3  ...  /

================================================================================

      mnemonic# - is the mnemonic of a property saved in the files *.0000.SUM,
                  *.0001.SUM, *.0002.SUM, etc for fluid-in-place regions.
                  If one of the mnemonics is ASCII then the formatted file is
                  saved. Mnemonic NOTHING clears the report list.
```

RPTFUNC

Reports a function previously specified with the FUNCSPEC keyword.

RPTFUNC syntax

```
-- in   - all sections except RUNSPEC and POST
        - OPTIM-ENDOPTIM subsection of the U-program

RPTFUNC
  funcid  n argmin argmax  outfile /

================================================================================

  funcid  - a user function id (must be 1,2,3,...,100);
  n       - number of report items in the range argmin through argmax;
  argmin  - minimal argument of the report interval;
  argmax  - maximal argument of the report interval;
  outfile - output file name.
```

RPTGRID

Defines the output from the GRID section.

RPTGRID syntax

```
-- in GRID section

RPTGRID
   mnemonic1  mnemonic2  mnemonic3  ...  /

================================================================================

      mnemonic# - is the mnemonic of a property saved in the file *.GRID.SUM.
                  If one of the mnemonics is ASCII then the formatted file
                  is saved.
```

RPTGRIDC

Defines the output from the GRID section for the COUPLED option.

RPTGRIDC syntax

```
-- in GRID section

RPTGRIDC
   mnemonic1  mnemonic2  mnemonic3  ...  /

================================================================================

      mnemonic# - is the mnemonic of a property saved from the GRID section
                  if COUPLED option is used.
                  If one of the mnemonics is ASCII then the formatted file
                  is saved.
```

RPTGROUP

Defines the output for the groups of wells or point sources. This is a synonym for RPTGRUP.

RPTGROUP syntax

```
-- in INIT or SCHEDULE section

RPTGROUP
   mnemonic1  mnemonic2  mnemonic3  ...  /

================================================================================

      mnemonic# - is the mnemonic of a property saved in the files *.0000.SUM,
                  *.0001.SUM, *.0002.SUM, etc for the groups of wells or point
                  sources.
                  If one of the mnemonics is ASCII then the formatted file is
                  saved. Mnemonic NOTHING clears the report list.
```

RPTGRUP

Defines the output for the groups of wells or point sources. This is a synonym for RPTGROUP.

RPTGRUP syntax

```
-- in INIT or SCHEDULE section

RPTGRUP
   mnemonic1  mnemonic2  mnemonic3  ...  /

================================================================================

      mnemonic# - is the mnemonic of a property saved in the files *.0000.SUM,
                  *.0001.SUM, *.0002.SUM, etc for the groups of wells or point
                  sources.
                  If one of the mnemonics is ASCII then the formatted file is
                  saved. Mnemonic NOTHING clears the report list.
```

RPTOBJ

Reports the current value of the objective function to the OBJ-file. The function name can be specified by user. More than one objective function can reported by multiple specification of that keyword.

RPTOBJ syntax

```
-- in POST section

RPTOBJ
   objname /

================================================================================

   objname   - the name of the objective function to be written in the
               OBJ-file.
```

RPTOBS

Defines the output for the observation points (see the OBSPOINT keyword).

RPTOBS syntax

```
-- in INIT or SCHEDULE section

RPTOBS
   mnemonic1  mnemonic2  mnemonic3  ...  /

================================================================================

      mnemonic# - is the mnemonic of a property saved in the files *.0000.SUM,
                  *.0001.SUM, *.0002.SUM, etc for point sources.
                  If one of the mnemonics is ASCII then the formatted file is
                  saved. Mnemonic NOTHING clears the report list.
```

RPTOBLOC

Reports grid block parameters to the OBJ-file.

RPTOBLOC syntax

```
-- in POST section

RPTOBLOC
   i1  j1  k1  gridname1  resname1 property1 repframe1  objname1  /
   i2  j2  k2  gridname2  resname2 property2 repframe2  objname2  /
   i3  j3  k3  gridname3  resname3 property3 repframe3  objname3  /
   ...                                                           /
/

================================================================================

      i#        - i-index of the grid block;
      j#        - j-index of the grid block;
      k#        - k-index of the grid block;
      gridname# - grid name (e.g., defined by the CARFIN keyword);
      resname#  - reservoir name;
      property# - a reported property for the specified well;
      repframe# - report time frame for which the property is reported;
      objname#  - the name of the property in the OBJ-file.
```

RPTOFLD

Reports field properties to the OBJ-file.

RPTOFLD syntax

```
-- in POST section

RPTOFLD
   property1  repframe1  objname1  /
   property2  repframe2  objname2  /
   property3  repframe3  objname3  /
   ...                             /
/

================================================================================

      property# - a reported property defined for the reservoir;
      repframe# - report time frame for which the property is reported;
      objname#  - the name of the property in the OBJ-file.
```

RPTOFPCE

Reports parameters for fluid-in-place regions (FIPNUM) to the OBJ-file.

RPTOFPCE syntax

```
-- in POST section

RPTOFPCE
   fipnum1 property1 repframe1  objname1  /
   fipnum2 property2 repframe2  objname2  /
   fipnum3 property3 repframe3  objname3  /
   ...                                    /
/

================================================================================

      fipnum#   - the fluid-in-place region number for which the output is
                  required;
      property# - a reported property for the specified fipnum region;
      repframe# - report time frame for which the property is reported;
      objname#  - the name of the property in the OBJ-file.
```

RPTOFPCO

Reports parameters on the interface between two fluid-in-place regions (FIPNUM) to the OBJ-file.

RPTOFPCO syntax

```
-- in POST section

RPTOFPCO
   fipnuma1  fipnumb1 property1 repframe1  objname1  /
   fipnuma2  fipnumb2 property2 repframe2  objname2  /
   fipnuma3  fipnumb3 property3 repframe3  objname3  /
   ...                                               /
/

================================================================================

      fipnuma#   - two fluid-in-place region numbers for which the output is
       -fipnumb#   required. The flow rate is reported in the direction
                   from fipnuma# to fipnumb#;
      property#  - a reported property for the specified fipnum regions;
      repframe#  - report time frame for which the property is reported;
      objname#   - the name of the property in the OBJ-file.
```

RPTOOBS

Reports parameters of the observation points to the OBJ-file.

RPTOOBS syntax

```
-- in POST section

RPTOOBS
   obspoint1 property1 repframe1  objname1  /
   obspoint2 property2 repframe2  objname2  /
   obspoint3 property3 repframe3  objname3  /
   ...                                     /
/

================================================================================

      obspoint#  - the name of observation point;
      property#  - a reported property for the specified fipnum regions;
      repframe#  - report time frame for which the property is reported;
      objname#   - the name of the property in the OBJ-file.
```

RPTOSRC

Reports point source/sink parameters to the OBJ-file.

RPTOSRC syntax

```
-- in POST section

RPTOSRC
   srcname1 property1 repframe1  objname1  /
   srcname2 property2 repframe2  objname2  /
   srcname3 property3 repframe3  objname3  /
   ...                                     /
/

================================================================================

      srcname#   - the point source/sink name for which the output is required;
      property#  - a reported property for the specified fipnum regions;
      repframe#  - report time frame for which the property is reported;
      objname#   - the name of the property in the OBJ-file.
```

RPTOVAR

Reports auxiliary variable to the OBJ-file.

RPTOVAR syntax

```
-- in POST section

RPTOVAR
   varname1 repframe1  objname1  /
   varname2 repframe2  objname2  /
   varname3 repframe3  objname3  /
   ...                                  /
/

================================================================================

      varname#   - the auxiliary variable name for which the output is required;
      repframe#  - report time frame for which the property is reported;
      objname#   - the name of the property in the OBJ-file.
```

RPTOWELL

Reports well parameters to the OBJ-file.

RPTOWELL syntax

```
-- in POST section

RPTOWELL
   wellname1 property1 repframe1  objname1  /
   wellname2 property2 repframe2  objname2  /
   wellname3 property3 repframe3  objname3  /
   ...                                      /
/

================================================================================

      wellname# - well name;
      property# - a reported property for the specified well;
      repframe# - report time frame for which the property is reported;
      objname#  - the name of the property in the OBJ-file.
```

RPTPOST

Defines the properties saved from the POST section. Note, that the program searches for these properties only in the summary files. Thus, these properties should be included in output from the SCHEDULE section, e.g., by the RPTSUM keyword.

RPTPOST syntax

```
-- in POST section

RPTPOST
   mnemonic1  mnemonic2  mnemonic3  ...  /

================================================================================

      mnemonic# - is the mnemonic of a property saved from the POST section.
                  If one of the mnemonics is ASCII then the formatted file
                  is saved. If one of the mnemonics is ASCII then the binary
                  file is saved. If one of the mnemonics is NOTHING then the
                  output list is cleared.
```

RPTSATTA

Enables the output of the saturation functions. This keyword should be used in the PROPS section outside the brackets SAT-ENDSAT. It should be used after the saturation functions are defined in the corresponding SATNUM region. The output file is saved in CSV format. The first column is the liquid phase saturation, the second is the liquid phase relative permeability, the third is the gas phase relative permeability, and the fourth column is the capillary pressure. The fourth column is saved only if the CAPPRES keyword was specified in the RUNSPEC section

RPTSATTA syntax

```
-- in PROPS section

RPTSATTA
  satnum  filename  n  smin  smax  sys   phasename /

================================================================================

      satnum    - SATNUM region ID for which the output is requested;
      filename  - the file name in which saturation functions are saved. This
                  file should have extension .csv;
      n         - number of output points in the interval [smin,smax];
      smin-smax - the boundaries of the output interval;
      sys       - 1 or 2. By using this flag user can choose a two-phase system
                  in three-phase models for which the functions are output
                  (the default value 1 is recommended);
      phasename - this should be name of an auxiliary phase provided by the
                  option ADDPHASE.
```

RPTSRC

Defines the output for point sources.

RPTSRC syntax

```
-- in INIT or SCHEDULE section

RPTSRC
   mnemonic1  mnemonic2  mnemonic3  ...  /

================================================================================

      mnemonic# - is the mnemonic of a property saved in the files *.0000.SUM,
                  *.0001.SUM, *.0002.SUM, etc for point sources.
                  If one of the mnemonics is ASCII then the formatted file is
                  saved. Mnemonic NOTHING clears the report list.
```

RPTSUM

Defines the output from the SCHEDULE section.

RPTSUM syntax

```
-- in INIT or SCHEDULE section

RPTSUM
   mnemonic1  mnemonic2  mnemonic3  ...  /

================================================================================

      mnemonic# - is the mnemonic of a property saved in the files *.0000.SUM,
                  *.0001.SUM, *.0002.SUM, etc. If one of the mnemonics is ASCII
                  then the formatted file is saved.
```

RPTSUMC

Defines the output from the SCHEDULE section for the COUPLED option.

RPTSUMC syntax

```
-- in INIT section

RPTSUMC
   mnemonic1  mnemonic2  mnemonic3  ...  /

================================================================================

      mnemonic# - is the mnemonic of a property saved from the SCHEDULE section
                  if COUPLED option is used. If one of the mnemonics is ASCII
                  then the formatted file is saved.
```

RPTVAR

Defines the output for auxiliary scalar variables (see AUXVAR keyword).

RPTVAR syntax

```
-- in INIT or SCHEDULE section

RPTVAR
   mnemonic1  mnemonic2  mnemonic3  ...  /

================================================================================

      mnemonic# - is the mnemonic of a auxiliary variable saved in the files
                  *.0000.SUM, *.0001.SUM, *.0002.SUM, etc.
                  If one of the mnemonics is ASCII then the formatted file is
                  saved. Mnemonic NOTHING clears the report list.
```

RPTWELL

Defines the output for wells.

RPTWELL syntax

```
-- in INIT or SCHEDULE section

RPTWELL
   mnemonic1  mnemonic2  mnemonic3  ...  /

================================================================================

      mnemonic# - is the mnemonic of a property saved in the files *.0000.SUM,
                  *.0001.SUM, *.0002.SUM, etc for wells.
                  If one of the mnemonics is ASCII then the formatted file is
                  saved. Mnemonic NOTHING clears the report list.
```

RTZBOUND

Defines the domain extent along R, Theta and Z axes if Radial grid is used. This keyword should be specified within brackets MAKE-ENDMAKE immediately after the opening bracket MAKE. The 1st argument of the keyword MAKE must be RADIAL.

RTZBOUND syntax

```
-- within MAKE/ENDMAKE brackets.

RTZBOUND
   rmin rmax  tmin tmax  zmin zmax   rincr tincr zincr  lr1 lt1 lz1 /

================================================================================

     rmin/rmax - the domain boundaries along axis r (rmin<rmax)
     tmin/tmax - the domain boundaries along axis theta (tmin<tmax) [rad]
     zmin/zmax - the domain boundaries along axis z (zmin<zmax)
     rincr     - the increment of the grid block sizes along axis R. With
                 increasing i-index every next grid block is xincr times larger
                 then the previous block;
     tincr     - the increment of the grid block sizes along axis Theta. With
                 increasing j-index every next grid block is tincr times larger
                 then the previous block;
     zincr     - the increment of the grid block sizes along axis Z. With
                 increasing k-index every next grid block is zincr times larger
                 then the previous block;
     lr1       - the length along Or of the grid block i=1. If this argument is
                 defined then rincr is automatically calculated within software;
     lt1       - the length along Ot of the grid block j=1. If this argument is
                 defined then tincr is automatically calculated within software;
     lz1       - the length along Oz of the grid block k=1. If this argument is
                 defined then zincr is automatically calculated within software;
```

RUNSPEC

Opens the RUNSPEC section.

RUNSPEC syntax

```
-- This should be the first keyword in any RUN-file

RUNSPEC
```

SAT-ENDSAT

SAT-ENDSAT brackets confine a section of the RUN-file for input of saturation functions, which are the relative permeability and capillary pressure. These brackets should be used in the PROPS section the same number of times as the number of SATNUM regions used in the simulation. If the second argument is not blank then the functions are loaded for the additional phase specified by the ADDPHASE keyword. Note, that the additional phase is always interpreted as "a more liquid phase" than the other phases related to the EOS-module.

SAT-ENDSAT syntax

```
-- in section PROPS

SAT
   satnum phasename /

-- keywords for input of realtive permeability and capillary pressure

ENDSAT

================================================================================

     satnum    - SATNUM region ID for which the properties are specified;
     phasename - name of the additional phase (see ADDPHASE). If this second
                 argument is blank then the properties for the EOS-module are
                 loaded. If it is not blank then the properties asscociated
                 with the additional phase are loaded.
```

SATCRT

Enables saturation functions scaling near critical conditions.

SATCRT syntax

```
-- in RUNSPEC section

SATCRT
```

SATNUM

Specifies the SATNUM property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SATNUM syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SATNUM
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SATNUM
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SATNUM
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SATOPT

Defines parameters of the interpolation/extrapolation methods for the saturation functions using the tabulated data. If specified outside the brackets SAT-ENDSAT then the parameters are applied to all following data in the brackets. If specified within the brackets SAT-ENDSAT then the parameters are applied only to the data in the current brackets.

SATOPT syntax

```
-- in section PROPS

SATOPT
  opt1 opt2 opt3 opt4 ... opt40 /


================================================================================

  opt#=0 - option is off;
  opt#=1 - option is on;

  Relative permeability:
  opt1  - piecewise linear function (default 0);
  opt2  - line break at the critical saturation (default 1);
  opt3  - relative permeability is not exceeding 1 if extrapolated for high
          saturations (default: depends on EOS module);
  opt4  - line break at the point rel-perm=1 if option opt3 is on
          (default 1);
  opt5  - line break at the maximum saturation in the table is option opt3 is on
          (default 1);


  Capillary pressure:
  opt11 - piecewise linear function (default 0);
  opt12 - line break on the left, if two values in the table are equal
          (default 1);
  opt13 - line break on the right, if two values in the table are equal
          (default 1);

  Surface tension on pressure and temperature:
  opt21 - piecewise linear function (default 0).
```

SATTAB

Defines the relative permeability and capillary pressure (the latter if CAPPRES is specified in the RUNSPEC section). This keyword must be used within SAT-ENDSAT brackets. The keyword is followed by a saturation table, where the column 1 is for the liquid phase saturation, the column 2 is for the liquid phase relative permeability, the column 3 is for the gas phase relative permeability, and the column 4 is for the capillary pressure. Every row of the table defines a point on the relative permeability and capillary pressure curves. There are the following constraints:

- the minimal relative permeability of either liquid or gas phase must be 0;
- the liquid phase saturation (column 1) must not decrease down the column;
- the liquid phase relative permeability (column 2) must not decrease down the column;
- the gas phase relative permeability (column 3) must not increase down the column.

SATTAB syntax

```
-- within SAT-ENDSAT brackets

SATTAB
  sliq1  krliq1  krgas1  pcap1 /
  sliq2  krliq2  krgas2  pcap2 /
  sliq3  krliq3  krgas3  pcap3 /
...
/

================================================================================

      sliq#  - liquid phase saturation
      krliq# - liquid phase relative permeability
      krgas# - gas phase relative permeability
      pcap#  - capillary pressure
```

SATTGAS

Defines the relative permeability and capillary pressure (the latter if CAPPRES is specified in the RUNSPEC section). This keyword must be used within SAT-ENDSAT brackets. The keyword is followed by a saturation table, where the column 1 is for the gas phase saturation, the column 2 is for the gas phase relative permeability, the column 3 is for the liquid phase relative permeability, and the column 4 is for the capillary pressure. Every row of the table defines a point on the relative permeability and capillary pressure curves. There are the following constraints:

- the minimal relative permeability of either liquid or gas phase must be 0;
- the gas phase saturation (column 1) must not decrease down the column;
- the gas phase relative permeability (column 2) must not decrease down the column;
- the liquid phase relative permeability (column 3) must not increase down the column.

SATTGAS syntax

```
-- within SAT-ENDSAT brackets

SATTGAS
  sgas1  krgas1  krliq1  pcap1 /
  sgas2  krgas2  krliq2  pcap2 /
  sgas3  krgas3  krliq3  pcap3 /
...
/

================================================================================

      sgas#  - gas phase saturation
      krgas# - gas phase relative permeability
      krliq# - liquid phase relative permeability
      pcap#  - capillary pressure
```

SATTLIQ

Defines the relative permeability and capillary pressure (the latter if CAPPRES is specified in the RUNSPEC section). This keyword must be used within SAT-ENDSAT brackets. The keyword is followed by a saturation table, where the column 1 is for the liquid phase saturation, the column 2 is for the liquid phase relative permeability, the column 3 is for the gas phase relative permeability, and the column 4 is for the capillary pressure. Every row of the table defines a point on the relative permeability and capillary pressure curves. There are the following constraints:

- the minimal relative permeability of either liquid or gas phase must be 0;
- the liquid phase saturation (column 1) must not decrease down the column;
- the liquid phase relative permeability (column 2) must not decrease down the column;
- the gas phase relative permeability (column 3) must not increase down the column.

SATTLIQ syntax

```
-- within SAT-ENDSAT brackets

SATTLIQ
  sliq1  krliq1  krgas1  pcap1 /
  sliq2  krliq2  krgas2  pcap2 /
  sliq3  krliq3  krgas3  pcap3 /
...
/

================================================================================

      sliq#  - liquid phase saturation
      krliq# - liquid phase relative permeability
      krgas# - gas phase relative permeability
      pcap#  - capillary pressure
```

This is a synonym for the SATTAB keyword.

SAVECPG

Enables the output of the grid file in the corner-point format. This option is available if the first argument of the MAKE keyword is CART, RADIAL or CORNER.

SAVECPG syntax

```
-- within MAKE-ENDMAKE brackets

SAVECPG
   filename  imin imax   jmin jmax  kmin kmax  /

================================================================================

     filename  - output file name
     imin/imax - the boundaries of the box along i-index axis, for which the
                 grid file is saved.
     jmin/jmax - the boundaries of the box along j-index axis, for which the
                 grid file is saved.
     kmin/kmax - the boundaries of the box along k-index axis, for which the
                 grid file is saved.
```

SAVEGRID

Enables the output of the GRID file (at present, this is the default option). The keyword allows to change the file name and output mode. See also the NOGRID keyword.

SAVEGRID syntax

```
-- in GRID section

SAVEGRID
   mode  filename /

================================================================================

    mode=ASCII  - formatted GRID file is saved,
        =BINARY - binary GRID file is saved;
       filename - GRID file name
```

SAVEMVS

Enables the output of the MVS file (at present, this is the default option). The keyword allows to change the file name and output mode. See also the NOMVS keyword.

SAVEMVS syntax

```
-- within brackets MAKE-ENDMAKE

SAVEMVS
   mode filename /

================================================================================

    mode=ASCII  - formatted MVS file is saved,
        =BINARY - binary MVS file is saved;
       filename - MVS file name
```

SAVEMVSW

Enables the output of the MVS file for wells (at present, this is the default option). The keyword allows to change the file name and output mode. See also the NOMVSW keyword.

SAVEMVSW syntax

```
-- within brackets MAKE-ENDMAKE

SAVEMVSW
   mode filename /

================================================================================

    mode=ASCII  - formatted MVS file is saved,
        =BINARY - binary MVS file is saved;
       filename - MVS file name for wells.
```

SCHEDULE

Opens the SCHEDULE section.

SCHEDULE syntax

```
-- after section INIT

SCHEDULE
```

SEPCOND

Specifies a new separator or changes an existing separator. This keyword also associates well or well group with separator conditions. Note, that currently the separators are enabled for use only with the COMPS module.

SEPCOND syntax

```
-- in the SCHEDULE section

SEPCOND
   name1 group1 stage1 tempc1 pres1 liqstg1  gasstg1  2*nu1  eosnum1 /
   name2 group2 stage2 tempc2 pres2 liqstg2  gasstg2  2*nu2  eosnum2 /
   name3 group3 stage3 tempc3 pres3 liqstg3  gasstg3  2*nu3  eosnum3 /
...                                                                  /
/

================================================================================

    name#      - the separator name (8-byte character).
    group#     - the name of the group of wells for which the separator acts as
                 default (8-byte character).
    stage#     - the separator stage index (i.e. 1,2,3,4,5, or 6). Default is
                 1.
    tempc#     - the stage temperature (relative). Default is the standard
                 temperature.
    pres#      - the stage pressure. Default is the standard pressure.
    liqstg#    - the index of the destination stage for liquid. Default is 0 -
                 means that liquid flows to the next stage. For the last stage
                 0 means 'to the stock tank'.
    gasstg#    - the index of the destination stage for gas. Default is 0,
                 which means 'add to separator gas'.
    nu#        - not used at present.
    eosnum#    - equation of state number (default 1).
```

SETDGRAV

Can be used with the GRAVIMET keyword. This keyword redefines the reference values of the density using the values of the fluid density at the current time in SCHEDULE. By default the simulator uses the density from the initial simulation time.

SETDGRAV syntax

```
-- in the SCHEDULE section

SETDGRAV

```

SETREFPT

Can be used with the GRDDISP keyword. This keyword redefines the reference values of the pore pressure and temperature using their values at the current time in SCHEDULE. By default the simulator uses the values of pressure and temperature from the initial simulation time.

SETREFPT syntax

```
-- in the SCHEDULE section

SETREFPT

```

SETOBJ

Equate the current value of the objective function to the value specified by user.

SETOBJ syntax

```
-- in POST section

SETOBJ
   val /

================================================================================

   val   - the objective function value.
```

SIGMA

Specifies default sigma factor in double porosity simulations (see also DUALPORO). This default value is used if sigma factor cannot be calculated using other input data, for instance specified by the SIGMAV keyword.

SIGMA syntax

```
-- in the GRID section

SIGMA
   sigma /

================================================================================

   sigma    -  the sigma factor.
```

SIGMAV

Specifies the SIGMAV property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SIGMAV syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SIGMAV
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SIGMAV
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SIGMAV
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SPECGRID

This keyword does not affect the data loaded in simulation. It can be used for checking that the data provided in an include file have the same dimension (ni,nj,nk) as specified by the MAKE keyword. If the (ni,nj,nk) numbers differ from those specified by the MAKE keyword, the simulation terminates prematurely with error.

SPECGRID syntax

```
-- within MAKE-ENDMAKE brackets

SPECGRID
   ni nj nk /

================================================================================

      ni - number of grid blocks along i-index direction;
      nj - number of grid blocks along j-index direction;
      nk - number of grid blocks along k-index direction.

      This keyword is used only for grid file checks. If the number of grid
      blocks in the grid file differs from those specified by the keyword MAKE
      then simulation is aborted with error.
```

SRCHEAT

Defines conductive heat flux through the point source. The synonym is PUMPHEAT.

SRCHEAT syntax

```
-- in SCHEDULE section

SRCHEAT
   name1  HeatFlux1 /
   name2  HeatFlux2 /
   name3  HeatFlux3 /
   ...
/

================================================================================

     name# - name of the pumping device (8-byte character) associated with the
             the point source;
 HeatFlux# - conductive heat flux through the pumping device/point source.
```

SRCINJE

Defines parameters of the injection for point sources. The flow rate is controlled by the pressure on the outlet of the pumping device. The synonym is PUMPINJE.

SRCINJE syntax

```
-- in SCHEDULE section

SRCINJE
  name1 targ1 injtype1 plim1 volrate1 massrate1 vp1 dp1 /
  name2 targ2 injtype2 plim2 volrate2 massrate2 vp2 dp2 /
  name3 targ3 injtype3 plim3 volrate3 massrate3 vp3 dp3 /
  ...
/

================================================================================

   name#     - pump name (8-byte character);
   targ#     - pump operational target. Available values: MASS - mass rate,
               RATEIN - volumetric rate on inlet, RATEOUT - volumetric rate on
               outlet;
   injtype#  - fluid used for operational control (default value recomended);
   plim#     - maximum pressure at the pump outlet;
   volrate#  - volumetric rate;
   massrate# - mass rate;
   vp#       - maximum volumetrix rate of the pumping device;
   dp#       - the pressure gap for control function.
```

SRCOPEN

Changes the mode of the point source. The default mode to be specified is OPEN. The synonym is PUMPOPEN.

SRCOPEN syntax

```
-- in SCHEDULE section

SRCOPEN
   name1  mode1 /
   name2  mode2 /
   name3  mode3 /
   ...
/

================================================================================

     name# - name of the pumping device (8-byte character) associated with the
             the point source;
     mode# - the pump mode. Available values are: OPEN (default value) - the
             pump is open, SHUT - the pump is shut.
```

SRCPOTEN

Enables calculation of the potential injection and production for wells (sources) and well groups.

SRCPOTEN syntax

```
-- in RUNSPEC section

SRCPOTEN
```

SRCPROD

Defines parameters of the production for point sources. The flow rate is controlled by the pressure on the inlet of the pumping device. The synonym is PUMPPROD.

SRCPROD syntax

```
-- in SCHEDULE section

SRCPROD
  name1 targ1 injtype1 plim1 volrate1 massrate1 vp1 dp1 /
  name2 targ2 injtype2 plim2 volrate2 massrate2 vp2 dp2 /
  name3 targ3 injtype3 plim3 volrate3 massrate3 vp3 dp3 /
  ...
/

================================================================================

   name#     - pump name (8-byte character);
   targ#     - pump operational target. Available values: MASS - mass rate,
               RATEIN - volumetric rate on inlet, RATEOUT - volumetric rate on
               outlet;
   injtype#  - fluid used for operational control (default value recomended);
   plim#     - minimum pressure at the pump inlet;
   volrate#  - volumetric rate;
   massrate# - mass rate;
   vp#       - maximum volumetrix rate of the pumping device;
   dp#       - the pressure gap for control function.
```

SRCSHUT

Changes the mode of the point source. The default mode to be specified is SHUT. The synonym is PUMPSHUT.

SRCSHUT syntax

```
-- in SCHEDULE section

SRCSHUT
   name1  mode1 /
   name2  mode2 /
   name3  mode3 /
   ...
/

================================================================================

     name# - name of the pumping device (8-byte character) associated with the
             point source;
     mode# - the pump mode. Available values are: OPEN - the pump is open,
             SHUT (default value) - the pump is shut.
```

SRCSPECG

Defines the point source. The location of the point source in the XY plane must be specified by either i and j indexes or x and y coordinates. The depth of the point source must be specified by either k index or Z coordinate.

SRCSPECG syntax

```
-- within MAKE-ENDMAKE brackets

SRCSPECG
   name1   i1 j1 k1   x1 y1 z1   mode1 /
   name2   i2 j2 k2   x2 y2 z2   mode2 /
   name3   i3 j3 k3   x3 y3 z3   mode3 /
   ...
/

================================================================================

    name#    - the point source name (a 8-byte character);
    i#-j#-k# - the i-j-k indexes of the grid block in which the point source
               is located;
    x#-y#-z# - the coordinates of the point source;
    mode#    - the point source mode, i.e. the pumping device mode (default
               value is SHUT).
```

SRCSPECM

Creates an array of point sources for the specified grid box.

SRCSPECM syntax

```
-- within MAKE-ENDMAKE brackets

SRCSPECM
   template1 srt1  imin1 imax1 jmin1 jmax1 kmin1 kmax1  mode1 /
   template2 srt2  imin2 imax2 jmin2 jmax2 kmin2 kmax2  mode2 /
   template3 srt3  imin3 imax3 jmin3 jmax3 kmin3 kmax3  mode3 /
   ...
/

================================================================================

     template# - the template for the point sources names
          srt# - initial integer to add to the template (default: srt#=1)
                 The sources names are created as follows
                 do k=kmin,kmax; do j=jmin,jmax; do i=imin,imax
                   source_name=template+srt
                   srt=srt+1
                 enddo;enddo;enddo
   imin#-imax# - range of the i-index for the grid blocks where the point
                 sources are placed
   jmin#-jmax# - range of the j-index for the grid blocks where the point
                 sources are placed
   kmin#-kmax# - range of the k-index for the grid blocks where the point
                 sources are placed
    mode#      - the point source mode, i.e. the pumping device mode (default
                 value is SHUT).
```

SSEPCOND

Associates point source or group of point sources with a separator. The separator must be defined by the SEPCOND keyword. This keyword is a synonym to WSEPCOND.

SSEPCOND syntax

```
-- in the SCHEDULE section

SSEPCOND
    template1 sepname1 /
    template2 sepname2 /
    template3 sepname3 /
...                    /
/

================================================================================

    template#  - point source name template or point source group template
                 (8-byte character).
    sepname#   - the name of the separator which will act for the well or well
                 group.
```

START

Defined the start date of the simulation. See also the DATES keyword.

START syntax

```
-- in RUNSPEC section

START
   day  month  year  time  sec /

================================================================================

      day   - calendar day   (1,2,3,...,31)
      month - calendar month (JAN,FEB,MAR,APR,MAY,JUN,JUL,AUG,SEP,OCT,NOV,DEC)
      year  - year (e.g., 2012)
      time  - time of the day in HH:MM format (default 12:00)
      sec   - number of seconds of float format (default 0.0)
```

STARTTIM

Defined the start time of the simulation.

STARTTIM syntax

```
-- in RUNSPEC section

STARTTIM
   time /

================================================================================

     time - time in days
```

STCOND

Changes the standard pressure and temperature.

STCOND syntax

```
-- in the RUNSPEC or PROPS section

STCOND
   pres  tempc /

================================================================================

   pres    -  standard pressure (default 1 atm)
   tempc   -  standard temperature (default 15.56 C)
```

STOP

Terminates the simulation and preceeds to the POST section.

STOP syntax

```
-- in the SCHEDULE section

STOP
```

SURFTENS

Enables surface tension (capillary pressure) dependence on pressure and temperature.

SURFTENS syntax

```
-- in section RUNSPEC

SURFTENS
```

TANKPRES

Defines default stock tank pressure.

TANKPRES syntax

```
-- in section RUNSPEC

TANKPRES
   pres /

================================================================================

     pres - default stock tank pressure
```

TEMP

Specifies the TEMP property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

TEMP syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

TEMP
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

TEMP
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

TEMP
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

TEMPC

Specifies the TEMPC property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

TEMPC syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

TEMPC
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

TEMPC
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

TEMPC
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

TIME

Defines the report times and advances the simulation to further report times. When the simulator encounters this keyword it proceeds the simulation further in time and produces the required outputs.

TIME syntax

```
-- in SCHEDULE section

TIME
   time1 time2 time3 ... /

================================================================================

     time# - report times (in days). Every time# initiates the
             program to advance the simulation to this time and save a new
             *.####.SUM file.
```

TITLE

Changes the prefix in the output file names. By default, this prefix is equal to the RUN-file name.

TITLE syntax

```
-- in RUNSPEC section

TITLE
   newtitle /

================================================================================

    newtitle - a string which is used for output file names. The output file
               names will be as follows newtitle.GRID.SUM, newtitle.0000.SUM,
               newtitle.0001.SUM, etc.
```

TOLENG

Switches the program to using weaker tollerances that are more typical for engineering simulations.

TOLENG syntax

```
-- in SCHEDULE section

TOLENG
```

TSTEP

Defines the report times and advances the simulation to further report times. When the simulator encounters this keyword it proceeds the simulation further in time and produces the required outputs.

TSTEP syntax

```
-- in SCHEDULE section

TSTEP
   tstep1 tstep2 tstep3 ... /

================================================================================

     tstep# - steps between report times (in days). Every tstep# initiates the
              program to advance the simulation to time->time+tstep#. After
              every time->time+tstep# a new *.####.SUM file is saved.
```

TUNING

Sets controls for the calculation procedures.

TUNING syntax

```
-- in SCHEDULE section

TUNING
  dtlimit dtmax dtexp dtmin /

================================================================================

    dtlimit - limit for the next time step;
    dtmax   - maximal time step;
    dtexp   - this field is for explicit specification of the next time step;
    dtmin   - the minimal time step.
```

TYPENUM

Specifies the TYPENUM property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

TYPENUM syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

TYPENUM
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

TYPENUM
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

TYPENUM
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

UF2DNAME

Defines the f(x,y) function ID loaded by the succeeding UF2DSPEC or UF2DSPE0 keyword. The function can be used by the keywords OPERATE, OPERAREG or OPERANAM if the operation is UF2D.

UF2DNAME syntax

```
-- in all sections except RUNSPEC and POST

UF2DNAME
  funcid /

================================================================================

  funcid - a user function id (must be 1,2,3,...,100; default 1). The function
           id affect the data loaded by the UF2DSPE0 and UF2DSPEC keywords.
```

UF2DSPE0

Defines in a tabular form the f(x,y) function. The function ID is defined by the preceding UF2DNAME keyword. The function can be used by the keywords OPERATE, OPERAREG or OPERANAM if the operation is UF2D. Alternatively, the function can be loaded by the LOADGRD keyword.

UF2DSPE0 syntax

```
-- in all sections except RUNSPEC and POST

UF2DSPE0
  nx xmin xmax   ny ymin ymax /
  val11  val21  val31   ...   valnx1  /
  val12  val22  val32   ...   valnx2  /
  val13  val23  val33   ...   valnx3  /
  ...
  val1ny val2ny val3ny  ...   valnxny /
/

================================================================================

  nx      - number of data points uniformly placed along axis X;
  xmin    - the lower interval limit along axis X;
  xmax    - the upper interval limit along axis X;
  ny      - number of data points uniformly placed along axis Y;
  ymin    - the lower interval limit along axis Y;
  ymax    - the upper interval limit along axis Y;
  valij   - the function value at
            x=xmin+(xmax-xmin)*(i-1)/(nx-1) and
            y=ymin+(ymax-ymin)*(j-1)/(ny-1).
```

UF2DSPEC

Defines in a tabular form the f(x,y) function. The function ID is defined by the preceding UF2DNAME keyword. The function can be used by the keywords OPERATE, OPERAREG or OPERANAM if the operation is UF2D. Alternatively, the function can be loaded by the LOADGRD keyword.

UF2DSPEC syntax

```
-- in all sections except RUNSPEC and POST

UF2DSPEC
      x1     x2     x3            xnx
  y1  val11  val21  val31   ...   valnx1  /
  y2  val12  val22  val32   ...   valnx2  /
  y3  val13  val23  val33   ...   valnx3  /
  ...
  yny val1ny val2ny val3ny  ...   valnxny /
/

================================================================================

  xi      - data points placed along X axis. xi must be ascending;
  yj      - data points placed along Y axis. yi must be ascending;
  valij   - the f(x,y) function value at x=xi and y=yj.
```

UF3DNAME

Defines the f(x,y,z) function ID loaded by the succeeding UF3DSPEC keyword. The function can be used by the keywords ARITH, ARITHCND and its aliases.

UF3DNAME syntax

```
-- in all sections except RUNSPEC and POST

UF3DNAME
  funcid  xscl yscl zscl  Nneighbor  Power  NBox  Smoothing /

================================================================================

  funcid    - a user function id (must be 1,2,3,...,100; default 1). The
              function id affect the data loaded by the UF3DSPEC keywords.
  xscl,
  yscl,
  zscl      - introduce the anisotropy when interp[olating the data points.
              By default al these scaling factors equal 1.0.
  NNeighbor - Defines the number of neighbors used in the interpolation. By
              default it is equal to 5.
  Power     - The power used in the smoothing function. Default 2.0.
  Smoothing - This constant characterizes the smoothnes of the interpolation.
              By default it equals 1e-10.
```

UF3DSPEC

Defines specific values of a f(x,y,z) function. The function ID as well as the interpolation method are defined by the preceding UF3DNAME keyword. The function can be used by the ARITH keyword and its aliases to distribute a property in 3D space.

UF3DSPEC syntax

```
-- in all sections except RUNSPEC and POST

UF3DSPEC
    x1  y1  z1  val1 /
    x2  y2  z2  val2 /
    x3  y3  z3  val3 /
...
/

================================================================================

  x#, y#, z#   - coordinates of a point in 3D space.
  val#         - the function value at the coorresponding coordinates.
```

UPDATE

Calculates (initializes) the vector of primary variables using the secondary variables in a box of grid blocks. The primary variables can be reset using the RESET keyword.

UPDATE syntax

```
-- in the INIT or SCHEDULE section

UPDATE
   imin1 imax1   jmin1 jmax1   kmin1 kmax1 /
   imin2 imax2   jmin2 jmax2   kmin2 kmax2 /
   imin3 imax3   jmin3 jmax3   kmin3 kmax3 /
   ...
/

================================================================================

   imin#/imax# - the boundaries of the input box along i-indexation axis.
                 By default these values are equal to the arguments 1 and 2
                 of the keyword BOX.
   jmin#/jmax# - the boundaries of the input box along j-indexation axis.
                 By default these values are equal to the arguments 3 and 4
                 of the keyword BOX.
   kmin#/kmax# - the boundaries of the input box along k-indexation axis.
                 By default these values are equal to the arguments 5 and 6
                 of the keyword BOX.
```

UPDATNAM

Calculates (initializes) the vector of primary variables in all cells which character names (if assigned) belongs to character template. The primary variables can be reset using the RESETNAM keyword.

UPDATNAM syntax

```
-- in the INIT or SCHEDULE section

UPDATNAM
   template1 /
   template2 /
   template3 /
   ...
/

================================================================================

   template#   - character name template.
```

VARS

Affect the chopping of primary variables in non-linear iterations.

VARS syntax

```
-- in SCHEDULE section

VARS
   varname1 constr1  para1 onoff1  parb1 /
   varname2 constr2  para2 onoff2  parb2 /
   varname3 constr3  para3 onoff3  parb3 /
   ...
/

================================================================================

    varname#   - name of the primary variable (e.g., PRES, TEMP, SAT, ...),
                 to which the constraint is applied;
    constr#    - character name of constraint applied in a single Newton
                 iteration:
                    DMAX - maximum alteration;
                    MINV - minimum value;
                    MAXV - maximum value;
                    ...  - the list is incomplete.
    para#      - the parameter defined by the previous argument 2;
    onoff#     - if ON then the constraint is applied, if OFF the constraint is
                 not applied;
    parb#      - an additional parameter defined by the argument 2.
```

VFPCHK

Specifies the threshhold value for defined PTAB quantities in VFP table (see VFPINJ keyword).

VFPCHK syntax

```
-- in the SCHEDULE section

VFPCHK
   vfpchk /

================================================================================

  vfpchk - threshhold value for VFP tables checking and extrapolation (default
           1e10).
```

VFPINJ

Specifies a vertical flow performance (VFP) table for injection wells. The tables can be associated with particular wells by the WELLINJE keyword. All PTAB values larger then VFPCHK threshhold are assumed as not-defined quantities. By default the program stabilizes the tables. This behavior can be changed by the VFPSTBLZ keyword.

VFPINJ syntax

```
-- in the SCHEDULE section

VFPINJ
  id  BHPdepth  FLOname PFIXname Units PTABname /
  FLO1   FLO2   FLO3  ... FLOn   /
  PFIX1  PFIX2  PFIX3 ... PFIXk  /
  p1   PTABp11 PTABp12 PTABp13 ... PTABp1n /
  p2   PTABp21 PTABp22 PTABp23 ... PTABp2n /
  p3   PTABp31 PTABp32 PTABp33 ... PTABp3n /
  ...                                      /
  pk   PTABpk1 PTABpk2 PTABpk3 ... PTABpkn /

================================================================================

  id        - VFPINJ table ID (an integer). By default it is 1.
  BHPdepth  - Bottom-hole datum depth.
  FLOname   - Definition of the FLO variable (e.g. GAS or WAT means that BHP is
              tabulated agains gas or water flow rates).
  PFIXname  - Must be THP.
  Units     - Units used in the table. If the units are not consistent with the
              simulation units, then the program issues a warning.
  PTABname  - Must be BHP.
  FLO#      - Flow rate (must be sorted in ascending order).
  n         - Number of FLO quanitites (must be greater than or equal to 2).
  PFIX$     - THP values.
  k         - Number of THP values (must be greater than or equal to 1).
  p$        - The sequence number of the THP value for which the BHP values
              are specified.
  PTABp$#   - BHP value for p$-th THP value and #-th FLO value.
```

VFPSTBLZ

Allows to change the processing of the VFP tables. By default the program slightly modifies the VFP tables in the unstabilized regions to avoid wells switching between different control modes. This behavior can be disabled by this keyword.

VFPSTBLZ syntax

```
-- in the SCHEDULE section

VFPSTBLZ
   onoff /

================================================================================

  onoff  - ON or OFF (default ON). If the switch is set to ON, then any new
           VFP table is internally modified in the unstable regions.
```

VISTAB

For additional phase loads the viscosity on pressure and temperature. This keyword must be used within PVT-ENDPVT brackets. See also the AD-DPHASE keyword.

VISTAB syntax

```
-- within EOS-ENDEOS, or PVT-ENDPVT brackets, or in the PROPS section

VISTAB
        temp1  temp2  temp3  ...  /
 pres1  vis11  vis12  vis13  ...  /
 pres2  vis21  vis22  vis23  ...  /
 pres3  vis31  vis32  vis33  ...  /
 ...    ...    ...    ...    ...  /
/

================================================================================

     temp#  -  temperature (K);
     pres$  -  pressure;
     vis$#  -  viscosity for a given pressure (pres$) and temperature (temp#).
```

V2KOFF

Disable the V2KON option.

V2KOFF syntax

```
-- in RUNSPEC section

V2KOFF
```

V2KON

Makes it possible to specify some argumetns of the keywords by the scalar variables (see AUXVAR keyword). By default this option is disables. Only specific keywords are influenced by that option.

V2KON syntax

```
-- in RUNSPEC section

V2KON
```

WBHPRATE

Defines the well pumping device volumetric rate used to reach the constant bottom-hole pressure target limit.

WBHPRATE syntax

```
-- in SCHEDULE section

WBHPRATE
   name1  bhprate1  dimflag1 /
   name2  bhprate2  dimflag2 /
   name3  bhprate3  dimflag3 /
   ...
/

================================================================================

   name#     - well name or well name template;
   bhprate#  - the volumetric rate of the well pumping device used to reach the
               BHP target;
   dimflag#  - if LIQ (default) then the vp# is in liquid rate units, if GAS
               then vp# is in gas rate units.
```

WEEKTOL

Applies less stringent convergence tolerances to reduce the number of Newton method iterations.

WEEKTOL syntax

```
-- in section SCHEDULE

WEEKTOL
```

WEFAC

Defines wells efficiency factors.

WEFAC syntax

```
-- in SCHEDULE section

WEFAC
   name1 eff1 /
   name2 eff2 /
   name3 eff3 /
   ...
/

================================================================================

   name# - well name or well name template;
   eff#  - efficiency factor.
```

WELLINJE

Defines the parameters and controls of injection wells.

WELLINJE syntax

```
-- in SCHEDULE section

WELLINJE
   name1 mode1 targ1 volrate1 massrate1 bhp1 vp1 injtype1 dimflag1 thp1 vfp1 /
   name2 mode2 targ2 volrate2 massrate2 bhp2 vp2 injtype2 dimflag2 thp2 vfp2 /
   name3 mode3 targ3 volrate3 massrate3 bhp3 vp3 injtype3 dimflag3 thp3 vfp3 /
   ...
/

================================================================================

   name#     - well name or well name template;
   mode#     - well mode. Available values: OPEN - well open for flow (default),
               STOP - well stoped above formation, SHUT - well completely
               isolated form the formation.
   targ#     - well operational target. Available values: MASS - mass rate,
               RATE - volumetric rate at stock tank conditions, BHP - constant
               bottom hole pressure, GRUP - under group control;
   volrate#  - volumetric rate;
   massrate# - mass rate;
   bhp#      - bottom-hole pressure (either limit (MASS,RATE) or target
               parameter (BHP));
   vp#       - maximum volumetrix rate of the pumping device;
   injtype#  - fluid to be injected (dependent on EOS module). At present
               no values defined in kernel, and fluid is pumped from
               the corresponding stock tank. If fluid is specified by the
               keyword then parameters in the stock tank are redefined;
   dimflag#  - if LIQ (default) then the volrate# and vp# are in liquid rate
               units, if GAS then volrate# and vp# are in gas rate units;
   thp#      - Tubing =head pressure limit or target (by default it is not
               defined);
   vfp#      - ID of VFPINJ table associated with the well.
```

WELLPROD

Defines the parameters and controls of production wells.

WELLPROD syntax

```
-- in SCHEDULE section

WELLPROD
   name1 mode1 targ1 volrate1 massrate1 bhp1 vp1 nu1 dimflag1 /
   name2 mode2 targ2 volrate2 massrate2 bhp2 vp2 nu2 dimflag2 /
   name3 mode3 targ3 volrate3 massrate3 bhp3 vp3 nu3 dimflag3 /
   ...
/

================================================================================

   name#     - well name or well name template;
   mode#     - well mode. Available values: OPEN - well open for flow (default),
               STOP - well stoped above formation, SHUT - well completely
               isolated form the formation.
   targ#     - well operational target. Available values: MASS - mass rate,
               RESV - volumetric rate at reservoir conditions, BHP - constant
               bottom hole pressure;
   volrate#  - volumetric rate;
   massrate# - mass rate;
   bhp#      - bottom-hole pressure (either limit (MASS,RESV) or target
               parameter (BHP));
   vp#       - maximum volumetrix rate of the pumping device;
   nu#       - not used at present;
   dimflag#  - if LIQ (default) then the volrate# and vp# are in liquid rate
               units, if GAS then volrate# and vp# are in gas rate units.
```

WELSPECC

Defines a well by using the map coordinates of the well head.

WELSPECC syntax

```
-- within MAKE-ENDMAKE brackets

WELSPECC
   name1 group1 xloc1 yloc1 datum1 nu1 r0_1 3*nu1 eosnum1 model1 4*nu1
                                            fluxnum1 flag1 z1/
   name2 group2 xloc2 yloc2 datum2 nu1 r0_2 3*nu2 eosnum2 model2 4*nu2
                                            fluxnum2 flag2 z2/
   name3 group3 xloc3 yloc3 datum3 nu1 r0_3 3*nu3 eosnum3 model3 4*nu3
                                            fluxnum3 flag3 z3/
   ...
/

================================================================================

   name#    - well name;
   group#   - name of the group to which the well belongs (default FIELD);
   xloc#    - x coordinate of the well head (map coordinates);
   yloc#    - y coordinate of the well head (map coordinates);
   datum#   - reference depth for bottom hole pressure;
   r0_#     - drainage radius;
   nu#      - a parameter not used at present;
   eosnum#  - equation of state region number used for the fluid properties
              calculation in the well;
   model#   - well model. If this flag is set to AVG (default option) then an
              averaged well model is used. If this flag is set to SEG then
              multisegemented model is used;
   fluxnum# - FLUXNUM region number assigned to all cells of which the well
              is constructed;
   flag#    - if this flag is T (default option) then all segements of the well
              are destributed on a single node at partition of the model. If
              this flag is F then different segements of the well can be
              allocated on different nodes of computer;
   z#       - the default vertical segment of the well is shown down to
              this depth.
```

WELSPECS

Defines the vertical wells. Note, that if this keyword is positioned in SCHEDULE section then the GETWELLS keyword must be specified within MAKE-ENDMAKE brackets.

WELSPECS syntax

```
-- within MAKE-ENDMAKE brackets or in SCHEDULE section

WELSPECS
   name1 group1 iloc1 jloc1 datum1 nu1 r0_1 3*nu1 eosnum1 model1 4*nu1
                                            fluxnum1 flag1 /
   name2 group2 iloc2 jloc2 datum2 nu1 r0_2 3*nu2 eosnum2 model2 4*nu2
                                            fluxnum2 flag2 /
   name3 group3 iloc3 jloc3 datum3 nu1 r0_3 3*nu3 eosnum3 model3 4*nu3
                                            fluxnum3 flag3 /
   ...
/

================================================================================

   name#    - well name;
   group#   - name of the group to which the well belongs (default FIELD);
   iloc#    - i-index of the grid block where the well head is located;
   jloc#    - j-index of the grid block where the well head is located;
   datum#   - reference depth for bottom hole pressure;
   r0_#     - drainage radius;
   nu#      - a parameter not used at present;
   eosnum#  - equation of state region number used for the fluid properties
              calculation in the well;
   model#   - well model. If this flag is set to AVG (default option) then an
              averaged well model is used. If this flag is set to SEG then
              multisegemented model is used;
   fluxnum# - FLUXNUM region number assigned to all cells of which the well
              is constructed;
   flag#    - if this flag is T (default option) then all segements of the well
              are destributed on a single node at partition of the model. If
              this flag is F then different segements of the well can be
              allocated on different nodes of computer.
```

WELLSHUT

Completely isolates a well from reservoir.

WELLSHUT syntax

```
-- in SCHEDULE section

WELLSHUT
   name1 /
   name2 /
   name3 /
   ...
/

================================================================================

   name#  - well name or well name template.
```

WELLSTOP

Stops a well.

WELLSTOP syntax

```
-- in SCHEDULE section

WELLSTOP
   name1 /
   name2 /
   name3 /
   ...
/

================================================================================

   name#  - well name or well name template.
```

WELTARG

Defines a new well target.

WELTARG syntax

```
-- in SCHEDULE section

WELTARG
   name1 targ1 param1 /
   name2 targ2 param2 /
   name3 targ3 param3 /
   ...
/

================================================================================

   name#  - well name or well name template;
   targ#  - new target. Available values: MASS - mass rate, RESV - volumetric
            rate at reservoir conditions, RATE - volumetric rate at stock tank
            conditions, BHP - constant bottom hole pressure.
   param# - the corresponding parameter (depends on the 2nd argument). It is
            either mass rate (MASS), or volumetric rate at reservoir conditions
            (RESV), or volumetric rate at stock tank conditions (RATE), or
            bottom-hole pressure (BHP).
```

WELTRACK

Defines well track (i.e., well trajectory). The keyword must be used for each separate bracnch of the well. The complitions for well tracks are specified by the COMPDATM keyword.

WELTRACK syntax

```
-- within MAKE-ENDMAKE brackets

WELTRACK
   wellname branchID /
   x1  y1  z1  md1
   x2  y2  z2  md2
   x3  y3  z3  md3
   ...               /

================================================================================

   wellname - well name;
   branchID - well branch ID (default 0);
   x#       - x-coordinate;
   y#       - y-coordinate;
   z#       - z-coordinate (i.e. absolute depth);
   md#      - measured depth (i.e. depth along the wellbore).
```

WPIMULT

Multiplies the well connection factor (transmissibility) by a given number.

WPIMULT syntax

```
-- in SCHEDULE section

WPIMULT
   name1 mult1 iloc1 jloc1 kloc1 /
   name2 mult2 iloc2 jloc2 kloc2 /
   name3 mult3 iloc3 jloc3 kloc3 /
   ...
/

================================================================================

   name#     - well name or well name template;
   mult#     - well transmissibility multiplier;
   iloc#/    - i-index, j-index and k-index of the grid block for which the
   jloc#/      transmissibility is multiplied.
   kloc#
```

WPROFILE

Reports in the LOG-file a summary on the porosity and permeability profiles for a vertical well.

WPROFILE syntax

```
-- within MAKE-ENDMAKE brackets

WPROFILE
  wellname /

================================================================================

   wellname      - well name.
```

WSEPCOND

Associates well or well group with a separator. The separator must be defined by the SEPCOND keyword. This keyword is a synonym to SSEPCOND.

WSEPCOND syntax

```
-- in the SCHEDULE section

WSEPCOND
    template1 sepname1 /
    template2 sepname2 /
    template3 sepname3 /
...                    /
/

================================================================================

    template#  - well name template or well group template (8-byte character).
    sepname#   - the name of the separator which will act for the well or well
                 group.
```

XYW2VAR

Reports well coordinates to scalar variables that are created with the AUXVAR keyword.

XYW2VAR syntax

```
-- within MAKE/ENDMAKE brackets

XYW2VAR
   xvar yvar wellname mode/

================================================================================

     xvar      - a scalar variable to that the x coordinate of the well
                 is reported
     yvar      - a scalar variable to that the x coordinate of the well
                 is reported
     wellname  - name of the well
     mode      - must be one of
                     GRID, i.e. the grid coordinates are reported (default)
                     MAP, i.e. the map coordinates are reported.

```

XYZBOUND

Defines the domain extent along X, Y and Z axes if Cartesian grid is used. This keyword should be specified within brackets MAKE-ENDMAKE immediately after the opening bracket MAKE. The 1st argument of the keyword MAKE must be CART.

XYZBOUND syntax

```
-- within MAKE/ENDMAKE brackets.

XYZBOUND
   xmin xmax  ymin ymax  zmin zmax xincr yincr zincr  lx1  ly1  lz1 /

================================================================================

     xmin/xmax - the domain boundaries along axis x (xmin<xmax)
     ymin/ymax - the domain boundaries along axis y (ymin<ymax)
     zmin/zmax - the domain boundaries along axis z (zmin<zmax)
     xincr     - the increment of the grid block sizes along axis X. With
                 increasing i-index every next grid block is xincr times larger
                 than the previous block;
     yincr     - the increment of the grid block sizes along axis Y. With
                 increasing j-index every next grid block is yincr times larger
                 than the previous block;
     zincr     - the increment of the grid block sizes along axis Z. With
                 increasing k-index every next grid block is zincr times larger
                 than the previous block;
     lx1       - the length along Ox of the grid block i=1. If this argument is
                 defined then xincr is automatically calculated within software;
     ly1       - the length along Oy of the grid block j=1. If this argument is
                 defined then yincr is automatically calculated within software;
     lz1       - the length along Oz of the grid block k=1. If this argument is
                 defined then zincr is automatically calculated within software.
```

ZCORN

Defines the depths of grid block corners in the current input box. The input is affected by the BOX keyword. This keyword should be used if the first argument of the MAKE is CART, RADIAL or CORNER.

ZCORN syntax

```
-- within MAKE-ENDMAKE brackets

ZCORN
  depth1 depth2 depth3 ... depthN /

================================================================================

    depth# - depth of a grid block corner.
         N - the total number of the grid block corners in the current input
             box. N=2*(imax-imin+1)*2*(jmax-jmin+1)*2*(kmax-kmin+1).
```

1.8. BINMIXT keywords

The keywords below are available only with BINMIXT EOS module.

BINMIXT

Switches the program to BINMIXT EOS module. Normally, this keyword should not be used because BINMIXT is the default module.

BINMIXT syntax

```
-- in section RUNSPEC

BINMIXT
```

COMP1T

Specifies the COMP1T property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

COMP1T syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

COMP1T
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

COMP1T
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

COMP1T
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

COMP2T

Specifies the COMP2T property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

COMP2T syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

COMP2T
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

COMP2T
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

COMP2T
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ENTHPRIM

The program uses firstly the PHX (pressure-enthalpy-composition) parameters for initialisation of the primeary variables in a grid block. The alternative is the TEMPPRIM keyword.

ENTHPRIM syntax

```
-- in RUNSPEC, INIT or SCHEDULE section

ENTHPRIM
```

ENTHT

Specifies the ENTHT property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ENTHT syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ENTHT
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ENTHT
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ENTHT
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

LOADEOS

Loads EOS-file with the properties of binary mixture. This keyword must be specified if the BINMIXT module is used. At present, the EOS regions are not available for the BINMIXT module. Thus, the LOADEOS keyword can be specified both within brackets EOS-ENDEOS or directly in the PROPS section.

LOADEOS syntax

```
-- in PROPS section

LOADEOS
   filename /

================================================================================

   filename - name of EOS file.
```

PHASES

Allows to define additional mnemonics for the properties of phases (PHASES option). See also the GETPHASE keyword.

PHASES syntax

```
-- in PROPS section

PHASES
   name1  pres1  entht1  comp1t1  comp2t1 /
   name2  pres2  entht2  comp1t2  comp2t2 /
   name3  pres3  entht3  comp1t3  comp2t3 /
   ...
/

================================================================================

   name#   - phase name (4-byte character);
   pres#   - pressure;
   entht#  - total molar enthalpy;
   comp1t# - total molar fraction of the 1st component;
   comp2t# - total molar fraction of the 2nd component.
```

SGASINIT

Specifies the SGASINIT property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SGASINIT syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SGASINIT
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SGASINIT
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SGASINIT
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SLIQINIT

Specifies the SLIQINIT property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SLIQINIT syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SLIQINIT
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SLIQINIT
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SLIQINIT
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

TEMPPRIM

The program uses firstly the PTX (pressure-temperature-composition) parameters for initialisation of the primary variables in a grid block. This is the default option. The alternative is the ENTHPRIM keyword.

TEMPPRIM syntax

```
-- in RUNSPEC, INIT or SCHEDULE section

TEMPPRIM
```

1.9. BLACKOIL keywords

The keywords below are available only with BLACKOIL EOS module.

BLACKOIL

Switches the program to BLACKOIL EOS module.

BLACKOIL syntax

```
-- in RUNSPEC section

BLACKOIL
```

DENSITY

Defines the densities of oil, water and gas at stock-tank conditions.

DENSITY syntax

```
-- within EOS-ENDEOS brackets

DENSITY
  denoil  denwat  dengas /

================================================================================

     denoil - oil density at stock-tank conditions;
     denwat - water density at stock-tank conditions;
     dengas - gas density at stock-tank conditions.
```

DISGAS

Enables the gas component dissolution in the oil phase.

DISGAS syntax

```
-- in RUNSPEC section

DISGAS
```

EQUIL

Defines parameters of initial equilibration. See also the PBVD, PDVD, RSVD and RVVD keywords.

EQUIL syntax

```
-- within EQL-ENDEQL brackets

EQUIL
   datum  pdatum  woc  pcow   goc  pcog /

================================================================================

     datum    - datum depth;
     pdatum   - pressure at the datum depth;
     woc      - water-oil contact depth;
     pcow     - capillary pressure at water-oil contact;
     goc      - gas-oil contact depth;
     pcog     - capillary pressure at gas-oil contact.
```

IKRG

Specifies the IKRG property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

IKRG syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

IKRG
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

IKRG
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

IKRG
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

IKRO

Specifies the IKRO property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

IKRO syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

IKRO
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

IKRO
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

IKRO
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

IKRW

Specifies the IKRW property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

IKRW syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

IKRW
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

IKRW
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

IKRW
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

IPCG

Specifies the IPCG property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

IPCG syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

             -- if the LOADING switch is in IJK position --

IPCG
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

IPCG
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

IPCG
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

IPCW

Specifies the IPCW property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

IPCW syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

IPCW
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

IPCW
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

IPCW
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ISGCR

Specifies the ISGCR property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ISGCR syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ISGCR
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ISGCR
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ISGCR
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ISGL

Specifies the ISGL property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ISGL syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ISGL
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ISGL
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ISGL
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ISGLPC

Specifies the ISGLPC property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ISGLPC syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ISGLPC
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ISGLPC
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ISGLPC
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ISGU

Specifies the ISGU property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ISGU syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ISGU
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ISGU
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ISGU
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ISOGCR

Specifies the ISOGCR property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ISOGCR syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ISOGCR
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ISOGCR
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ISOGCR
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ISOWCR

Specifies the ISOWCR property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ISOWCR syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ISOWCR
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ISOWCR
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ISOWCR
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ISOWL

Specifies the ISOWL property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ISOWL syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ISOWL
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ISOWL
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ISOWL
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ISOWU

Specifies the ISOWU property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ISOWU syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ISOWU
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ISOWU
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ISOWU
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ISWCR

Specifies the ISWCR property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ISWCR syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ISWCR
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ISWCR
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ISWCR
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ISWL

Specifies the ISWL property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ISWL syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ISWL
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ISWL
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ISWL
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ISWLPC

Specifies the ISWLPC property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ISWLPC syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ISWLPC
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ISWLPC
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ISWLPC
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ISWU

Specifies the ISWU property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ISWU syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ISWU
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ISWU
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ISWU
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ISWUPC

Specifies the ISWUPC property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ISWUPC syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ISWUPC
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ISWUPC
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ISWUPC
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

GAS

Enables the gas phase. This keyword indicates that the simulation contains the gas phase.

GAS syntax

```
-- in RUNSPEC section

GAS
```

KRG

Specifies the KRG property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

KRG syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

KRG
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

KRG
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

KRG
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

KRO

Specifies the KRO property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

KRO syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

KRO
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

KRO
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

KRO
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

KRW

Specifies the KRW property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

KRW syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

KRW
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

KRW
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

KRW
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

OIL

Enables the oil phase. This keyword indicates that the simulation contains the oil phase.

OIL syntax

```
-- in RUNSPEC section

OIL
```

PBUB

Specifies the PBUB property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

PBUB syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

PBUB
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

PBUB
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

PBUB
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

PBVD

Defines the initial distribution of the bubble pressure versus depth. The depth values must increase down the column. See also the EQUIL keyword.

PBVD syntax

```
-- within EQL-ENDEQL brackets

PBVD
  depth1  rs1  /
  depth2  rs2  /
  depth3  rs3  /
  ...
/

================================================================================

     depth#  - depth;
     pbub#   - bubble pressure at the specified depth.
```

PCG

Specifies the PCG property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

PCG syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

PCG
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

PCG
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

PCG
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

PCW

Specifies the PCW property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

PCW syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

PCW
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

PCW
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

PCW
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

PDEW

Specifies the PDEW property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

PDEW syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

PDEW
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

PDEW
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

PDEW
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

PDVD

Defines the initial distribution of the dew pressure versus depth. The depth values must increase down the column. See also the EQUIL keyword.

PDVD syntax

```
-- within EQL-ENDEQL brackets

PDVD
  depth1  rs1  /
  depth2  rs2  /
  depth3  rs3  /
  ...
/

================================================================================

     depth#  - depth;
     pdew#   - dew pressure at the specified depth.
```

PVCDO

Defines PVT properties of dead oil. See also the PVDO keyword.

PVCDO syntax

```
-- within EOS-ENDEOS brackets

PVCDO
   pref  bo  compro  mo  viscmo /

================================================================================

    pref   - reference presssure;
    bo     - oil formation volume factor at the reference pressure;
    compro - oil compressibility at the reference pressure;
    mo     - oil viscosity at the reference pressure;
    viscmo - oil 'viscosibility' at the reference pressure.
```

PVCO

Defines PVT properties of live oil. See also the PVTO keyword.

PVCO syntax

```
-- within EOS-ENDEOS brackets

PVCO
  pres1  rs1  bo1   mo1  compr1  viscm1 /
  pres2  rs2  bo2   mo2  compr2  viscm2 /
  pres3  rs3  bo3   mo3  compr3  viscm3 /
  ...
/

================================================================================

      pres#  - buble pressure (must increase down the column);
      rs#    - gas-in-oil ration at buble pressure;
      bo#    - saturated oil formation volume factor;
      mo#    - saturated oil viscosity;
      compr1 - oil compressibility (-dbo#/bo#);
      viscm1 - oil "viscosibility" (dmo#/mo#).
```

PVDG

Defines PVT properties of dry gas.

PVDG syntax

```
-- within EOS-ENDEOS brackets

PVDG
  pres1   bg1   mg1 /
  pres2   bg2   mg2 /
  pres3   bg3   mg3 /
  ...
/

================================================================================

      pres# - pressure (must increase down the column);
      bg#   - gas formation volume factor;
      mg#   - gas viscosity.
```

PVDO

Defines PVT properties of dead oil. See also the PVCDO keyword.

PVDO syntax

```
-- within EOS-ENDEOS brackets

PVDO
  pres1   bo1   mo1 /
  pres2   bo2   mo2 /
  pres3   bo3   mo3 /
  ...
/

================================================================================

      pres# - pressure (must increase down the column);
      bo#   - oil formation volume factor;
      mo#   - oil viscosity.
```

PVTG

Defines PVT properties of wet gas.

PVTG syntax

```
-- within EOS-ENDEOS brackets

PVTG
  pres1   rv1-1    bg1-1    mg1-1
          rv1-2    bg1-2    mg1-2
          rv1-3    bg1-3    mg1-3
          ...                        /
  pres2   rv2-1    bg2-1    mg2-1
          rv2-2    bg2-2    mg2-2
          rv2-3    bg2-3    mg2-3
          ...                        /
  pres3   rv3-1    bg3-1    mg3-1
          rv3-2    bg3-2    mg3-2
          rv3-3    bg3-3    mg3-3
          ...                        /
   ...
/

================================================================================

      pres# - pressure (must increase down the column);
      rv#-# - oil-in-gas ratio;
      bg#-# - gas formation volume factor;
      mg#-# - gas viscosity.
```

PVTO

Defines PVT properties of live oil. See also the PVCO keyword.

PVTO syntax

```
-- within EOS-ENDEOS brackets

PVTO
   rs1    pres1-1    bo1-1    mo1-1
          pres1-2    bo1-2    mo1-2
          pres1-3    bo1-3    mo1-3
          ...                      /
   rs2    pres2-1    bo2-1    mo2-1
          pres2-2    bo2-2    mo2-2
          pres2-3    bo2-3    mo2-3
          ...                      /
   rs3    pres3-1    bo3-1    mo3-1
          pres3-2    bo3-2    mo3-2
          pres3-3    bo3-3    mo3-3
          ...                      /
   ...
/

================================================================================

      rs#     - gas to oil ratio (must increase down the column);
      pres#-# - pressure (must increase down the column for every rs#);
      bo#-#   - oil volume factor;
      mo#-#   - oil viscosity.
```

PVTW

Defines PVT properties of water. See also the WATERTAB keyword.

PVTW syntax

```
-- within EOS-ENDEOS brackets

PVTW
   pref  bw  comprw  mw  viscmw /

================================================================================

    pref   - reference presssure;
    bw     - water formation volume factor at the reference pressure;
    comprw - water compressibility at the reference pressure;
    mw     - water viscosity at the reference pressure;
    viscmw - water 'viscosibility' at the reference pressure.
```

RS

Specifies the RS property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

RS syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

RS
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

RS
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

RS
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

RSVD

Defines the initial distribution of the gas-in-oil ratio versus depth. The depth values must increase down the column. See also the EQUIL keyword.

RSVD syntax

```
-- within EQL-ENDEQL brackets

RSVD
  depth1  rs1  /
  depth2  rs2  /
  depth3  rs3  /
  ...
/

================================================================================

     depth#  - depth;
     rs#     - gas-in-oil ratio at the specified depth.
```

RV

Specifies the RV property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

RV syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

RV
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

RV
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

RV
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

RVVD

Defines the initial distribution of the oil-in-gas ratio versus depth. The depth values must increase down the column. See also the EQUIL keyword.

RVVD syntax

```
-- within EQL-ENDEQL brackets

RVVD
  depth1  rv1  /
  depth2  rv2  /
  depth3  rv3  /
  ...
/

================================================================================

     depth#  - depth;
     rv#     - oil-in-gas ratio at the specified depth.
```

SGAS

Specifies the SGAS property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SGAS syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SGAS
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SGAS
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SGAS
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SGCR

Specifies the SGCR property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SGCR syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SGCR
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SGCR
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SGCR
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SGFN

Defines gas saturation functions.

SGFN syntax

```
-- within SAT-ENDSAT brackets

SGFN
   sgas1  krg1  pcog1 /
   sgas2  krg2  pcog2 /
   sgas3  krg3  pcog3 /
   ...
/

================================================================================

     sgas# - gas saturation;
     krg#  - gas relative permeability;
     pcog# - oil-gas capillary pressure.
```

SGL

Specifies the SGL property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SGL syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SGL
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SGL
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SGL
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SGOF

Defines gas-oil saturation functions versus gas saturation.

SGOF syntax

```
-- within SAT-ENDSAT brackets

SGOF
   sgas1  krg1  krog1  pcog1 /
   sgas2  krg2  krog2  pcog2 /
   sgas3  krg3  krog3  pcog3 /
   ...
/

================================================================================

     sgas# - gas saturation;
     krg#  - gas relative permeability;
     krog# - oil relative permeability;
     pcog# - oil-gas capillary pressure.
```

SGU

Specifies the SGU property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SGU syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SGU
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SGU
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SGU
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SGWFN

Defines gas-water saturation functions.

SGWFN syntax

```
-- within SAT-ENDSAT brackets

SGWFN
   sgas1  krg1  krw1  pcgw1 /
   sgas2  krg2  krw2  pcgw2 /
   sgas3  krg3  krw3  pcgw3 /
   ...
/

================================================================================

     sgas# - gas saturation;
     krg#  - gas relative permeability;
     krw#  - water relative permeability;
     pcgw# - gas-water capillary pressure.
```

SLGOF

Defines gas-oil saturation functions versus liquid saturation.

SLGOF syntax

```
-- within SAT-ENDSAT brackets

SLGOF
   soil1  krg1  krog1  pcog1 /
   soil2  krg2  krog2  pcog2 /
   soil3  krg3  krog3  pcog3 /
   ...
/

================================================================================

     soil# - oil saturation;
     krg#  - gas relative permeability;
     krog# - oil relative permeability;
     pcog# - oil-gas capillary pressure.
```

SLGOF2

Defines gas-oil saturation functions versus liquid saturation.

SLGOF2 syntax

```
-- within SAT-ENDSAT brackets

SLGOF2
   soil1  krog1  krg1  pcog1 /
   soil2  krog2  krg2  pcog2 /
   soil3  krog3  krg3  pcog3 /
   ...
/

================================================================================

     soil# - oil saturation;
     krog# - oil relative permeability;
     krg#  - gas relative permeability;
     pcog# - oil-gas capillary pressure.
```

SOF2

Defines oil saturation functions (2-phase case).

SOF2 syntax

```
-- within SAT-ENDSAT brackets

SOF2
   soil1  krow1  /
   soil2  krow2  /
   soil3  krow3  /
   ...
/

================================================================================

     soil# - oil saturation;
     krow# - oil relative permeability.
```

SOF3

Defines oil saturation functions (3-phase case).

SOF3 syntax

```
-- within SAT-ENDSAT brackets

SOF3
   soil1  krow1  krog1 /
   soil2  krow2  krog2 /
   soil3  krow3  krog3 /
   ...
/

================================================================================

     soil# - oil saturation;
     krow# - oil relative permeability for oil-water flows;
     krog# - oil relative permeability for oil-gas flows.
```

SOGCR

Specifies the SOGCR property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SOGCR syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SOGCR
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SOGCR
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SOGCR
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SOIL

Specifies the SOIL property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SOIL syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SOIL
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SOIL
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SOIL
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SOWCR

Specifies the SOWCR property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SOWCR syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SOWCR
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SOWCR
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SOWCR
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SOWL

Specifies the SOWL property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SOWL syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SOWL
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SOWL
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SOWL
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SOWU

Specifies the SOWU property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SOWU syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SOWU
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SOWU
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SOWU
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

STOG

Defines oil-gas surface tension dependence on pressure.

STOG syntax

```
-- within SAT-ENDSAT brackets

STOG
   refpres  /
   pres1  surftens1 /
   pres2  surftens2 /
   pres3  surftens3 /
   ...
/

================================================================================

   refpres   - reference pressure at that the saturation functions for oil-gas
               system are specified;
   pres#     - oil phase presure (values must increase down the column);
   surtens#  - oil-gas surface tension.
```

STONE1

Switches three-phase oil relative permeability calculation to the 1st Stone model. This keyword influences only the current saturation functions region (see the SAT-ENDSAT keyword).

STONE1 syntax

```
-- within SAT-ENDSAT brackets

STONE1
```

STONE2

Switches three-phase oil relative permeability calculation to the 2st Stone model. This keyword influences only the current saturation functions region (see the SAT-ENDSAT keyword).

STONE2 syntax

```
-- within SAT-ENDSAT brackets

STONE2
```

STOW

Defines oil-water surface tension dependence on pressure.

STOW syntax

```
-- within SAT-ENDSAT brackets

STOW
   refpres  /
   pres1  surftens1 /
   pres2  surftens2 /
   pres3  surftens3 /
   ...
/

================================================================================

   refpres   - reference pressure at that the saturation functions for oil-water
               system are specified;
   pres#     - oil phase presure (values must increase down the column);
   surtens#  - oil-water surface tension.
```

SWAT

Specifies the SWAT property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SWAT syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

             -- if the LOADING switch is in IJK position --

SWAT
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SWAT
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SWAT
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SWATINIT

Specifies the SWATINIT property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SWATINIT syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

             -- if the LOADING switch is in IJK position --

SWATINIT
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SWATINIT
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SWATINIT
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SWCR

Specifies the SWCR property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SWCR syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SWCR
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SWCR
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SWCR
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SWFN

Defines water saturation functions.

SWFN syntax

```
-- within SAT-ENDSAT brackets

SWFN
   swat1  krw1  pcow1 /
   swat2  krw2  pcow2 /
   swat3  krw3  pcow3 /
   ...
/

================================================================================

     swat# - water saturation;
     krw#  - water relative permeability;
     pcow# - water-oil capillary pressure.
```

STWG

Defines gas-water surface tension dependence on pressure.

STWG syntax

```
-- within SAT-ENDSAT brackets

STWG
   refpres  /
   pres1  surftens1 /
   pres2  surftens2 /
   pres3  surftens3 /
   ...
/

================================================================================

   refpres   - reference pressure at that the saturation functions for gas-water
               system are specified;
   pres#     - gas phase presure (values must increase down the column);
   surtens#  - gas-water surface tension.
```

SWL

Specifies the SWL property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SWL syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SWL
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SWL
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SWL
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SWLPC

Specifies the SWLPC property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SWLPC syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SWLPC
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SWLPC
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SWLPC
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SWOF

Defines water-oil saturation functions.

SWOF syntax

```
-- within SAT-ENDSAT brackets

SWOF
   swat1  krw1  krow1  pcow1 /
   swat2  krw2  krow2  pcow2 /
   swat3  krw3  krow3  pcow3 /
   ...
/

================================================================================

     swat# - water saturation;
     krw#  - water relative permeability;
     kro#  - oil relative permeability;
     pcow# - water-oil capillary pressure.
```

SWU

Specifies the SWU property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SWU syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SWU
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SWU
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SWU
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SWUPC

Specifies the SWUPC property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SWUPC syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SWUPC
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SWUPC
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SWUPC
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

VAPOIL

Enables the oil component evaporation in the gas phase.

VAPOIL syntax

```
-- in RUNSPEC section

VAPOIL
```

WATER

Enables the water phase. This keyword indicates that the simulation contains the water phase.

WATER syntax

```
-- in RUNSPEC section

WATER
```

WATERTAB

Defines PVT properties of water. See also the PVTW keyword.

WATERTAB syntax

```
-- within EOS-ENDEOS brackets

WATERTAB
  pres1   bw1   mw1 /
  pres2   bw2   mw2 /
  pres3   bw3   mw3 /
  ...
/

================================================================================

      pres# - pressure (must increase down the column);
      bg#   - water formation volume factor;
      mg#   - water viscosity.
```

WCONHIST

Defines the control data for production wells based on observed rates.

WCONHIST syntax

```
-- in SCHEDULE section

WCONHIST
   name1 mode1 targ1 oilrate1 watrate1 gasrate1 3*nu1 bhp1 /
   name2 mode2 targ2 oilrate2 watrate2 gasrate2 3*nu2 bhp2 /
   name3 mode3 targ3 oilrate3 watrate3 gasrate3 3*nu3 bhp3 /
   ...
/

================================================================================

   name#     - well name or well name template;
   mode#     - well mode. Available values: OPEN - well open for flow (default),
               STOP - well stoped above formation, SHUT - well completely
               isolated form the formation.
   targ#     - well operational target. Available values:
                        ORAT - oil vol. rate at surface conditions;
                        WRAT - water vol. rate at surface conditions;
                        GRAT - gas vol. rate at surface conditions;
                        LRAT - liquid rate at surface conditions;
                        MASS - mass rate;
                        RESV - volumetric rate at reservoir conditions;
                        BHP  - constant bottom hole pressure.
   oilrate#  - oil vol. rate;
   watrate#  - water vol. rate;
   gasrate#  - gas vol. rate;
   nu#       - not used at present;
   bhp#      - bottom-hole pressure (either limit or target parameter (BHP)).
```

WCONINJE

Defines the control data for injection wells

WCONINJE syntax

```
-- in SCHEDULE section

WCONINJE
   name1 injtype1 mode1 targ1 surfrate1 resrate1 bhp1 /
   name2 injtype2 mode2 targ2 surfrate2 resrate2 bhp2 /
   name3 injtype3 mode3 targ3 surfrate3 resrate3 bhp3 /
   ...
/

================================================================================

...
```

1.10. COMPS keywords

The keywords below are available only with the COMPS EOS module.

ACF

Defines the acentric factors for the EoS components.

ACF syntax

```
-- within the EOS-ENDEOS brackets

ACF
   acf(1)  acf(2)  acf(3) ... acf(nc) /

================================================================================

   acf(i) - acentric factor for the ith component (i=1,...,nc);
   nc     - number of components given by the COMPS keyword.
```

ADDEOS

Adds new Equation of State type to the build-in library, or modifies parameters of existing EoS.

ADDEOS syntax

```
-- in the PROPS section

ADDEOS
   eos-name  temp-func  def-coef  use-sshift /

================================================================================

   eos-name    - name of the Equation of State to be added or modified;
   temp-func   - temperature fuction type. Possible values:
                  1 - Peng-Robinson type function (default for new EoS);
                  2 - Redlich-Kwong type function.
   def-coef    - default OMEGAA and OMEGAB coefficients. Should be one of
                  1 - Peng-Robinson coefficients - OMEGAA=0.457235529,
                      OMEGAB=0.077796074 (default for new EoS);
                  2 - Redlich-Kwong coefficients - OMEGAA=0.4274802,
                      OMEGAB=0.08664035.
                 This option also influences calculation of the default value
                 for PSI in the temperature function.
   use-sshift  - must be one of
                  T - use the volume shift coefficients;
                  F - don't use the volume shift coefficients (default for
                 new EoS).
```

BIC

Depending on the modelling options, the simulator calculates the binary interaction coefficient, τ, for the ith and jth component using the following equation

$$\tau = \tau_0 + \tau_1 T + \tau_2 T^2 + \tau_3 T^3 + \left(\varphi_0 + \varphi_1 T + \varphi_2 T^2 + \varphi_3 T^3\right) \frac{1}{2} \frac{x_{(i)} - x_{(j)}}{x_{(i)} + x_{(j)}}$$

The keyword BIC defines the quantities τ_0 for all binary interactions in the fluid. This keyword has a synonym BIC0.

BIC syntax

```
-- within the EOS-ENDEOS brackets

BIC
   tau0-2-1
   tau0-3-1  tau0-3-2
   tau0-4-1  tau0-4-2  tau0-4-3
   tau0-5-1  tau0-5-2  tau0-5-3  tau0-5-4
   ...
   tau0-nc-1 ...       ...       ...       ...    tau0-nc-(nc-1) /

================================================================================

   tau0-i-j    - tau0 coefficient for the ith and jth
                 components of the mixture;
   nc          - number of components given by the COMPS keyword.
```

BIC0

Depending on the modelling options, the simulator calculates the binary interaction coefficient, τ, for the ith and jth component using the following equation

$$\tau = \tau_0 + \tau_1 T + \tau_2 T^2 + \tau_3 T^3 + \left(\varphi_0 + \varphi_1 T + \varphi_2 T^2 + \varphi_3 T^3\right) \frac{1}{2} \frac{x_{(i)} - x_{(j)}}{x_{(i)} + x_{(j)}}$$

The keyword BIC0 defines the quantities τ_0 for all binary interactions in the fluid.

BIC0 syntax

```
-- within the EOS-ENDEOS brackets

BIC0
   tau0-2-1
   tau0-3-1  tau0-3-2
   tau0-4-1  tau0-4-2  tau0-4-3
   tau0-5-1  tau0-5-2  tau0-5-3  tau0-5-4
   ...
   tau0-nc-1 ...       ...       ...       ...    tau0-nc-(nc-1) /

================================================================================

   tau0-i-j   - tau0 coefficient for the ith and jth
                components of the mixture;
   nc         - number of components given by the COMPS keyword.
```

BIC1

Depending on the modelling options, the simulator calculates the binary interaction coefficient, τ, for the ith and jth component using the following equation

$$\tau = \tau_0 + \tau_1 T + \tau_2 T^2 + \tau_3 T^3 + \left(\varphi_0 + \varphi_1 T + \varphi_2 T^2 + \varphi_3 T^3\right) \frac{1}{2} \frac{x_{(i)} - x_{(j)}}{x_{(i)} + x_{(j)}}$$

The keyword BIC1 defines the quantities τ_1 for all binary interactions in the fluid.

BIC1 syntax

```
-- within the EOS-ENDEOS brackets

BIC1
   tau1-2-1
   tau1-3-1  tau1-3-2
   tau1-4-1  tau1-4-2  tau1-4-3
   tau1-5-1  tau1-5-2  tau1-5-3  tau1-5-4
   ...
   tau1-nc-1 ...      ...      ...       ...    tau1-nc-(nc-1) /

================================================================================

   tau1-i-j   - tau1 coefficient for the ith and jth
                components of the mixture;
   nc         - number of components given by the COMPS keyword.
```

BIC2

Depending on the modelling options, the simulator calculates the binary interaction coefficient, τ, for the ith and jth component using the following equation

$$\tau = \tau_0 + \tau_1 T + \tau_2 T^2 + \tau_3 T^3 + \left(\varphi_0 + \varphi_1 T + \varphi_2 T^2 + \varphi_3 T^3\right) \frac{1}{2} \frac{x_{(i)} - x_{(j)}}{x_{(i)} + x_{(j)}}$$

The keyword BIC2 defines the quantities τ_2 for all binary interactions in the fluid.

BIC2 syntax

```
-- within the EOS-ENDEOS brackets

BIC2
   tau2-2-1
   tau2-3-1  tau2-3-2
   tau2-4-1  tau2-4-2  tau2-4-3
   tau2-5-1  tau2-5-2  tau2-5-3  tau2-5-4
   ...
   tau2-nc-1 ...       ...       ...       ...    tau2-nc-(nc-1) /

================================================================================

   tau2-i-j   - tau2 coefficient for the ith and jth
                components of the mixture;
   nc         - number of components given by the COMPS keyword.
```

BIC3

Depending on the modelling options, the simulator calculates the binary interaction coefficient, τ, for the ith and jth component using the following equation

$$\tau = \tau_0 + \tau_1 T + \tau_2 T^2 + \tau_3 T^3 + \left(\varphi_0 + \varphi_1 T + \varphi_2 T^2 + \varphi_3 T^3\right) \frac{1}{2} \frac{x_{(i)} - x_{(j)}}{x_{(i)} + x_{(j)}}$$

The keyword BIC3 defines the quantities τ_3 for all binary interactions in the fluid.

BIC3 syntax

```
-- within the EOS-ENDEOS brackets

BIC3
   tau3-2-1
   tau3-3-1  tau3-3-2
   tau3-4-1  tau3-4-2  tau3-4-3
   tau3-5-1  tau3-5-2  tau3-5-3  tau3-5-4
   ...
   tau3-nc-1 ...      ...      ...       ...    tau3-nc-(nc-1) /

================================================================================

   tau3-i-j   - tau3 coefficient for the ith and jth
                components of the mixture;
   nc         - number of components given by the COMPS keyword.
```

BICX

Depending on the modelling options, the simulator calculates the binary interaction coefficient, τ, for the ith and jth component using the following equation

$$\tau = \tau_0 + \tau_1 T + \tau_2 T^2 + \tau_3 T^3 + \left(\varphi_0 + \varphi_1 T + \varphi_2 T^2 + \varphi_3 T^3\right) \frac{1}{2} \frac{x_{(i)} - x_{(j)}}{x_{(i)} + x_{(j)}}$$

The keyword BICX defines the quantities φ_0 for all binary interactions in the fluid. This keyword has a synonym BICX0.

BICX syntax

```
-- within the EOS-ENDEOS brackets

BICX
   phi0-2-1
   phi0-3-1  phi0-3-2
   phi0-4-1  phi0-4-2  phi0-4-3
   phi0-5-1  phi0-5-2  phi0-5-3  phi0-5-4
   ...
   phi0-nc-1 ...       ...       ...       ...    phi0-nc-(nc-1) /

================================================================================

   phi0-i-j    - phi0 coefficient for the ith and jth
                 components of the mixture;
   nc          - number of components given by the COMPS keyword.
```

BICX0

Depending on the modelling options, the simulator calculates the binary interaction coefficient, τ, for the ith and jth component using the following equation

$$\tau = \tau_0 + \tau_1 T + \tau_2 T^2 + \tau_3 T^3 + \left(\varphi_0 + \varphi_1 T + \varphi_2 T^2 + \varphi_3 T^3\right) \frac{1}{2} \frac{x_{(i)} - x_{(j)}}{x_{(i)} + x_{(j)}}$$

The keyword BICX0 defines the quantities φ_0 for all binary interactions in the fluid.

BICX0 syntax

```
-- within the EOS-ENDEOS brackets

BICX0
   phi0-2-1
   phi0-3-1  phi0-3-2
   phi0-4-1  phi0-4-2  phi0-4-3
   phi0-5-1  phi0-5-2  phi0-5-3  phi0-5-4
   ...
   phi0-nc-1 ...       ...       ...       ...    phi0-nc-(nc-1) /

================================================================================

   phi0-i-j   - phi0 coefficient for the ith and jth
                components of the mixture;
   nc         - number of components given by the COMPS keyword.
```

BICX1

Depending on the modelling options, the simulator calculates the binary interaction coefficient, τ, for the ith and jth component using the following equation

$$\tau = \tau_0 + \tau_1 T + \tau_2 T^2 + \tau_3 T^3 + \left(\varphi_0 + \varphi_1 T + \varphi_2 T^2 + \varphi_3 T^3\right) \frac{1}{2} \frac{x_{(i)} - x_{(j)}}{x_{(i)} + x_{(j)}}$$

The keyword BICX1 defines the quantities φ_1 for all binary interactions in the fluid.

BICX1 syntax

```
-- within the EOS-ENDEOS brackets

BICX1
   phi1-2-1
   phi1-3-1  phi1-3-2
   phi1-4-1  phi1-4-2  phi1-4-3
   phi1-5-1  phi1-5-2  phi1-5-3  phi1-5-4
   ...
   phi1-nc-1 ...      ...      ...       ...    phi1-nc-(nc-1) /

================================================================================

   phi1-i-j   - phi1 coefficient for the ith and jth
                components of the mixture;
   nc         - number of components given by the COMPS keyword.
```

BICX2

Depending on the modelling options, the simulator calculates the binary interaction coefficient, τ, for the ith and jth component using the following equation

$$\tau = \tau_0 + \tau_1 T + \tau_2 T^2 + \tau_3 T^3 + \left(\varphi_0 + \varphi_1 T + \varphi_2 T^2 + \varphi_3 T^3\right) \frac{1}{2} \frac{x_{(i)} - x_{(j)}}{x_{(i)} + x_{(j)}}$$

The keyword BICX2 defines the quantities φ_2 for all binary interactions in the fluid.

BICX2 syntax

```
-- within the EOS-ENDEOS brackets

BICX2
   phi2-2-1
   phi2-3-1  phi2-3-2
   phi2-4-1  phi2-4-2  phi2-4-3
   phi2-5-1  phi2-5-2  phi2-5-3  phi2-5-4
   ...
   phi2-nc-1 ...      ...      ...       ...    phi2-nc-(nc-1) /

================================================================================

   phi2-i-j   - phi2 coefficient for the ith and jth
                components of the mixture;
   nc         - number of components given by the COMPS keyword.
```

BICX3

Depending on the modelling options, the simulator calculates the binary interaction coefficient, τ, for the ith and jth component using the following equation

$$\tau = \tau_0 + \tau_1 T + \tau_2 T^2 + \tau_3 T^3 + \left(\varphi_0 + \varphi_1 T + \varphi_2 T^2 + \varphi_3 T^3\right) \frac{1}{2} \frac{x_{(i)} - x_{(j)}}{x_{(i)} + x_{(j)}}$$

The keyword BICX3 defines the quantities φ_3 for all binary interactions in the fluid.

BICX3 syntax

```
-- within the EOS-ENDEOS brackets

BICX3
   phi3-2-1
   phi3-3-1  phi3-3-2
   phi3-4-1  phi3-4-2  phi3-4-3
   phi3-5-1  phi3-5-2  phi3-5-3  phi3-5-4
   ...
   phi3-nc-1 ...      ...      ...       ...    phi3-nc-(nc-1) /

================================================================================

   phi3-i-j   - phi3 coefficient for the ith and jth
                components of the mixture;
   nc         - number of components given by the COMPS keyword.
```

CALVAL

Defines the calorific values for every component of reservoir fluid for respective EOSNUM number. These values are used in the gas quality calculations.

CALVAL syntax

```
-- within the EOS-ENDEOS brackets

CALVAL
   cv(1)  cv(2)  cv(3) ... cv(nc) /

================================================================================

   cv(i)    - calorific value for every component of reservoir fluid
              (i=1,...,nc). By default all cv(i)=0.
   nc       - number of components given by the COMPS keyword.
```

CHARACT

Characterises the fluid component using data on its molar weight, specific gravity, etc. The keyword can be used in addition to LIBCOMPS to characterise hydrocarbon groups not defined in the EoS library.

CHARACT syntax

```
-- within the EOS-ENDEOS brackets

CHARACT
   ID1  Name1  MW1  SpecGrav1  Tb1  Kw1  Tref1  CritProp1  AcfCorr1 /
   ID2  Name2  MW2  SpecGrav2  Tb2  Kw2  Tref2  CritProp2  AcfCorr2 /
   ID3  Name3  MW3  SpecGrav3  Tb3  Kw3  Tref3  CritProp3  AcfCorr3 /
   ...
/

================================================================================

   ID#       - serial number of the fluid component
   Name#     - character name of the component
   MW#       - molar weight
   SpecGrav# - specific gravity
   Tb#       - boiling temperature and normal pressure [K]
   Kw#       - Watson characterisation factor
   Tref#     - reference temperature
   CritProp# - correlations used for the critical properties.
               Possible values:
               K - Kesler-Lee model (default)
               R - Riazi-Daubert model
   AcfCorr#  - correlations used for the acentric factor.
               Possible values:
               K - Kesler-Lee model (default)
               E - Edmister model

   NOTE: It is sufficient to specify just 2 parameters of MW, SpecGrav, Tb,
         and Tref to characterise the fluid.
```

CNAMES

Defines the names of the EoS components. Every name is not longer than 8-byte character.

CNAMES syntax

```
-- within the EOS-ENDEOS brackets

CNAMES
   name(1)  name(2)  name(3) ... name(nc) /

================================================================================

   name(i)  - name of the ith component (i=1,...,nc). The name is an 8-byte
              character;
   nc       - number of components given by the COMPS keyword.
```

CO2SOL

Enables CO_2 solubility in the water phase.

CO2SOL syntax

```
-- in RUNSPEC section

CO2SOL
```

COMPS

Switches the program to the COMPS EOS module.

COMPS syntax

```
-- in section RUNSPEC

COMPS
   nc /

================================================================================

   nc   - number of the components characterized with the cubic equation of
          state.
```

COMPVD

Defines the vapor and liquid composition changes with depth. The depth values must increase down the column. See also the EQUIL and ZMFVD keywords.

COMPVD syntax

```
-- within EQL-ENDEQL brackets

COMPVD
  depth1  cmf1(1)  cmf1(2) cmf1(3) ... cmf1(nc)  phase1 /
  depth2  cmf2(1)  cmf2(2) cmf2(3) ... cmf2(nc)  phase2 /
  depth3  cmf3(1)  cmf3(2) cmf3(3) ... cmf3(nc)  phase3 /
  ...
/

================================================================================

     depth#   - depth;
     cmf#(i)  - molar fraction of the ith component in oil or gas. The phase
                is defined by the pind# flag;
     phase#   - phase for which the compositin is given
                  if 0 then cmf# is the vapor composition;
                  if 1 then cmf# is the liquid composition.
     nc       - number of components given by the COMPS keyword.
```

CXPARAM

Parameters of the COMPS module.

CXPARAM syntax

```
-- within EOS-ENDEOS brackets

CXPARAM
  method1 dens light heavy maxsigma method2 /

================================================================================

     method1 - abbreviation for the method applied for determining the phase
               type, i.e. liquid or gas, in single-phase equilibria. Should be
               one of

               VCRIT - the pseudo-critical volume is used. If the fluid volume
               is less than the pseudo-critical volume then the phase is liquid,
               and it is gas otherwise if phase volume is larger than pseudo-
               critical volume (default option).

               DENSITY - if the fluid density is less than the 2nd argument
               (dens) then it is gas, and is liquid otherwise.

               LIGHT - if the concentration of the first component is larger
               than the 3rd argument, then the phase is labled as gas,
               otherwise liqiud.

               HEAVY - if the concentration of the last component is larger
               than the 4th argument, then the phase is labled as liquid,
               otherwisee gas.

               LIGHT-I - if the concentration of the first component is larger
               than the 3rd argument, then the phase is labled as liquid,
               otherwise gas.

               HEAVY-I - if the concentration of the last component is larger
               than the 4th argument, then the phase is labled as gas,
               otherwisee liquid.

     dens -    the density threshhold for the DENSITY method (default is
               300 kg/m3).
     light -   the concentration threshhold for the LIGHT and LIGHT-I methods
               (default is 0.5).
     heavy -   the concentration threshhold for the HEAVY and HEAVY-I methods
               (default is 0.1).

     method2 - abbreviation for the method applied for determining the phase
               type, i.e. liquid or gas, in two-phase equilibria. Should be
```

```
45              one of
46
47              DENSITY - the phase with the largest density is liquid.
48              The other phase is gas.
49
50              LIGHT - the phase with the largest concentration of the first
51              component is gas. The other phase is liquid.
52
53              HEAVY - the phase with the largest concentration of the last
54              component is liquid. The other phase is gas.
55
56              LIGHT-I - the phase with the largest concentration of the first
57              component is liquid. The other phase is gas.
58
59              HEAVY-I - the phase with the largest concentration of the last
60              component is gas. The other phase is liquid.
```

DENSITY

Defines the densities of oil, water and gas at stock-tank conditions.

DENSITY syntax

```
-- within EOS-ENDEOS brackets

DENSITY
  denoil  denwat  dengas /

================================================================================

     denoil - oil density at stock-tank conditions;
     denwat - water density at stock-tank conditions;
     dengas - gas density at stock-tank conditions.
```

DIFFAGAS

Defines the activity corrected diffusion coefficients for the gas phase. The simulation of molecular diffusion is enabled by the DIFFCHEM keyword.

DIFFAGAS syntax

```
-- within EOS-ENDEOS brackets

DIFFAGAS
   dc(1)   dc(2)   ... dc(nc) /

================================================================================

   dc(i)  - activity corrected gas diffusion coefficient for the ith
            component. Here, nc is the total number of the fluid components.

            Units:   M2/DAY   (METRIC, DEFUNITS)
                     FT2/DAY  (FIELD)
                     CM2/DAY  (LAB)
```

DIFFAOIL

Defines the activity corrected diffusion coefficients for the oil phase. The simulation of molecular diffusion is enabled by the DIFFCHEM keyword.

DIFFAOIL syntax

```
-- within EOS-ENDEOS brackets

DIFFAOIL
   dc(1)   dc(2)   ... dc(nc) /

================================================================================

   dc(i)  - activity corrected oil diffusion coefficient for the ith
            component. Here, nc is the total number of the fluid components.

            Units:   M2/DAY   (METRIC, DEFUNITS)
                     FT2/DAY  (FIELD)
                     CM2/DAY  (LAB)
```

EOSTATE

Defines the EoS type. If the keyword is specified outside the EOS-ENDEOS brackets then the default EoS type is changed. If it is placed within the EOS-ENDEOS brackets then it influences EoS only for the corresponding EOSNUM region. Within EOS-ENDEOS brackets the EOS keyword can be used as synonym to EOSTATE. The default EoS is BR3.

EOSTATE syntax

```
-- in the RUNSPEC or PROPS section

EOSTATE
  eosname /

================================================================================

   eosname  -  abbreviation of the equation of state (default is BR3). Possible
               other values are
               PR3 - the three-parametric Peng-Robinson EoS (with volume shift);
               SRK3 - the three-parametris Soave-Redlich-Kwong EoS (with volume
               shift).
```

EQUALEOS

Equals parameters of the EoS for particular components. If the component name specified by EQUALEOS is equal to the thermodynamic system name then the simulator changes the corresponding EoS parameter for that component.

EQUALEOS syntax

```
-- within the EOS-ENDEOS brackets

EQUALEOS
   cname1  parname1  value1 /
   cname2  parname2  value2 /
   cname3  parname3  value3 /
   ...
/


================================================================================

   cname#    - component name;
   parname#  - abbreviation of the EoS parameter. Possible values of the
               parname# are ACF, MW, PARACHOR, PCRIT, ...;
   value#    - the parameter value.
```

EQUIL

Defines parameters of initial equilibration. See also the COMPVD, ZMFVD, TEMPVD, and TEMPCVD keywords.

EQUIL syntax

```
-- within EQL-ENDEQL brackets

EQUIL
   datum  pdatum  woc  pcow   goc  pcog /

================================================================================

     datum    - datum depth;
     pdatum   - pressure at the datum depth;
     woc      - water-oil contact depth;
     pcow     - capillary pressure at water-oil contact;
     goc      - gas-oil contact depth;
     pcog     - capillary pressure at gas-oil contact.
```

FTDOIL

Enables the following correction of the oil density

$$\rho_o := \rho_o + (a_0 + a_1 T + a_2 T^2 + a_3 T^3)(P - P_{ref})$$

where T units are °C. If SW EoS is used, then the parameters P_{ref} and a_i, $i = 0, \ldots, 3$ are tabulated in the simulator to reproduce the properties of brine. The default values of P_{ref} and a_i, $i = 0, \ldots, 3$ are recommended when using SW section of the EoS library.

FTDOIL syntax

```
-- within the EOS-ENDEOS brackets

FTDOIL
   Pref  a0 a1 a2 a3 /

================================================================================

   Pref   - reference pressure;
   ai     - coefficients. i=0,...,3.
```

IKRG

Specifies the IKRG property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

IKRG syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

IKRG
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

IKRG
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

IKRG
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

IKRO

Specifies the IKRO property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

IKRO syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

IKRO
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

IKRO
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

IKRO
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

IPCG

Specifies the IPCG property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

IPCG syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

IPCG
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

IPCG
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

IPCG
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

IPCW

Specifies the IPCW property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

IPCW syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

IPCW
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

IPCW
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

IPCW
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ISGCR

Specifies the ISGCR property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ISGCR syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ISGCR
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ISGCR
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ISGCR
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ISGL

Specifies the ISGL property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ISGL syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ISGL
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ISGL
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ISGL
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ISGLPC

Specifies the ISGLPC property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ISGLPC syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ISGLPC
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ISGLPC
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ISGLPC
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ISGU

Specifies the ISGU property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ISGU syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ISGU
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ISGU
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ISGU
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ISOGCR

Specifies the ISOGCR property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ISOGCR syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ISOGCR
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ISOGCR
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ISOGCR
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ISOWCR

Specifies the ISOWCR property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ISOWCR syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ISOWCR
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ISOWCR
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ISOWCR
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ISOWL

Specifies the ISOWL property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ISOWL syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ISOWL
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ISOWL
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ISOWL
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ISOWU

Specifies the ISOWU property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ISOWU syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ISOWU
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ISOWU
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ISOWU
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ISWCR

Specifies the ISWCR property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ISWCR syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ISWCR
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ISWCR
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ISWCR
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ISWL

Specifies the ISWL property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ISWL syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ISWL
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ISWL
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ISWL
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ISWLPC

Specifies the ISWLPC property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ISWLPC syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ISWLPC
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ISWLPC
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ISWLPC
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ISWU

Specifies the ISWU property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ISWU syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ISWU
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ISWU
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ISWU
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ISWUPC

Specifies the ISWUPC property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ISWUPC syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ISWUPC
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ISWUPC
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ISWUPC
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

KRG

Specifies the KRG property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

KRG syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

KRG
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

KRG
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

KRG
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

KRO

Specifies the KRO property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

KRO syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

KRO
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

KRO
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

KRO
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

KRW

Specifies the KRW property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

KRW syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

KRW
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

KRW
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

KRW
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

KVIG0

The initial guess of K-values for the stability and phase-split algorithms is internally calculated as

$$K = \left(K_0 + \frac{K_P}{P} \right) \exp\left(K_T \left(1 - \frac{T_c}{T} \right) \right)$$

The KVIG0 keyword allows for changing K_0 for each fluid component. By the default option, the Wilson's formula is used with PR3 and SRK3 EoS ($K_0 = 0$, $K_P = P_c$ and $K_T = 5.372697(1 + \omega)$).

KVIG0 syntax

```
-- within the EOS-ENDEOS brackets

KVIG0
   kvig0(1)  kvig0(2)  kvig0(3) ... kvig0(nc) /

================================================================================

   kvig0(i) - parameter kvig0 for the ith component (i=1,...,nc);
   nc       - number of components given by the COMPS keyword.
```

KVIGP

The initial guess of K-values for the stability and phase-split algorithms is internally calculated as

$$K = \left(K_0 + \frac{K_P}{P}\right) \exp\left(K_T \left(1 - \frac{T_c}{T}\right)\right)$$

The KVIGP keyword allows for changing K_P for each fluid component. By the default option, the Wilson's formula is used with PR3 and SRK3 EoS ($K_0 = 0$, $K_P = P_c$ and $K_T = 5.372697(1 + \omega)$).

KVIGP syntax

```
-- within the EOS-ENDEOS brackets

KVIGP
   kvigp(1)  kvigp(2)  kvigp(3) ... kvigp(nc) /

================================================================================

   kvigp(i) - parameter kvigp for the ith component (i=1,...,nc) [bar in METRIC
              units];
   nc       - number of components given by the COMPS keyword.
```

KVIGT

The initial guess of K-values for the stability and phase-split algorithms is internally calculated as

$$K = \left(K_0 + \frac{K_P}{P}\right) \exp\left(K_T \left(1 - \frac{T_c}{T}\right)\right)$$

The KVIGT keyword allows for changing K_T for each fluid component. By the default option, the Wilson's formula is used with PR3 and SRK3 EoS ($K_0 = 0$, $K_P = P_c$ and $K_T = 5.372697(1 + \omega)$).

KVIGT syntax

```
-- within the EOS-ENDEOS brackets

KVIGT
   kvigt(1)  kvigt(2)  kvigt(3) ... kvigt(nc) /

================================================================================

   kvigt(i) - parameter kvigp for the ith component (i=1,...,nc) ;
   nc       - number of components given by the COMPS keyword.
```

LBCCOEF

Changes coefficients of the Lorenz-Bray-Clark viscosity correlation.

LBCCOEF syntax

```
-- within the EOS-ENDEOS brackets

LBCCOEF
   coef1 coef2 coef3 coef4 coef5 /

================================================================================

   coef#  - coefficients of the Lorenz-Bray-Clark viscosity correlation. The
            default values are
               coef1=0.1023;
               coef2=0.023364;
               coef3=0.058533;
               coef4=-0.040758;
               coef5=0.0093324.
```

LIBCOMPS

Loads the components and binary interaction coefficients from the library for the built-in library of components (see section 7.3). The EoS coefficients depend on the EoS defined by the EOSTATE keyword. Particular components can be loaded from the library and then modified by the OMEGAA, OMEGAB or other keywords. In addition, the fluid components can be characterized by the CHARACT keyword.

LIBCOMPS syntax

```
-- within the EOS-ENDEOS brackets

LIBCOMPS
   name(1)  name(2)  name(3) ... name(nc) /

================================================================================

   name(i)  - name of the ith component (i=1,...,nc) to be loaded from the
              built-in library of components. A name could be defaulted with 1*,
              then the corresponding component and binary interaction
              coefficients are expected to be characterized by user;
   nc       - number of components given by the COMPS keyword.
```

LOADBIC

Characterizes a new binary interaction coefficient to be added to the build-in library of components and binary interaction coefficients.

LOADBIC syntax

```
-- in the PROPS section

LOADBIC
   property1  value1 /
   property2  value2 /
   property3  value3 /
...                  /
/

================================================================================

   property#   - a property characterizing the binary interaction coefficient.
                 Should be one of NAME1, NAME2, EOSTATE, BIC0, BIC1, BIC2.
   value#      - value of the property.


NOTE: The binary interaction coefficinet is calculated as follows

           BIC=BIC0+BIC1*TEMPC+BIC2*TEMPC^2

      where TEMPC is temperature (degee Celcius).
```

LOADCMP

Characterizes a new component to be added to the build-in library of components and binary interaction coefficients.

LOADCMP syntax

```
-- in the PROPS section

LOADCMP
   property1  value1 /
   property2  value2 /
   property3  value3 /
...                  /
/

================================================================================

   property#   - a property characterizing the component. Should be one of
                 NAME, EOSTATE, MW, PCRIT, TCRIT, ACF, ZCRIT, ZCRITVIS, VCRIT,
                 VCRITVIS, OMEGAA, OMEGAB, OMEGAC, ZCRITA, PSI, SSHIFT,
                 PARACHOR, SPECH0, SPECH1, SPECH2, SPECH3, SPECH4, SPECH5,
                 SPECG.
   value#      - value of the property.
```

MW

Defines the molar weights for the EoS components.

MW syntax

```
-- within the EOS-ENDEOS brackets

MW
   mw(1)  mw(2)  mw(3) ... mw(nc) /

================================================================================

   mw(i)  - molar weight for the ith component (i=1,...,nc);
   nc     - number of components given by the COMPS keyword.
```

NCOMPS

Checks that the number of EOS components is equal to that specified by the COMPS keyword. If the number of components in NCOMPS is not equal to that in COMPS then the simulation is terminated with error.

NCOMPS syntax

```
-- within the EOS-ENDEOS brackets

NCOMPS
   kc /

================================================================================

   kc     - number of EoS components. If kc is not equal to nc then the
            simulation is terminated with error;
   nc     - number of components given by the COMPS keyword.
```

OMEGAA

Defines the Ω_a coefficients for the EoS components.

OMEGAA syntax

```
-- within the EOS-ENDEOS brackets

OMEGAA
   omegaa(1)  omegaa(2)  omegaa(3) ... omegaa(nc) /

================================================================================

   omegaa(i) - the Omega_a parameter for the ith component (i=1,...,nc);
   nc        - number of components given by the COMPS keyword.
```

OMEGAB

Defines the Ω_b coefficients for the EoS components.

OMEGAB syntax

```
-- within the EOS-ENDEOS brackets

OMEGAB
   omegab(1)  omegab(2)  omegab(3) ... omegab(nc) /

================================================================================

   omegab(i) - the Omega_b parameter for the ith component (i=1,...,nc);
   nc        - number of components given by the COMPS keyword.
```

OMEGAC

Defines the Ω_c coefficients for the EoS components.

OMEGAC syntax

```
-- within the EOS-ENDEOS brackets

OMEGAC
   omegac(1)  omegac(2)  omegac(3) ... omegac(nc) /

================================================================================

   omegac(i) - the Omega_c parameter for the ith component (i=1,...,nc);
   nc        - number of components given by the COMPS keyword.
```

OPTCX

Additional options for the COMPS module.

OPTCX syntax

```
-- in the PROPS, INIT or SCHEDULE section

OPTCX
  opt1 opt2 opt3 ... /

================================================================================

     opt#   - if 1 the corresponding option is enabled, if 0 the option is
              disabled.

     opt1=1 - the water properties of the PVTW format are internally converted
              to the WATERTAB format (default 0).
     opt3=1 - if the binary interaction coefficient is not found in the
              build-in library then it is set to 0.0. (default 1).
     opt8=1 - if the stability algorithm indicates that the 1-phase state is
              unstable, then the initial guess for K-values are used to specify
              the initial composition of the phases for the phase-split
              algorithm. Otherwise (i.e. opt8=0). The output of the stability
              algorithm is used as the initial gues for the phase-split
              algorithm. By default opt8=0.
```

PARACHOR

Defines parachors for the EoS components.

PARACHOR syntax

```
-- within the EOS-ENDEOS brackets

PARACHOR
   parachor(1)  parachor(2)  parachor(3) ... parachor(nc) /

================================================================================

   parachor(i) - the parachor for the ith component (i=1,...,nc);
   nc          - number of components given by the COMPS keyword.
```

PASSUT

Defines the parameters of the ideal gas. The simulator looks through the thermodynamic system and if a name of component matches that in a data line in the table then the simulator uses the Passut & Danner (1972) coefficients for ideal gas state enthalpy calculation. The coefficients should be given exactly in the same form and units as in the original paper.

PASSUT syntax

```
-- within the EOS-ENDEOS brackets

PASSUT
   name-1   spechA-1  spechB-1 spechC-1 spechD-1 spechE-1 spechF-1 spechG-1 /
   name-2   spechA-2  spechB-2 spechC-2 spechD-2 spechE-2 spechF-2 spechG-2 /
   name-3   spechA-3  spechB-3 spechC-3 spechD-3 spechE-3 spechF-3 spechG-3 /
   ...
/

================================================================================

   name-#    - component name;
   spechI-#  - the Ith coefficient of the Passut-Danner correlation for the
               ideal-state enthalpy. The coefficients must be specifyed in
               in the same units as in the original paper by Passut & Danner
               (1972).

--------------------------------------------------------------------------------

Example:

PASSUT
 H2O -2.46342  0.457392  -0.052512   0.064594  -0.202759   0.236310  -0.339830 /
 CO2  4.77805  0.114433   0.101132  -0.026494   0.034706  -0.013140   0.343357 /
 C1  -5.58114  0.564834  -0.282973   0.417399  -1.525576   1.958857  -0.623373 /
/
```

PCG

Specifies the PCG property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

PCG syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

PCG
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

PCG
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

PCG
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

PCRIT

Defines the critical pressures for the EoS components.

PCRIT syntax

```
-- within the EOS-ENDEOS brackets

PCRIT
   pcrit(1)  pcrit(2)  pcrit(3) ... pcrit(nc) /

================================================================================

   pcrit(i) - the critical pressure of the ith component (i=1,...,nc);
   nc       - number of components given by the COMPS keyword.
```

PRCORR

Invokes the corrected specification of the ψ coefficients for the Peng-Robinson equation of state.

PRCORR syntax

```
-- within the EOS-ENDEOS brackets

PRCORR
```

PSI

Defines the ψ coefficients for the EoS components.

PSI syntax

```
-- within the EOS-ENDEOS brackets

PSI
   psi(1)  psi(2)  psi(3) ... psi(nc) /

================================================================================

   psi(i)    - the psi parameter for the ith component (i=1,...,nc);
   nc        - number of components given by the COMPS keyword.
```

PVTW

Defines PVT properties of water. See also the WATERTAB keyword.

PVTW syntax

```
-- within EOS-ENDEOS brackets

PVTW
   pref  bw  comprw  mw  viscmw /

================================================================================

    pref   - reference presssure;
    bw     - water formation volume factor at the reference pressure;
    comprw - water compressibility at the reference pressure;
    mw     - water viscosity at the reference pressure;
    viscmw - water 'viscosibility' at the reference pressure.
```

RPTMMP

Reports to the LOG-file an estimation of the "true" and "dynamic" minimum miscibility pressure for the reservoir fluid. Optionally, the pressure can be saved to a variable created with the AUXVAR keyword.

RPTMMP syntax

```
-- within EOS-ENDEOS brackets

RPTMMP
   true  z(1)  ... z(nc) dynamic /

================================================================================

      true - the variable name used to store the true MMP;
         z(i) - the bulk composition of the fluid to mix with;
   dynamic - the variable name used to store the dynamic MMP;
        nc - number of components given by the COMPS keyword.
```

RPTPBUB

Reports to the LOG-file an estimation of the bubble pressure for the reservoir fluid. Optionally, the pressure can be saved to a variable created with the AUXVAR keyword.

RPTPBUB syntax

```
-- within EOS-ENDEOS brackets

RPTPBUB
   varname /

================================================================================

      varname - the variable name used to store the bubble pressure.
```

RSMOL

Defines the NaCl molality of the reservoir brine. This keyword is mandatory when using SW EoS. Alternatively, the salt molality can be specified by the keyword RXSM or RXS.

RSMOL syntax

```
-- within the EOS-ENDEOS brackets

RSMOL
   smol /

================================================================================

            smol - NaCl molality of the reservoir brine. The
                   units of the NaCl molality are mol/kg.
```

RTEMP

Defines initial reservoir temperature for the corresponding EOSNUM. This temperature is used as a default value for the corresponding EOSNUM region.

RTEMP syntax

```
-- within the EOS-ENDEOS brackets

RTEMP
  restemp /

================================================================================

   restemp  - initial reservoir temperature (deg Celcius/deg Fahrenheit
              depending on units).
```

RXS

Defines the NaCl molality of the reservoir brine. This keyword is mandatory when using SW EoS. Alternatively, the salt molality can be specified by the keyword RXSM or RSMOL.

RXS syntax

```
-- within the EOS-ENDEOS brackets

RXS
   xs /

================================================================================

   xs - molar concentration of NaCl in the reservoir brine.
```

RXSM

Defines the NaCl molality of the reservoir brine. This keyword is mandatory when using SW EoS. Alternatively, the salt molality can be specified by the keyword RXS or RSMOL.

RXSM syntax

```
-- within the EOS-ENDEOS brackets

RXSM
   xsm /

================================================================================

   xsm - mass concentration of NaCl in the reservoir brine.
```

SAVEPVO

Requests the PVO-file to be exported for the corresponding EOSNUM region.

SAVEPVO syntax

```
-- in the PROPS section

SAVEPVO
   eosnum  filename  prop1 prop2 prop3 prop4 ... /

================================================================================

   eosnum     - the EOSNUM region number for which the file is requested
                (default 1);
   filename   - name of the file for output. Must be not empty;
   prop#      - property to be reported in the file. If the report list is
                empty (i.e. all prop# are not specified) then the defaul report
                list is used. The default report listconsists of RTEMP, EOSTATE,
                NCOMPS, PRCORR, CNAMES, MW, OMEGAA, OMEGAB, TCRIT, PCRIT, VCRIT,
                ZCRIT, SSHIFT, ACF, BIC, PARACHOR, VCRITVIS, ZCRITVIS, LBCCOEF,
                ZI.

   NOTE: prop1 can be set to ADD. In this case, other prop# are added to the
         default report list.
```

SGCR

Specifies the SGCR property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SGCR syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SGCR
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SGCR
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SGCR
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SGFN

Defines gas saturation functions.

SGFN syntax

```
-- within SAT-ENDSAT brackets

SGFN
   sgas1  krg1  pcog1 /
   sgas2  krg2  pcog2 /
   sgas3  krg3  pcog3 /
   ...
/

================================================================================

     sgas# - gas saturation;
     krg#  - gas relative permeability;
     pcog# - oil-gas capillary pressure.
```

SGL

Specifies the SGL property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SGL syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SGL
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SGL
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SGL
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SGLPC

Specifies the SGLPC property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SGLPC syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SGLPC
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SGLPC
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SGLPC
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SGOF

Defines gas-oil saturation functions versus gas saturation.

SGOF syntax

```
-- within SAT-ENDSAT brackets

SGOF
   sgas1  krg1  krog1  pcog1 /
   sgas2  krg2  krog2  pcog2 /
   sgas3  krg3  krog3  pcog3 /
   ...
/

================================================================================

     sgas# - gas saturation;
     krg#  - gas relative permeability;
     krog# - oil relative permeability;
     pcog# - oil-gas capillary pressure.
```

SGU

Specifies the SGU property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SGU syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SGU
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SGU
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SGU
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SLGOF

Defines gas-oil saturation functions versus liquid saturation.

SLGOF syntax

```
-- within SAT-ENDSAT brackets

SLGOF
   soil1  krg1  krog1  pcog1 /
   soil2  krg2  krog2  pcog2 /
   soil3  krg3  krog3  pcog3 /
   ...
/

================================================================================

     soil# - oil saturation;
     krg#  - gas relative permeability;
     krog# - oil relative permeability;
     pcog# - oil-gas capillary pressure.
```

SLGOF2

Defines gas-oil saturation functions versus liquid saturation.

SLGOF2 syntax

```
-- within SAT-ENDSAT brackets

SLGOF2
   soil1  krog1  krg1  pcog1 /
   soil2  krog2  krg2  pcog2 /
   soil3  krog3  krg3  pcog3 /
   ...
/

================================================================================

     soil# - oil saturation;
     krog# - oil relative permeability;
     krg#  - gas relative permeability;
     pcog# - oil-gas capillary pressure.
```

SOF2

Defines oil saturation functions (2-phase case).

SOF2 syntax

```
-- within SAT-ENDSAT brackets

SOF2
   soil1  krow1  /
   soil2  krow2  /
   soil3  krow3  /
   ...
/

================================================================================

     soil# - oil saturation;
     krow# - oil relative permeability.
```

SOF3

Defines oil saturation functions (3-phase case).

SOF3 syntax

```
-- within SAT-ENDSAT brackets

SOF3
   soil1  krow1  krog1 /
   soil2  krow2  krog2 /
   soil3  krow3  krog3 /
   ...
/

================================================================================

     soil# - oil saturation;
     krow# - oil relative permeability for oil-water flows;
     krog# - oil relative permeability for oil-gas flows.
```

SOGCR

Specifies the SOGCR property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SOGCR syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SOGCR
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SOGCR
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SOGCR
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SOLUBILI

Load PVT table for the water phase in the case of CO2SOL option. The table characterizes the properties of the water phase placed in equilibrium with the CO_2 phase. This table can be exported from the GASSTORE module using the GSTOSOLU keyword.

SOLUBILI syntax

```
-- within EOS-ENDEOS brackets

SOLUBILI
  pres1   Rs1   FVF1  visc1  compr1  /
  pres2   Rs2   FVF2  visc2  compr2  /
  pres3   Rs3   FVF3  visc3  compr3  /
  ...
/

================================================================================

      pres#  - pressure (must increase down the column);
      Rs#    - dissolved CO2 solution ratio;
      FVF#   - formation volume factor of the water phase;
      visc#  - the water viscosity;
            compr# - the water compressibility.
```

SOWCR

Specifies the SOWCR property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SOWCR syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SOWCR
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SOWCR
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SOWCR
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SOWL

Specifies the SOWL property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SOWL syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SOWL
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SOWL
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SOWL
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SOWU

Specifies the SOWU property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SOWU syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SOWU
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SOWU
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SOWU
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SSHIFT

The simulator calculates the dimensionless volume shift, ξ, using the equation

$$\xi(T) = \xi_0 + \xi_1 T + \xi_2 T^2 + \xi_3 T^3$$

where the temperature units are °C. The keyword SSHIFT allows for entering the parameter ξ_0 for every fluid component. This keyword has a synonym SSHIFT0.

SSHIFT syntax

```
-- within the EOS-ENDEOS brackets

SSHIFT
   xi0(1)  xi0(2)  xi0(3) ... xi0(nc) /

================================================================================

   xi0(i)    - the dimensionless volume shift parameter for the ith component
               (i=1,...,nc);
   nc        - number of components given by the COMPS keyword.
```

SSHIFT0

The simulator calculates the dimensionless volume shift, ξ, using the equation

$$\xi(T) = \xi_0 + \xi_1 T + \xi_2 T^2 + \xi_3 T^3$$

where the temperature units are °C. The keyword SSHIFT0 allows for entering the parameter ξ_0 for every fluid component.

SSHIFT0 syntax

```
-- within the EOS-ENDEOS brackets

SSHIFT0
   xi0(1)  xi0(2)  xi0(3) ... xi0(nc) /

================================================================================

   xi0(i)    - the dimensionless volume shift parameter for the ith component
               (i=1,...,nc);
   nc        - number of components given by the COMPS keyword.
```

SSHIFT1

The simulator calculates the dimensionless volume shift, ξ, using the equation

$$\xi(T) = \xi_0 + \xi_1 T + \xi_2 T^2 + \xi_3 T^3$$

where the temperature units are °C. The keyword SSHIFT1 allows for entering the parameter ξ_1 for every fluid component.

SSHIFT1 syntax

```
-- within the EOS-ENDEOS brackets

SSHIFT1
   xi1(1)  xi1(2)  xi1(3) ... xi1(nc) /

================================================================================

   xi1(i)    - the dimensionless volume shift parameter for the ith component
               (i=1,...,nc);
   nc        - number of components given by the COMPS keyword.
```

SSHIFT2

The simulator calculates the dimensionless volume shift, ξ, using the equation

$$\xi(T) = \xi_0 + \xi_1 T + \xi_2 T^2 + \xi_3 T^3$$

where the temperature units are °C. The keyword SSHIFT2 allows for entering the parameter ξ_2 for every fluid component.

SSHIFT2 syntax

```
-- within the EOS-ENDEOS brackets

SSHIFT2
   xi2(1)  xi2(2)  xi2(3) ... xi2(nc) /

================================================================================

   xi2(i)    - the dimensionless volume shift parameter for the ith component
               (i=1,...,nc);
   nc        - number of components given by the COMPS keyword.
```

SSHIFT3

The simulator calculates the dimensionless volume shift, ξ, using the equation

$$\xi(T) = \xi_0 + \xi_1 T + \xi_2 T^2 + \xi_3 T^3$$

where the temperature units are °C. The keyword SSHIFT3 allows for entering the parameter ξ_3 for every fluid component.

SSHIFT3 syntax

```
-- within the EOS-ENDEOS brackets

SSHIFT3
   xi3(1)  xi3(2)  xi3(3) ... xi3(nc) /

================================================================================

   xi3(i)    - the dimensionless volume shift parameter for the ith component
               (i=1,...,nc);
   nc        - number of components given by the COMPS keyword.
```

STONE1

Switches three-phase oil relative permeability calculation to the 1st Stone model. This keyword influences only the current saturation functions region (see the SAT-ENDSAT keyword).

STONE1 syntax

```
-- within SAT-ENDSAT brackets

STONE1
```

STONE2

Switches three-phase oil relative permeability calculation to the 2st Stone model. This keyword influences only the current saturation functions region (see the SAT-ENDSAT keyword).

STONE2 syntax

```
-- within SAT-ENDSAT brackets

STONE2
```

SWAT

Specifies the SWAT property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SWAT syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SWAT
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SWAT
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SWAT
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SWFIXT

When using the SW section of the EoS library, entering this keyword forces the simulator to calculate binary interaction coefficients and volume shifts at constant temperature specified by the RTEMP. The interaction coefficients and volume shifts remain constant even if tempeature chenges during simulation.

SWFIXT syntax

```
-- within the EOS-ENDEOS brackets

SWFIXT
```

SWCR

Specifies the SWCR property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SWCR syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SWCR
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SWCR
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SWCR
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SWFN

Defines water saturation functions.

SWFN syntax

```
-- within SAT-ENDSAT brackets

SWFN
   swat1  krw1  pcow1 /
   swat2  krw2  pcow2 /
   swat3  krw3  pcow3 /
   ...
/

================================================================================

     swat# - water saturation;
     krw#  - water relative permeability;
     pcow# - water-oil capillary pressure.
```

SWL

Specifies the SWL property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SWL syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SWL
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SWL
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SWL
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SWLPC

Specifies the SWLPC property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SWLPC syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SWLPC
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SWLPC
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SWLPC
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SWOF

Defines water-oil saturation functions.

SWOF syntax

```
-- within SAT-ENDSAT brackets

SWOF
   swat1  krw1  krow1  pcow1 /
   swat2  krw2  krow2  pcow2 /
   swat3  krw3  krow3  pcow3 /
   ...
/

================================================================================

     swat# - water saturation;
     krw#  - water relative permeability;
     kro#  - oil relative permeability;
     pcow# - water-oil capillary pressure.
```

SWORIG

Forces the simulator to use the original correlations of Søreide and Whitson (1992).

SWORIG syntax

```
-- within the EOS-ENDEOS brackets

SWORIG
```

SWU

Specifies the SWU property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SWU syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SWU
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SWU
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SWU
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SWUPC

Specifies the SWUPC property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SWUPC syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SWUPC
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SWUPC
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SWUPC
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

TCRIT

Defines the critical temperatures for the EoS components.

TCRIT syntax

```
-- within the EOS-ENDEOS brackets

TCRIT
   tcrit(1)  tcrit(2)  tcrit(3) ... tcrit(nc) /

================================================================================

   tcrit(i) - the critical temperature of the ith component (i=1,...,nc);
   nc       - number of components given by the COMPS keyword.
```

TEMPCVD

Defines the initial distribution of the temperature (degrees Celsius) versus depth. The depth values must increase down the column. See also the EQUIL keyword.

TEMPCVD syntax

```
-- within EQL-ENDEQL brackets

TEMPCVD
  depth1  tempc1  /
  depth2  tempc2  /
  depth3  tempc3  /
  ...
/

================================================================================

    depth#  - depth;
    tempc#  - temperature (degrees Celcius).
```

TEMPVD

Defines the initial distribution of the temperature (degrees Kelvin) versus depth. The depth values must increase down the column. See also the EQUIL keyword.

TEMPVD syntax

```
-- within EQL-ENDEQL brackets

TEMPVD
  depth1  temp1  /
  depth2  temp2  /
  depth3  temp3  /
  ...
/

================================================================================

     depth#  - depth;
     temp#   - temperature (degrees Kelvin).
```

VCRIT

Defines the critical molar volume for the EoS components.

VCRIT syntax

```
-- within the EOS-ENDEOS brackets

VCRIT
   vcrit(1)  vcrit(2)  vcrit(3) ... vcrit(nc) /

================================================================================

   vcrit(i) - the critical molar volume of the ith component (i=1,...,nc);
   nc       - number of components given by the COMPS keyword.
```

VCRITVIS

Defines the critical molar volumes for viscosity calculation.

VCRITVIS syntax

```
-- within the EOS-ENDEOS brackets

VCRITVIS
   vcritvis(1)  vcritvis(2)  vcritvis(3) ... vcritvis(nc) /

================================================================================

   vcritvis(i) - the critical molar volume of the ith component (i=1,...,nc)
                 for the viscosity calculation;
   nc          - number of components given by the COMPS keyword.
```

VISCREF

Defines the reference pressure for the viscosity function given by the WATVISCT keyword.

VISCREF syntax

```
-- within EOS-ENDEOS brackets

VISCREF
  Pref /

================================================================================

      Pref - reference pressure for the water viscocity as function of
             temperature.
```

VISEXPLT

Allows explicit specification of the gas and oil phase viscosities. This option overrides the default method for predicting phase viscosity.

VISEXPLT syntax

```
-- within the EOS-ENDEOS brackets

VISEXPLT
   visgas  visoil /

================================================================================

    visgas - gas phase viscosity (default: NaN)
    visoil - oil phase viscosity (default: NaN)

    NOTE: For using this option, the 121st flag of the OPTIONS keyword must be
          set to 1.
```

WATDENT

Defines water density dependence on temperature.

WATDENT syntax

```
-- within EOS-ENDEOS brackets

WATDENT
  Tref  Coef1 Coef2 /

================================================================================

      Tref   - reference temperature (default 293.15 K);
      Coef1  - 1st coefficient (default 3e-4 1/K);
      Coef2  - 2nd coefficient (default 3e-6 1/K).
```

WATENTHT

Defines water specific enthalpy dependence on temperature.

WATENTHT syntax

```
-- within EOS-ENDEOS brackets

WATENTHT
  Tref  Coef1 Coef2 /

================================================================================

      Tref   - reference temperature (default 293.15 K);
      Coef1  - 1st coefficient (default 4.2 kJ/kg/K);
      Coef2  - 2nd coefficient (default 0 kJ/kg/K^2).

NOTE:

    Water Specific Enthalpy = Coef1*(T-Tref)+Coef2*(T-Tref)^2
```

WATER

Enables the water phase. This keyword indicates that the simulation contains the water phase.

WATER syntax

```
-- in RUNSPEC section

WATER
```

WATERTAB

Defines PVT properties of water. See also the PVTW keyword.

WATERTAB syntax

```
-- within EOS-ENDEOS brackets

WATERTAB
  pres1   bw1   mw1 /
  pres2   bw2   mw2 /
  pres3   bw3   mw3 /
  ...
/

================================================================================

      pres# - pressure (must increase down the column);
      bg#   - water formation volume factor;
      mg#   - water viscosity.
```

WATVISCT

Defines water viscosity as function of temperature. Before loading this data, the reference pressure for this dependence must be specified by the VISCREF keyword.

WATVISCT syntax

```
-- within EOS-ENDEOS brackets

WATVISCT
  temp1   visc1 /
  temp2   visc2 /
  temp3   visc3 /
  ...
/

================================================================================

      temp# - temperature (must increase down the column);
      visc# - water viscosity.
```

WSTREAM

Defines composition of the injected fluid.

WSTREAM syntax

```
-- in the INIT or SCHEDULE section

WSTREAM
  template1  pres1 tempc1 swat1 zi1(1)  zi1(2)  zi1(3) ... zi1(nc) /
  template2  pres2 tempc2 swat2 zi2(1)  zi2(2)  zi2(3) ... zi2(nc) /
  template3  pres3 tempc3 swat3 zi3(1)  zi3(2)  zi3(3) ... zi3(nc) /
  ...                                                             /
/

================================================================================

   template# - well name template;
   pres#     - pressure in the tank containing the injectected fluid;
   tempc#    - temperature (rel) in the tank containing the injectected fluid;
   swat#     - volume fraction of water in the injected fluid;
   zi#(i)    - initial total molar concentrations (i=1,...,nc) in the
               injected fluid. All zi(i) must sum up to 1;
   nc        - number of components given by the COMPS keyword.
```

ZCRIT

Defines the critical z-factor for the EoS components.

ZCRIT syntax

```
-- within the EOS-ENDEOS brackets

ZCRIT
   zcrit(1)  zcrit(2)  zcrit(3) ... zcrit(nc) /

================================================================================

   zcrit(i) - the critical z-factor of the ith component (i=1,...,nc);
   nc       - number of components given by the COMPS keyword.
```

ZCRITA

Defines the Z_c^* coefficients for the EoS components.

ZCRITA syntax

```
-- within the EOS-ENDEOS brackets

ZCRITA
   zcrita(1)  zcrita(2)  zcrita(3) ... zcrita(nc) /

================================================================================

   zcrita(i) - the Z_c^* parameter for the ith component (i=1,...,nc);
   nc        - number of components given by the COMPS keyword.
```

ZCRITVIS

Defines the critical z-factors for viscosity calculation.

ZCRITVIS syntax

```
-- within the EOS-ENDEOS brackets

ZCRITVIS
   zcritvis(1)  zcritvis(2)  zcritvis(3) ... zcritvis(nc) /

================================================================================

   zcritvis(i) - the critical z-factor of the ith component (i=1,...,nc)
                 for the viscosity calculation;
   nc          - number of components given by the COMPS keyword.
```

ZI

Defines initial fluid composition. This composition is used for initialization parameters in a grid cell with respective EOSNUM number when the composition is not specified by user.

ZI syntax

```
-- within the EOS-ENDEOS brackets

ZI
   zi(1)  zi(2)  zi(3) ... zi(nc) /

================================================================================

   zi(i)    - initial total molar concentrations (i=1,...,nc). All zi(i) must
              sum up to 1;
   nc       - number of components given by the COMPS keyword.
```

ZMFVD

Defines the initial fluid bulk composition versus depth. The depth values must increase down the column. See also the EQUIL and COMPVD keywords.

ZMFVD syntax

```
-- within EQL-ENDEQL brackets

ZMFVD
  depth1  zmf1(1)  zmf1(2) zmf1(3) ... zmf1(nc)  /
  depth2  zmf2(1)  zmf2(2) zmf2(3) ... zmf2(nc)  /
  depth3  zmf3(1)  zmf3(2) zmf3(3) ... zmf3(nc)  /
  ...
/

================================================================================

     depth#   - depth;
     zmf#(i)  - bulk molar fraction of the ith component;
     nc       - number of components given by the COMPS keyword.
```

ZMF$c

Specifies the ZMF$c property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ZMF$c syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ZMF$c
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ZMF$c
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ZMF$c
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

1.11. GASSTORE keywords

The keywords below are available only with the GASSTORE EOS module.

AIR

Enables the air component and the gas phase.

AIR syntax

```
-- in RUNSPEC section

AIR
```

CH4

Enables the CH_4 gas component and the gas phase.

CH4 syntax

```
-- in RUNSPEC section

CH4
```

CO2-SEQ

Enables the CO_2 gas component and the gas phase for modelling underground CO_2 storage

CO2-SEQ syntax

```
-- in RUNSPEC section

CO2-SEQ
```

DENSTCND

Defines gas and water densities at stock tank conditions

DENSTCND syntax

```
-- within EOS-ENDEOS brackets

DENSTCND
  dengas  denwat /

================================================================================

     dengas - oil density at stock-tank conditions;
     denwat - water density at stock-tank conditions.
```

DENTAB

Loads gas density on pressure and temperature (only for CO2-SEQ gas).

DENTAB syntax

```
-- within EOS-ENDEOS, or PVT-ENDPVT brackets, or in the PROPS section

DENTAB
        temp1  temp2  temp3  ...  /
 pres1  den11  den12  den13  ...  /
 pres2  den21  den22  den23  ...  /
 pres3  den31  den32  den33  ...  /
 ...    ...    ...    ...    ...  /
/

================================================================================

     temp#  -  temperature (degree Kelvin);
     pres$  -  pressure;
     den$#  -  density for a given pressure (pres$) and temperature (temp#).
```

DISGAS

Enables the gas component dissolution in the water phase.

DISGAS syntax

```
-- in RUNSPEC section

DISGAS
```

ENTHTAB

Loads gas specific enthalpy on pressure and temperature (only for CO2-SEQ gas).

ENTHTAB syntax

```
-- within EOS-ENDEOS, or PVT-ENDPVT brackets, or in the PROPS section

ENTHTAB
        temp1  temp2  temp3  ...  /
 pres1  enth11 enth12 enth13  ...  /
 pres2  enth21 enth22 enth23  ...  /
 pres3  enth31 enth32 enth33  ...  /
 ...    ...    ...    ...    ...  /
/

================================================================================

     temp#  -  temperature (degree Kelvin);
     pres$  -  pressure;
     enth$# -  specific enthalpy for a given pressure (pres$) and
               temperature (temp#).
```

EQUILGS

Defines parameters of initial equilibration. See also the TEMPVD, TEMPCVD, SSOLVD, XSMVD, XGMVD and YWMVD keywords.

EQUILGS syntax

```
-- within EQL-ENDEQL brackets

EQUILGS
   datum  pdatum  wgc  pcgw /

================================================================================

     datum    - datum depth;
     pdatum   - pressure at the datum depth;
     wgc      - water-gas contact depth;
     pcgw     - capillary pressure at gas-water contact.
```

GASSTORE

Switches the program to GASSTORE EOS module.

GASSTORE syntax

```
-- in RUNSPEC section

GASSTORE
```

GSTOBO

Exports PVT data for black-oil simulations. For the CO2-SEQ option, the keyword must be used after the gas properties are specified using the DENTAB, ENTHTAB and VISTAB keywords. The output data are affected by the WATER, NACL, DISGAS, VAPWAT, CO2-SEQ, CH4, etc. keywords.

GSTOBO syntax

```
-- in PROPS section

GSTOBO
  filename temp xsm smol presst tempst dliqst dgasst opt /
  pres1 pres2 pres3 pres4 ... /

================================================================================

  filename - name of the output file;
  temp     - reservoir temperature (degree Kelvin);
  xsm      - salt mass fraction;
  smol     - molality (mol/kg). This item is used if xsm is not specified;
  presst   - pressure at surface (stock tank) conditions (default 1 atm);
  tempst   - temperature at surface (stock tank) conditions (degree Kelvin)
             (default 293.15 K);
  dliqst   - liquid phase density at surface (stock tank) conditions. If not
             specified by user then dliqst is calculated at presst and
             tempst;
  dgasst   - gas density at surface (stock tank) conditions. If not specified
             by user then dgasst is calculated at presst and tempst;
  opt      - must be T or F. If T then liquid and gas are assumed comletely
             separated at surface conditions (default option). If F then liquid
             and gas are assumed to be in thermodynamic equilibrium at surface
             conditions (gas is dissoled in brine; water vapour is in gas);
  pres#    - reservoir pressures for which the tabular data are exported; Must
             increase in the sequence; Recommended value for pres1 is 1 atm.
```

GSTOSOLU

Exports PVT data for the compositional simulations complicated by the CO2SOL option. The keyword results in exporting the SOLUBILI keyword in a specified external file. For the CO2-SEQ option, the keyword must be used after the gas properties are specified using the DENTAB, ENTHTAB and VISTAB keywords. The output data are affected by the WATER, NACL, DISGAS, VAPWAT, CO2-SEQ, CH4, etc. keywords.

GSTOSOLU syntax

```
-- in PROPS section

GSTOSOLU
  filename temp xsm smol presst tempst dliqst dgasst opt /
  pres1 pres2 pres3 pres4 ... /

================================================================================

  filename - name of the output file;
  temp     - reservoir temperature (degree Kelvin);
  xsm      - salt mass fraction;
  smol     - molality (mol/kg). This item is used if xsm is not specified;
  presst   - pressure at surface (stock tank) conditions (default 1 atm);
  tempst   - temperature at surface (stock tank) conditions (degree Kelvin)
             (default 293.15 K);
  dliqst   - liquid phase density at surface (stock tank) conditions. If not
             specified by user then dliqst is calculated at presst and
             tempst;
  dgasst   - gas density at surface (stock tank) conditions. If not specified
             by user then dgasst is calculated at presst and tempst;
  opt      - must be T or F. If T then liquid and gas are assumed comletely
             separated at surface conditions (default option). If F then liquid
             and gas are assumed to be in thermodynamic equilibrium at surface
             conditions (gas is dissoled in brine; water vapour is in gas);
  pres#    - reservoir pressures for which the tabular data are exported; Must
             increase in the sequence; Recommended value for pres1 is 1 atm.
```

IKRG

Specifies the IKRG property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

IKRG syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

IKRG
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

IKRG
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

IKRG
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

IKRL

Specifies the IKRL property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

IKRL syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

IKRL
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

IKRL
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

IKRL
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

IPCG

Specifies the IPCG property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

IPCG syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

             -- if the LOADING switch is in IJK position --

IPCG
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

IPCG
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

IPCG
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ISGCR

Specifies the ISGCR property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ISGCR syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

             -- if the LOADING switch is in IJK position --

ISGCR
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

             -- if the LOADING switch is in CELLID position --

ISGCR
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ISGCR
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ISGL

Specifies the ISGL property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ISGL syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ISGL
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ISGL
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ISGL
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ISGLPC

Specifies the ISGLPC property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ISGLPC syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ISGLPC
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ISGLPC
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ISGLPC
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ISGU

Specifies the ISGU property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ISGU syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ISGU
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ISGU
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ISGU
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ISLCR

Specifies the ISLCR property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ISLCR syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ISLCR
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ISLCR
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ISLCR
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

ISSCR

Specifies the ISSCR property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

ISSCR syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

ISSCR
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

ISSCR
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ISSCR
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

KRG

Specifies the KRG property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

KRG syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

KRG
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

KRG
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

KRG
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

KRL

Specifies the KRL property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

KRL syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

KRL
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

KRL
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

KRL
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

NACL

Enables the NaCl component and the solid phase. This keyword indicates that the simulation contains NaCl which can form a solid phase or can dissolve in water.

NACL syntax

```
-- in RUNSPEC section

NACL
```

N2

Enables the N_2 gas component and the gas phase.

N2 syntax

```
-- in RUNSPEC section

N2
```

PCG

Specifies the PCG property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

PCG syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

PCG
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

PCG
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

PCG
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SGAS

Specifies the SGAS property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SGAS syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SGAS
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SGAS
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SGAS
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SGCR

Specifies the SGCR property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SGCR syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

             -- if the LOADING switch is in IJK position --

SGCR
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SGCR
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SGCR
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SGL

Specifies the SGL property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SGL syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SGL
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SGL
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SGL
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SGLPC

Specifies the SGLPC property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SGLPC syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SGLPC
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SGLPC
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SGLPC
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SGLF

Defines gas-liquid saturation functions versus gas saturation.

SGLF syntax

```
-- within SAT-ENDSAT brackets

SGLF
   sgas1  krg1  krl1  pclg1 /
   sgas2  krg2  krl2  pclg2 /
   sgas3  krg3  krl3  pclg3 /
   ...
/

================================================================================

     sgas# - gas saturation;
     krg#  - gas relative permeability;
     krl#  - liquid (brine) relative permeability;
     pclg# - liquid-gas capillary pressure.
```

SGU

Specifies the SGU property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SGU syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SGU
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SGU
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SGU
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SLCR

Specifies the SLCR property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SLCR syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SLCR
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SLCR
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SLCR
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SLGF

Defines gas-liquid saturation functions versus liquid saturation.

SLGF syntax

```
-- within SAT-ENDSAT brackets

SLGF
   sliq1  krl1  krg1  pclg1 /
   sliq2  krl2  krg2  pclg2 /
   sliq3  krl3  krg3  pclg3 /
   ...
/

================================================================================

     sliq# - liquid (brine) saturation;
     krl#  - liquid (brine) relative permeability;
     krg#  - gas relative permeability;
     pclg# - liquid-gas capillary pressure.
```

SLIQ

Specifies the SLIQ property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SLIQ syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SLIQ
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SLIQ
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SLIQ
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SMOL

Specifies the SMOL property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SMOL syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SMOL
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SMOL
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SMOL
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SMOLVD

Defines the initial distribution of the salt molality versus depth. The depth values must increase down the column. See also the EQUILGS keyword.

SMOLVD syntax

```
-- within EQL-ENDEQL brackets

SMOLVD
  depth1  smol1  /
  depth2  smol2  /
  depth3  smol3  /
  ...
/

================================================================================

     depth#  - depth;
     smol#   - salt molality.
```

SSFN

Defines salt saturation functions.

SSFN syntax

```
-- within SAT-ENDSAT brackets

SSFN
   ssol1  krf1 /
   ssol2  krf2 /
   ssol3  krf3 /
   ...
/

================================================================================

     ssol# - salt saturation;
     krf#  - multiplier for fluid phases relative permeability.
```

SSOL

Specifies the SSOL property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SSOL syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SSOL
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SSOL
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SSOL
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SSOLVD

Defines the initial distribution of the saturation of the solid phase versus depth. The depth values must increase down the column. See also the EQUILGS keyword.

SSOLVD syntax

```
-- within EQL-ENDEQL brackets

SSOLVD
  depth1  ssol1  /
  depth2  ssol2  /
  depth3  ssol3  /
  ...
/

================================================================================

     depth#  - depth;
     ssol#   - solid phase saturation.
```

TEMPCVD

Defines the initial distribution of the temperature (degrees Celsius) versus depth. The depth values must increase down the column. See also the EQUILGS keyword.

TEMPCVD syntax

```
-- within EQL-ENDEQL brackets

TEMPCVD
  depth1  tempc1  /
  depth2  tempc2  /
  depth3  tempc3  /
  ...
/

================================================================================

    depth#  - depth;
    tempc#  - temperature (degrees Celcius).
```

TEMPVD

Defines the initial distribution of the temperature (degrees Kelvin) versus depth. The depth values must increase down the column. See also the EQUILGS keyword.

TEMPVD syntax

```
-- within EQL-ENDEQL brackets

TEMPVD
  depth1  temp1  /
  depth2  temp2  /
  depth3  temp3  /
  ...
/

================================================================================

    depth#  - depth;
    temp#   - temperature (degrees Kelvin).
```

VAPWAT

Enables the water component evaporation in the gas phase.

VAPWAT syntax

```
-- in RUNSPEC section

VAPWAT
```

VISTAB

Loads gas viscosity on pressure and temperature (only for CO2-SEQ gas).

VISTAB syntax

```
-- within EOS-ENDEOS, or PVT-ENDPVT brackets, or in the PROPS section

VISTAB
        temp1  temp2  temp3  ...  /
 pres1  vis11  vis12  vis13  ...  /
 pres2  vis21  vis22  vis23  ...  /
 pres3  vis31  vis32  vis33  ...  /
 ...    ...    ...    ...    ...  /
/

================================================================================

     temp#  -  temperature (K);
     pres$  -  pressure;
     vis$#  -  viscosity for a given pressure (pres$) and temperature (temp#).
```

WATER

Enables the water phase. This keyword indicates that the simulation contains the water phase.

WATER syntax

```
-- in RUNSPEC section

WATER
```

XGM

Specifies the XGM property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

XGM syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

XGM
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

XGM
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

XGM
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

XGMVD

Defines the initial distribution of the gas mass fraction in the liquid phase versus depth. The depth values must increase down the column. See also the EQUILGS keyword.

XGMVD syntax

```
-- within EQL-ENDEQL brackets

XGMVD
  depth1  xgm1  /
  depth2  xgm2  /
  depth3  xgm3  /
  ...
/

================================================================================

     depth#  - depth;
     xgm#    - gas mass fraction in the liquid phase.
```

XSM

Specifies the XSM property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

XSM syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

XSM
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

XSM
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

XSM
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

XSMVD

Defines the initial distribution of the salt mass fraction versus depth. The depth values must increase down the column. See also the EQUILGS keyword.

XSMVD syntax

```
-- within EQL-ENDEQL brackets

XSMVD
  depth1  xsm1  /
  depth2  xsm2  /
  depth3  xsm3  /
  ...
/

================================================================================

     depth#  - depth;
     xsm#    - salt mass fraction in two-component water-salt system.
```

YWM

Specifies the YWM property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

YWM syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

YWM
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

YWM
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

YWM
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

YWMVD

Defines the initial distribution of the water vapour mass fraction in the gas phase versus depth. The depth values must increase down the column. See also the EQUILGS keyword.

YWMVD syntax

```
-- within EQL-ENDEQL brackets

YWMVD
  depth1  ywm1  /
  depth2  ywm2  /
  depth3  ywm3  /
  ...
/

================================================================================

     depth#  - depth;
     ywm#    - water vapour mass fraction in the gas phase.
```

1.12. MAGFLUID keywords

The keywords below are available only with MAGFLUID EOS module.

MAGFLUID

Switches the program to the MAGFLUID EOS-module and defines the numbers and the names of the components and phases.

MAGFLUID syntax

```
-- in the RUNSPEC section or within the EOS-ENDEOS brackets

MAGFLUID
    nc   cname(1)  cname(2)  ...  cname(nc)  /
    np   pname(1)  pname(2)  ...  pname(np)  /

================================================================================

    nc         - number of fluid components;
    cname(i)   - name of the i-th component (8-byte character);
    np         - number of fluid phases;
    pname(j)   - name of the j-th phase (8-byte character).
```

MFDIAG

Defines the fluid parameters for a given pressure and temperature.

MFDIAG syntax

```
-- within the EOS-ENDEOS brackets

MFDIAG
  ptname   pres   temprel /
  vertex1  exist1  den1  enthm1  visc1  comp1(1) comp1(2) ... comp1(nc) /
  vertex2  exist2  den2  enthm2  visc2  comp2(1) comp2(2) ... comp2(nc) /
  vertex3  exist3  den3  enthm3  visc3  comp3(1) comp3(2) ... comp3(nc) /
  ...                                                                   /
/

================================================================================

    ptname    - name of the point in the pres-temp space;
    pres      - pressure;
    temprel   - temperature (degree Celsius);
    vertex#   - name of the vertex;
    exist#    - must be YES (default) or NO:
                YES - the corresponding vertex# is defined at pres and temprel;
    den#      - fluid density;
    enthm#    - specific enthalpy of the fluid;
    visc#     - the fluid viscosity;
    comp#(i)  - mass fraction of the i-th component;
    nc        - number of the fluid components defined by the MAGFLUID keyword.
```

MFPT

Defines the interpolation in the $P-T$ space. Each line defines the elements of in which linear (triangle) or bilinear (quadrangle) interpolation of the fluid parameters is applied.

MFPT syntax

```
-- within the EOS-ENDEOS brackets

MFPT
    ptname1(1) ptname2(1) ptname3(1) ptname4(1)  /
    ptname1(2) ptname2(2) ptname3(2) ptname4(2)  /
    ptname1(3) ptname2(3) ptname3(3) ptname4(3)  /
    ...                                          /
/

================================================================================

    ptnamei#   - the names of MFDIAG data. Each line defines the element of
                 interpolation in the P-T space.
                 If i=1,2,3 then the correponding element is triangle and the
                 linear interpolation is implemented in such element.
                 If i=1,2,3,4 then the correponding element is quadrangle and
                 the bilinear interpolation is implemented in such element.
```

MFSIMPLX

Defines the simplexes of the phase diagram.

MFSIMPLX syntax

```
-- within the EOS-ENDEOS brackets

MFSIMPLX
  simplex1   p1   pname1(1)   vertex1(1,1)  vertex1(1,2)  vertex1(1,3)  ... /
                  pname1(2)   vertex1(2,1)  vertex1(2,2)  vertex1(2,3)  ... /
                                    ...
                  pname1(p1)  vertex1(p1,1) vertex1(p1,2) vertex1(p1,3) ... /
  simplex2   p2   pname2(1)   vertex2(1,1)  vertex2(1,2)  vertex2(1,3)  ... /
                  pname2(2)   vertex2(2,1)  vertex2(2,2)  vertex2(2,3)  ... /
                                    ...
                  pname2(p2)  vertex2(p2,1) vertex2(p2,2) vertex2(p2,3) ... /
  simplex3   p3   pname3(1)   vertex3(1,1)  vertex3(1,2)  vertex3(1,3)  ... /
                  pname3(2)   vertex3(2,1)  vertex3(2,2)  vertex3(2,3)  ... /
                                    ...
                  pname3(p3)  vertex3(p3,1) vertex3(p3,2) vertex3(p3,3) ... /
...
/

================================================================================

    simplex#     -  simplex name;
        p#           -  number of phase corresponding to the simplex;
        pname#(i)    -  phase name corresponding to the group of vertices;
        vertex#(i,j) -  vertex name corrsponding to the pname#(i).

```

MFVERTEX

Defines the vertexes of the simplexes.

MFVERTEX syntax

```
-- within the EOS-ENDEOS brackets

MFVERTEX
    pname1  vertex1 /
    pname2  vertex2 /
    pname3  vertex3 /
    ...           /
/

================================================================================

    pname#     - phase name;
    vertex#    - name of the vertex.
```

1.13. SIMPLMOD keywords

The keywords below are available only with SIMPLMOD EOS module.

DENTAB

Loads fluid density on pressure and temperature. This keyword must be used within EOS-ENDEOS brackets.

DENTAB syntax

```
-- within EOS-ENDEOS, or PVT-ENDPVT brackets, or in the PROPS section

DENTAB
        temp1  temp2  temp3  ...  /
 pres1  den11  den12  den13  ...  /
 pres2  den21  den22  den23  ...  /
 pres3  den31  den32  den33  ...  /
 ...    ...    ...    ...    ...  /
/

================================================================================

     temp#  -  temperature (degree Kelvin);
     pres$  -  pressure;
     den$#  -  density for a given pressure (pres$) and temperature (temp#).
```

ENTHTAB

Loads fluid specific enthalpy on pressure and temperature. This keyword must be used within EOS-ENDEOS brackets.

ENTHTAB syntax

```
-- within EOS-ENDEOS, or PVT-ENDPVT brackets, or in the PROPS section

ENTHTAB
        temp1  temp2  temp3  ...  /
 pres1  enth11 enth12 enth13  ...  /
 pres2  enth21 enth22 enth23  ...  /
 pres3  enth31 enth32 enth33  ...  /
 ...    ...    ...    ...    ...  /
/

================================================================================

     temp#  -  temperature (degree Kelvin);
     pres$  -  pressure;
     enth$# -  specific enthalpy for a given pressure (pres$) and
               temperature (temp#).
```

SIMPLMOD

Switches the program to SIMPLMOD EOS module.

SIMPLMOD syntax

```
-- in section RUNSPEC

SIMPLMOD
```

VISTAB

Loads fluid viscosity on pressure and temperature. This keyword must be used within EOS-ENDEOS brackets.

VISTAB syntax

```
-- within EOS-ENDEOS, or PVT-ENDPVT brackets, or in the PROPS section

VISTAB
        temp1  temp2  temp3  ...  /
 pres1  vis11  vis12  vis13  ...  /
 pres2  vis21  vis22  vis23  ...  /
 pres3  vis31  vis32  vis33  ...  /
 ...    ...    ...    ...    ...  /
/

================================================================================

     temp#  -  temperature (K);
     pres$  -  pressure;
     vis$#  -  viscosity for a given pressure (pres$) and temperature (temp#).
```

1.14. T2EOS1 keywords

The keywords below are available only with T2EOS1 EOS module.

SVAP

Specifies the SVAP property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SVAP syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SVAP
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SVAP
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SVAP
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

SWAT

Specifies the SWAT property on a block by block basis in the current input box (see also the BOX keyword). The array loading method depends on the LOADING switch.

SWAT syntax

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in IJK position --

SWAT
   value1  value2  value3  ... value(nbox) /

================================================================================

    value(#) - value assigned to the corresponding grid block in the input box.
               The grid blocks are ordered with i index cycling fastest,
               followed by j and k indexes.
    nbox     - number of grid blocks in the current input box.
               nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
               imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
               BOX.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

              -- if the LOADING switch is in CELLID position --

SWAT
   cellid1  property1 /
   cellid2  property2 /
   cellid3  property3 /
...
/

================================================================================

     cellid#   - the CELLID of cell for which the property is specified
     property# - the property value assighned to the cell

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 3 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

SWAT
```

```
   filetype filename /

================================================================================

        This option enables the property loading from an include file

        filetype = ASCII  - formatted include file option
                 = BINARY - binary include file

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

T2EOS1

Switches the program to T2EOS1 EOS module.

T2EOS1 syntax

```
-- in section RUNSPEC

T2EOS1
```

1.15. USEREOS keywords

The keywords below are available only with USEREOS EOS module. The PVT properties and, optionally, the saturation functions are exported from a shared library. The library can be programmed by user.

USEREOS

Switches the program to USEREOS EOS module. The paths to the shared library and the configuration file are the arguments of the keyword.

USEREOS syntax

```
-- in section RUNSPEC

USEREOS
    dll_file  config_file /

================================================================================

   dll_file     - path to the dynamic link library (i.e. shared library).
                  Default values:
                  - USEREOS.DLL on Windows
                  - USEREOS.SO  on Linux OS
                  - USEREOS.DYLIB on Mac OS
   config_file  - path to the configuration file, i.e. the input data-file for
                  the shared library. By default, config_file is empty string.
```

1.16. U-program. Section CSV-ENDCSV

The keywords below are available only with the U-program (Utilities) in the CSV-ENDCSV section.

Using this section the CSV file can be conviniently edited and processed for further usage with the hydrodynamic simulator.

ARITHCSV

This keyword is useful for performing arithmetic operations with the data in a particular set of rows.

ARITHCSV syntax

```
-- within CSV-ENDCSV brackets

ARITHCSV
  'expression1'  'condition1' /
  'expression2'  'condition2' /
  'expression3'  'condition3' /
  ...                         /
/

================================================================================

    expression#  - arithmetic operation with the data in every row.
    condition#   - logical expression defining the set of rows which the
                   arithmetic operation is applied to. If it is empty, then
                   the operation is applied to all rows.
```

COLNAMES

Defines the columns order in the CSV files loaded by LOADCSV. By default the program uses the order specified by the opening bracket of CSV-ENDCSV.

COLNAMES syntax

```
-- within CSV-ENDCSV brackets

COLNAMES
  name1 name2 name3 ... /

================================================================================

    name#      - name of the variables in the CSV file. Use 1* for the colmuns
                 that need to be skipped.
```

CSV-ENDCSV

The CSV subsection delimiters. The variables (i.e. the columns of CSV) must be defined with the opening bracket. The predefined variables are ACTNUM and ROW. ACTNUM is the activity flag. If it is larger than 0, then the row is active. If ACTNUM is less than or equal to 0 then the row is inactive. The quantity ROW is the row number.

CSV-ENDCSV syntax

```
-- in U-program

CSV
  name1 name2 name3 ... /

-- the CSV-ENDCSV section keywords

ENDCSV

================================================================================

    name#      - name of the variable that can be used in this occurence of
                 CSV-ENDCSV. All variables must be defined in that keyword.
                 Each name is an 8-byte character.
```

HISTOGRA

Creates histogram of average (AVG), minimum (MIN) or maximum (MAX) values. Only active rows are processed by that keyword.

HISTOGRA syntax

```
-- within CSV-ENDCSV brackets

HISTOGRA
   OutFile NameOut  Mode  NameBin Nbins MinV MaxV SkipNoData Period SkipHeader /

================================================================================

    OutFile    -  The output file.
    NameOut    -  Name of the processed parameter.
    Mode       -  NUMB - numer of occurances
                  AVG  - average (default option)
                  MIN  - minimum
                  MAX  - maximum
    NameBin    -  Name of the quantity used for bins.
    Nbins      -  Number of bins.
    MinV       -  Lower limit for the bins axis.
    MaxV       -  Upper limit for the bins axis.
    SkipNoData -  T of F (default option). All bins that produce NaN are
                  skipped.
    Period     -  The period in the case the produced function is periodic.
                  By default the period is Nan, i.e. the function is not
                  periodic.
    SkipHeader -  T or F (default option). If true, then the FUNCSPEC header
                  is skipped.
```

IPOLATOR

Parameters of the interpolation procedure used by MAPXY.

IPOLATOR syntax

```
-- within CSV-ENDCSV brackets

IPOLATOR
  MaxDist  NPoints  Power  Threshold /

================================================================================

    MaxDist    - the maximum distance to the data to be used in the
                 interpolation (default 1e10)
    NPoints    - the maximum number of the closest points used in the
                 interpolation.
    Power      - the exponent used in the iterpolation (default 2.0).
    Treshold   - By default is 1e-12.
```

LOADCSV

Loads the CSV file. The column order in the file is defined by COLNAMES. Note that comma can used as a delimiter between the columns. The closing slash in every row can be skipped.

LOADCSV syntax

```
-- within CSV-ENDCSV brackets

LOADCSV
  var1-1 var2-1 var3-1 ... /
  var1-2 var2-2 var3-2 ... /
  var1-3 var2-3 var3-3 ... /
  ...                      /
/

================================================================================

    var#-$    - the value of var# in the $th row of CSV.
```

MAPXY

Creates an interpolation of the variable and saves it as function of two other variable. The output is ready to load back by UF2DSPEC. Only active rows are used in the interpolation.

MAPXY syntax

```
-- within CSV-ENDCSV brackets

MAPXY
  NameOut OutFile   NameX nx xmin xmax   NameY ny ymin ymax /

================================================================================

    NameOut    -  Name of the interpolated parameter.
    OutFile    -  The output file.
    NameX      -  Name of the quantity changing along axis X.
    nx         -  Number of points along axis X.
    xmin/xmax  -  The lower/upper limit of the interval along axis X.
    NameY      -  Name of the quantity changing along axis Y.
    ny         -  Number of points along axis Y.
    ymin/ymax  -  The lower/upper limit of the interval along axis Y.
```

SAVECSV

Saves the CSV file. The order of columns in the file can be explicitely specified by the keyword.

SAVECSV syntax

```
-- within CSV-ENDCSV brackets

SAVECSV
  Outfile  OnlyAct  name1 name2 name3 ... /

================================================================================

    Outfile   - the output file name.
    OnlyAct   - T or F. If T, then only active rows are saved to the file.
                By default all rows are saved (F).
    name#     - name of the variable (i.e. column) save to the file.

```

SETWHT

Specifies the variable weights used in the interpolation with IPOLATOR. By default all weights equal 1.

SETWHT syntax

```
-- within CSV-ENDCSV brackets

SETWHT
  VarTemplate1  weight1 /
  VarTemplate2  weight2 /
  VarTemplate3  weight3 /
  ...                   /
/

================================================================================

    VarTemplate#  - template for the variable name.
    weight#       - weight for every variable matchin the template.
```

1.17. U-program. Section IAPWS-ENDIAPWS

The keywords below are available only with the U-program (Utilities) in the IAPWS-ENDIAPWS section.

Using this section of the U-program one can calculate the thermophysical properties of pure water and steam using IAPWS tables (see http://www.iapws.org/).

IAPWS-ENDIAPWS

The IAPWS subsection delimiters. This subsection provides options for water and steam properties calculation using IAPWS formulation (see www.iapws.org).

IAPWS-ENDIAPWS syntax

```
-- in U-program

IAPWS

-- the IAPWS section keywords

ENDIAPWS
```

PTREPORT

Reports the phase equilibria parameters for a given pressure and temperature. The data are reported in the LOG-file

PTREPORT syntax

```
-- in IAPWS-ENDIAPWS subsection

PTREPORT
   pres1   temp1 /
   pres2   temp2 /
   pres3   temp3 /
   ...     ...   /
/

================================================================================

   pres# - pressure
   temp# - temperature (degree Celcius)
```

1.18. U-program. Section OPTIM-ENDOPTIM

The keywords below are available only with the U-program (Utilities) in the OPTIM-ENDOPTIM section.

Using this section of the U-program one can

- Conduct inverse modelling using the hydrodynamic program of the MU-FITS simulator;
- Perform history matching to variety of observations, including known data for grid blocks, wells, fluxes between different reservoir regions, point sources etc;
- Conduct automated parametric study of a flow in a subsurface reservoir;
- Optimize a reservoir model.

See also Sect. 7.2.

APPLYDEP

Invokes instant application of the declaration dependencies.

APPLYDEP syntax

```
-- in OPTIM-ENDOPTIM or SCHED-ENDSCHED section of the U-program

APPLYDEP
```

ARITHDCL

Performs an arithmetic operation with scalar variable (i.e. the declarations). This keyword should be regarded a more convinient analogue to OPERDECL and DECDEPND. The operations and functions available for use in the arithmetic expressions are listed in section 1.5. It is recommended to place any expression in single quotes.

ARITHDCL syntax

```
-- within the OPTIM-ENDOPTIM or SCHED-ENDSCHED brackets

ARITHDCL
   'expression1'  /
   'expression2'  /
   'expression3'  /
   ...            /
/

================================================================================

   expression# - an arithmetic expression.
```

AUTO

Affects the interpretation of the ARITHDCL keywords. If AUTO is placed ahead of this keyword, then ARITHDCL is understood as the automatic keyword, i.e. executed before any new simulation.

AUTO syntax

```
-- in the INIT or SCHEDULE section of the H-program
-- within SCHED-ENDSCHED or OPTIM-ENDOPTIM bractkets of the U-program

AUTO
```

BESTFIT

Equates the declarations to current best-fit values. The best-fit values are also reported to the LOG-file.

BESTFIT syntax

```
-- in OPTIM-ENDOPTIM subsection of the U-program

BESTFIT
```

CP2STUDY

Reports the current best-fit to the STUDY-file.

CP2STUDY syntax

```
-- in OPTIM-ENDOPTIM subsection

CP2STUDY
```

CROSS

Performs multiple hydrodynamic simulations for specied ranges of declaration values. The declaration values to be tried are spaced along variation of specified parameters.

CROSS syntax

```
-- in OPTIM-ENDOPTIM subsection of the U-program

CROSS
  decl1   minlimit1  maxlimit1 /
  decl2   minlimit2  maxlimit2 /
  ...
  declN   minlimitN  maxlimitN /
/

================================================================================

    decl#     - declaration to be changed in the cross procedure;
    minlimit# - defines the minimun limit of the parameter;
    maxlimit# - defines the maximum limit of the parameter.

================================================================================

    Parameters to be tried:
    1.  decl1:=decl1-minlimit1; decl2; decl3 ...
    2.  decl1:=decl1+maxlimit1; decl2; decl3 ...
    3.  decl1:=decl1; decl2-minlimit2; decl3 ...
    4.  decl1:=decl1; decl2+maxlimit2; decl3 ...
        ...
```

DECDEPND

Defines dependencies between declarations.

DECDEPND syntax

```
-- in OPTIM-ENDOPTIM or SCHED-ENDSCHED section of the U-program

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

DECDEPND
   ... other record of Option 2 ...
   NOTHING /
   ... other record of Option 2 ...
/

            The record NOTHING clears the depency list.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

DECDEPND
  declmod1  decldep1  oper1 par1_1 par2_1 par3_1 /
  declmod2  decldep2  oper2 par1_2 par2_2 par3_2 /
  declmod3  decldep3  oper3 par1_3 par2_3 par3_3 /
   ...
/

================================================================================

   declmod#    - declaration which is modified (M).
   decldep#    - mnemonic of the dependent declaration (X).
   oper#       - mnemonic of operation which is applied (see the list).
   par1_#      - dependent scalar value.
   par2_#      - dependent scalar value.
   par3_#      - dependent scalar value.

================================================================================

   The keyword results in the following:

       declmod1:=Oper1(declmod1,decldep1,par1_1,par2_1,par3_1)
       declmod2:=Oper2(declmod2,decldep2,par1_2,par2_2,par3_2)
       declmod3:=Oper3(declmod3,decldep3,par1_3,par2_3,par3_3)
       ...
   before any Template transformation to the H-program RUN-file.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

List of available operations		
ADD	-	$M := M + par_1$
ADDARR	-	$M := M + X$
ADDX	-	$M := X + par_1$
COPY	-	$M := X$
COS	-	$M := M + par_1 * \cos(par_2 * X + par_3)$
COSSPEC	-	$M := M + par_1 * \cos(2 * \pi * X/par_2 + par_3)$
CRUSTPER	-	$M := min(10^{par_1 - par_2 * \log_{10}(X/10^3)}, par_3)$
CRUSTPOR	-	$M := max(min(10^{par_1 + par_2 * (X/10^3) + par_3 * (X/10^3)^2}, 1.0), 0.001)$
DISTURB	-	$M := M + par_1 + rand(-1, 1) * par_2$, where $rand(-1, 1)$ is a function generating random numbers in the interval $(-1, 1)$
DIVIARR	-	$M := M/X$
DIVIDE	-	$M := M/par_1$
EQUALS	-	$M := par_1$
EXP	-	$M := M + par_1 \exp(par_2 X) + par_3$
EXP2	-	$M := M + par_3 \exp\left(-\frac{(X - par_2)^2}{par_1^2}\right)$
IFGT	-	$M := X$ if $X > 0$
IFLT	-	$M := X$ if $X < 0$
INV	-	$M := 1/X$
LOGE	-	$M := \ln(X)$
LOG10	-	$M := \log_{10}(X)$
MAXARR	-	$M := max(M, X)$
MAXLIM	-	$M := min\{X, par_1\}$
MAXVALUE	-	$M := min\{M, par_1\}$
MINARR	-	$M := min(M, X)$
MINLIM	-	$M := max\{X, par_1\}$
MINVALUE	-	$M := max\{M, par_1\}$
MOD	-	$M := mod(X, par_1)$; the remainder when X is divided by par_1
MULTA	-	$M := par_1 + X * par_2$
MULTARR	-	$M := M * X$
MULTP	-	$M := par_1 * X^{par_2}$
MULTX	-	$M := X * par_1$
MULTIPLY	-	$M := M * par_1$

List of available operations		
NORMDIST	-	$M := M + \frac{par_3}{2\pi par_1^2} \exp\left(-\frac{(X-par_2)^2}{2par_1^2}\right)$
POLY	-	$M := M + par_1 * X^{par_2}$
POW	-	$M := M^{par_1 * X + par_2}$
SCALE	-	$M := par_1 * X + par_2 * (M - par_1)$
SETINT	-	$M := par_1$ if $par_2 \leq X \leq par_3$
SETEXT	-	$M := par_1$ if $X \leq par_2$ or $X \geq par_3$
SIGN	-	$M := par_1 * sign(X)$
SIN	-	$M := M + par_1 * \sin(par_2 * X + par_3)$
SINSPEC	-	$M := M + par_1 * \sin(2 * \pi * X / par_2 + par_3)$
SLOG	-	$M := 10^{par_1 + X * par_2}$
UF2D	-	$M := UF2D_{par_1}(M, X)$, where $UF2D_{par_1}$ is a function supplied by user
USERFUNC	-	$M := UserFunc_{par_1}(X)$, where $UserFunc_{par_1}$ is a function supplied by user

DECLARE

Defines current declaration values.

DECLARE syntax

```
-- in OPTIM-ENDOPTIM or SCHED-ENDSCHED section of the U-program

DECLARE
   dname1  dvalue1  dtype1  format1 /
   dname2  dvalue2  dtype2  format2 /
   dname3  dvalue3  dtype3  format3 /
   ...     ...      ...     ...     /
/

================================================================================

   dname#   - declaration name (the name must consist of up to 8 capital
              characters strating with $D# (e.g. $D#MYDEC);
   dvalue#  - the current declaration value;
   dtype#   - the declaration data type. Available options are
                REAL - real number;
                INT  - integer;
                CHAR - string of characters.
              The type influences the data specified in the dvalue# field.
    format#  - output format.

               Possible values of format#:

               1PEw.d (or Ew.d) - transfers real values with E exponent.
               Can be applyied only to the Real data types.
               w-d must be greater than 6 (Example: 1PE12.5).

               Fw.d - transfers real values with no exponents.
               Can be applyied only to the Real data types.
               w-d must be greater than 3 (Example: F9.6).

               Iw - transfers decimal integer values.
               Can be applyied only to the Integer data types.
               (Example: I8).

               Aw - transfers characters.
               Can be applyied only to the Character data types.
               (Example: A8).

               Here,
               w is the field width,
               d is the number of digits to the right of the decimal point.
```

DELINDIR

Deletes particular list of files from WORKDIR before running a new simulation.

DELINDIR syntax

```
-- in OPTIM-ENDOPTIM subsection of the U-program

DELINDIR
   ext1  ext2  ext3  ...  /

================================================================================

           ext# - extentsion of the files to be deleted in WORKDIR before any
                  new simulation. By placing NOTHING you can clear the list.

```

EQUALOBJ

Equals the current value of the objective function to the number specified by user.

EQUALOBJ syntax

```
-- in OPTIM-ENDOPTIM subsection of the U-program

EQUALOBJ
  obj_value /
================================================================================

    obj_value  - the value of the objective function.
```

EXECCMD

Defines the command line (execution command) for running the H-program on particular machine.

EXECCMD syntax

```
-- in OPTIM-ENDOPTIM subsection of the U-program

EXECCMD
  execution_command /

================================================================================

   execution_command  - the command for launching the H-program (hydrodynamic
                        simulator.
                        Example 1: 'mpirun -n 4 ./../../BIN/H64.EXL';
                        Example 2: '"C:\Program Files\MPICH2\bin\mpiexec.exe"
                                    -n 1 ../../BIN/H64.EXE'.
                        Note: The command line must not include the RUN-file
                              of the H-program.
```

FUNCNAME

Defines the user function ID loaded by the succeeding FUNCSPEC keyword. The function can be used by the keyword DECDEPND if the operation is USERFUNC.

FUNCNAME syntax

```
-- in all sections except RUNSPEC and POST

FUNCNAME
  funcid /

================================================================================

  funcid - a user function id (must be 1,2,3,...,100; default 1). The function
           id affect the data loaded by the FUNCSPEC keyword.
```

FUNCSPEC

Defines in a tabular form the a function. The function ID is defined by the preceding FUNCNAME keyword. The function can be used by the keywords DECDEPND if the operation is USERFUNC.

FUNCSPEC syntax

```
-- in all sections except RUNSPEC and POST

FUNCSPEC
  arg1  value1 /
  arg2  value2 /
  arg3  value3 /
  ...
/

================================================================================

  arg#   - the argument;
  value# - the function value.
```

NOAUTO

Affects the interpretation of the ARITHDCL keywords. If NOAUTO is placed ahead of this keyword, then ARITHDCL is understood as a nonutomatic keyword, i.e. executed before any new simulation.

NOAUTO syntax

```
-- in the INIT or SCHEDULE section of the H-program
-- within SCHED-ENDSCHED or OPTIM-ENDOPTIM bractkets of the U-program

NOAUTO
```

OPERDECL

Applies instantaneous operations with declarations.

OPERDECL syntax

```
-- in OPTIM-ENDOPTIM or SCHED-ENDSCHED section of the U-program

OPERDECL
  declmod1  decldep1  oper1 par1_1 par2_1 par3_1 /
  declmod2  decldep2  oper2 par1_2 par2_2 par3_2 /
  declmod3  decldep3  oper3 par1_3 par2_3 par3_3 /
   ...
/

================================================================================

   declmod#    - declaration which is modified (M).
   decldep#    - mnemonic of the dependent declaration (X).
   oper#       - mnemonic of operation which is applied (see the list).
   par1_#      - dependent scalar value.
   par2_#      - dependent scalar value.
   par3_#      - dependent scalar value.

================================================================================

   The keyword results in the following instantaneous operations:

       declmod1:=Oper1(declmod1,decldep1,par1_1,par2_1,par3_1)
       declmod2:=Oper2(declmod2,decldep2,par1_2,par2_2,par3_2)
       declmod3:=Oper3(declmod3,decldep3,par1_3,par2_3,par3_3)
       ...
```

List of available operations		
ADD	-	$M := M + par_1$
ADDARR	-	$M := M + X$
ADDX	-	$M := X + par_1$
COPY	-	$M := X$
COS	-	$M := M + par_1 * \cos(par_2 * X + par_3)$
COSSPEC	-	$M := M + par_1 * \cos(2 * \pi * X/par_2 + par_3)$
CRUSTPER	-	$M := min(10^{par_1 - par_2 * \log_{10}(X/10^3)}, par_3)$
CRUSTPOR	-	$M := max(min(10^{par_1 + par_2 * (X/10^3) + par_3 * (X/10^3)^2}, 1.0), 0.001)$
DISTURB	-	$M := M + par_1 + rand(-1,1) * par_2$, where $rand(-1,1)$ is a function generating random numbers in the interval $(-1,1)$

List of available operations		
DIVIARR	-	$M := M/X$
DIVIDE	-	$M := M/par_1$
EQUALS	-	$M := par_1$
EXP	-	$M := M + par_1 \exp(par_2 X) + par_3$
EXP2	-	$M := M + par_3 \exp\left(-\frac{(X-par_2)^2}{par_1^2}\right)$
IFGT	-	$M := X$ if $X > 0$
IFLT	-	$M := X$ if $X < 0$
INV	-	$M := 1/X$
LOGE	-	$M := \ln(X)$
LOG10	-	$M := \log_{10}(X)$
MAXARR	-	$M := max(M, X)$
MAXLIM	-	$M := min\{X, par_1\}$
MAXVALUE	-	$M := min\{M, par_1\}$
MINARR	-	$M := min(M, X)$
MINLIM	-	$M := max\{X, par_1\}$
MINVALUE	-	$M := max\{M, par_1\}$
MOD	-	$M := mod(X, par_1)$; the remainder when X is divided by par_1
MULTA	-	$M := par_1 + X * par_2$
MULTARR	-	$M := M * X$
MULTP	-	$M := par_1 * X^{par_2}$
MULTX	-	$M := X * par_1$
MULTIPLY	-	$M := M * par_1$
NORMDIST	-	$M := M + \frac{par_3}{2\pi par_1^2} \exp\left(-\frac{(X-par_2)^2}{2par_1^2}\right)$
POLY	-	$M := M + par_1 * X^{par_2}$
POW	-	$M := M^{par_1 * X + par_2}$
SCALE	-	$M := par_1 * X + par_2 * (M - par_1)$
SETINT	-	$M := par_1$ if $par_2 \leq X \leq par_3$
SETEXT	-	$M := par_1$ if $X \leq par_2$ or $X \geq par_3$
SIGN	-	$M := par_1 * sign(X)$
SIN	-	$M := M + par_1 * \sin(par_2 * X + par_3)$
SINSPEC	-	$M := M + par_1 * \sin(2 * \pi * X/par_2 + par_3)$
SLOG	-	$M := 10^{par_1 + X * par_2}$

List of available operations		
UF2D	-	$M := UF2D_{par_1}(M, X)$, where $UF2D_{par_1}$ is a function supplied by user
USERFUNC	-	$M := UserFunc_{par_1}(X)$, where $UserFunc_{par_1}$ is a function supplied by user

OPTFILES

Specifies the names of the files used in the OPTIM-ENDOPTIM section.

OPTFILES syntax

```
-- in OPTIM-ENDOPTIM subsection

OPTFILES
   title  run-file  log-file  obj-file  rpt-file  bestfit-file  workfile
   study-file /

================================================================================

    title        - title of the hydrodynamic simulation. It is automatically
                   determined after the TEMPLATE keyword (default value is
                   recommened);
    run-file     - the RUN-file of the H-program. It is automatically
                   determined after the TEMPLATE keyword (default value is
                   recommened);
    log-file     - the LOG-file of the H-program. It is automatically
                   determined after the TEMPLATE keyword (default value is
                   recommened);
    obj-file     - the OBJ-file of the H-program. It is automatically
                   determined after the TEMPLATE keyword (default value is
                   recommened);
    rpt-file     - the name of the file (RPT-file) containg a summary of
                   the simulation runs (default value is recommended);
    bestfit-file - the name of the H-program RUN-file with the optimum
                   parameters;
    workfile     - name of the temporary workfile used to trasfer date
                   between the H- and U-program;
    study-file   - the name of the file (STUDY-file) containg a summary of
                   the simulation runs reported by CP2STUDY keyword.
```

OPTIM-ENDOPTIM

The OPTIM section delimiters. This subsection aims at history matching and optimization of hydrodynamic simulations.

OPTIM-ENDOPTIM syntax

```
-- in U-program

OPTIM

-- the OPTIM-ENDOPTIM section keywords

ENDOPTIM
```

OPTIMOPT

Changes options within the OPTIM-ENDOPTIM section.

OPTIMOPT syntax

```
-- in OPTIM-ENDOPTIM subsection

OPTIMOPT
  opt1 opt2 opt3 ... /

================================================================================

    opt#  - if 1 then the corresponding option is enabled, if 0 then the option
            is disabled;
    opt1  - if 1 then every simulation is performaed in a separate working
            directory. The RUNID is added to the name of working directory
            (default 0);
    opt2  - if 1 then the RUNID is added to the RUN-file name (default 0);
    opt3  - if 1 then the RUNID is added to the LOG-file name (default 0);
    opt4  - if 1 then the RUNID is added to the OBJ-file name (default 0);
    opt11 - influences input data of the SEARCH keyword (default 0);
    opt12 - influences the processing of the SEARCH keyword. If 1 then the
            SEARCH procedure is repeated until no new best fit is found
            (default 0);
    opt13 - influences the processing of the CROSS keyword. If 1 then the
            CROSS procedure is repeated until no new best fit is found
            (default 0).
```

OPTPARAM

Changes the optimization parameters.

OPTPARAM syntax

```
-- in OPTIM-ENDOPTIM subsection

OPTPARAM
   obj-name  opt-direction /

================================================================================

    obj-name       - the objective function name (default OBJFUNC);
    opt-direction  - MIN (default value) = the objective function is minimized;
                     MAX = the objective function is maximized.
```

PARALLEL

Changes the maximum number of simulations run in parallel when performing the tasks associated with the SEARCH, SEARCHRD and CROSS keywords.

PARALLEL syntax

```
-- in OPTIM-ENDOPTIM subsection

PARALLEL
  n /

================================================================================

      n - the number of threads.
```

PSO

Applies the particle swarm optimization.

PSO syntax

```
-- in OPTIM-ENDOPTIM subsection of the U-program

PSO
  Nsims  MaxTime  Opt1  Opt2  NParticles omega phip phig vmax reflection/
  decl1  minlimit1  maxlimit1 /
  decl2  minlimit2  maxlimit2 /
  ...
  declN  minlimitN  maxlimitN /
/

================================================================================

    NSims     - Maximum number of simulations (default 1000000);
    MaxTime   - Maximum overall simulation time (minutes). If this value is
                exceeded than the U-program terminates the search
                (default 1e10 min);
    Opt1      - Must be T or F (default F).
                If F then
                  mininum value of decl# = minlimit#;
                  maximum value of decl# = maxlimit#.
                If T then
                  minimum value of decl# = bestfit_of_decl# + minlimit#;
                  maximum value of decl# = bestfit_of_decl# + maxlimit#.
    Opt2      - Must be T or F (default F).
                If F then the bestfit_of_decl# is fixed over the time the
                search procedure is running. The bestfit_of_decl# remains
                equal to its value at the procedure beginning;
                If T then the bestfit_of_decl# is redefined are every
                simulation with smaller objective function.
    NParticles - number of particles in the PSO algorithm (default 8).
    omega      - intertia weight (default 0.5).
    phip       - acceleration coefficient to the best location of a
                 particle (default 2.0).
    phig       - acceleration coefficient to the best location of the
                 swarm (default 2.0).
    vmax       - initial (maximum) velocity of the particles (default
                 0.5). The velocity is normalized to the lenght of the
                 search space.
    Reflection - the reflection coefficient when a particle hits the
                 boundary of the search space. If 0, then the particle
                 loses all normal velocity to the boundary. If 1, then
                 the particle reflects elastically (default 0).
    decl#     - declaration to be changed in the search procefure;
    minlimit# - defines the minimun limit of the parameter;
```

```
45      maxlimit# - defines the maximum limit of the parameter.
46                  If OPTIMOPT(11)=0 then
47                    mininum value of decl# = minlimit#;
48                    maximum value of decl# = maxlimit#.
49                  If OPTIMOPT(11)=1 then
50                    minimum value of decl# = bestfit_of_decl# + minlimit#;
51                    maximum value of decl# = bestfit_of_decl# + maxlimit#.
```

RANDSEED

Initializes the random numbers generator.

RANDSEED syntax

```
-- in sections RNSPEC, GRID, INIT, SCHEDULE, in U-PROGRAM

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

RANDSEED
   seed1  seed2   /

================================================================================

    seed# - an integer used for intialization of the random numbers generator.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

-- to set the seed internaly (using time) the following command should be used.

RANDSEED
   INTERNAL   /

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

RPTFUNC

Reports a function previously specified with the FUNCSPEC keyword.

RPTFUNC syntax

```
-- in   - all sections except RUNSPEC and POST
        - OPTIM-ENDOPTIM subsection of the U-program

RPTFUNC
  funcid  n argmin argmax  outfile /

================================================================================

  funcid  - a user function id (must be 1,2,3,...,100);
  n       - number of report items in the range argmin through argmax;
  argmin  - minimal argument of the report interval;
  argmax  - maximal argument of the report interval;
  outfile - output file name.
```

RUNALL

Runs hydrodynamic simulation with the current declarations in all working directories. The number of the directories is defined by the PARALLEL keyword. The simulations to be run are identical.

RUNALL syntax

```
-- in OPTIM-ENDOPTIM subsection of the U-program

RUNALL

```

RUNCUR

Runs hydrodynamic simulation with the current declarations.

RUNCUR syntax

```
-- in OPTIM-ENDOPTIM subsection of the U-program

RUNCUR
   runid /

================================================================================

     runid - simulation ID.
```

SAVEBEST

Saves to the text file information on the current best fit. Later this file can be uploaded in a new calculation using the INCLUDE keyword.

SAVEBEST syntax

```
-- in OPTIM-ENDOPTIM subsection of the U-program

SAVEBEST
  filename /
================================================================================

    filename  - the name of the file in which the information on the
                best fit parameters are to be exported. Later, this file can be
                uploaded using INCLUDE keyword.
```

SAVERUN

Saves the RUN-file with current declaration values.

SAVERUN syntax

```
-- in OPTIM-ENDOPTIM subsection of the U-program

SAVERUN
   filename /

================================================================================

       filename - the RUN-file name.
```

SEARCH

Performs multiple hydrodynamic simulations for specied ranges of declaration values. The declaration values to be tried are regularly spaced within specified intervals. See also SEARCHRD.

SEARCH syntax

```
-- in OPTIM-ENDOPTIM subsection of the U-program

SEARCH
  decl1  m1  minlimit1  maxlimit1 /
  decl2  m2  minlimit2  maxlimit2 /
  ...
  declN  mN  minlimitN  maxlimitN /
/

================================================================================

    decl#     - declaration to be changed in the search procefure;
    m#        - number of the points between the minvalue and maxvalue
                     of the declaration value;
    minlimit# - defines the minimun limit of the parameter;
    maxlimit# - defines the maximum limit of the parameter.
                If OPTIMOPT(11)=0 then
                  mininum value of decl# = minlimit#;
                  maximum value of decl# = maxlimit#.
                If OPTIMOPT(11)=1 then
                  minimum value of decl# = bestfit_of_decl# + minlimit#;
                  maximum value of decl# = bestfit_of_decl# + maxlimit#.
```

SEARCHRD

Performs multiple hydrodynamic simulations for specied ranges of declaration values. The declaration values to be tried are randomly placed within specified intervals. The interval can be changed if a new best-fit values are determined. See also SEARCH.

SEARCHRD syntax

```
-- in OPTIM-ENDOPTIM subsection of the U-program

SEARCHRD
  Nsims  MaxTime  Opt1  Opt2 /
  decl1   minlimit1  maxlimit1 /
  decl2   minlimit2  maxlimit2 /
  ...
  declN   minlimitN  maxlimitN /
/

================================================================================

    NSims     - Maximum number of simulations (default 1000000);
    MaxTime   - Maximum overall simulation time (minutes). If this value is
                exceeded than the U-program terminates the search
                (default 1e10 min);
    Opt1      - Must be T or F (default F).
                If F then
                  mininum value of decl# = minlimit#;
                  maximum value of decl# = maxlimit#.
                If T then
                  minimum value of decl# = bestfit_of_decl# + minlimit#;
                  maximum value of decl# = bestfit_of_decl# + maxlimit#.
    Opt2      - Must be T or F (default F).
                If F then the bestfit_of_decl# is fixed over the time the
                search procedure is running. The bestfit_of_decl# remains
                equal to its value at the procedure beginning;
                If T then the bestfit_of_decl# is redefined are every
                simulation with smaller objective function.
    decl#     - declaration to be changed in the search procedure;
    minlimit# - defines the minimun limit of the parameter;
    maxlimit# - defines the maximum limit of the parameter.
```

TEMPLATE

Defines the path to the template file.

TEMPLATE syntax

```
-- in OPTIM-ENDOPTIM subsection

TEMPLATE
    template_file/

================================================================================

   template_file  - the template file name (the path to the template file)
```

UF2DNAME

Defines the f(x,y) function ID loaded by the succeeding UF2DSPEC or UF2DSPE0 keyword. The function can be used by the keyword DECDEPND if the operation is UF2D.

UF2DNAME syntax

```
-- in all sections except RUNSPEC and POST

UF2DNAME
  funcid /

================================================================================

  funcid - a user function id (must be 1,2,3,...,100; default 1). The function
           id affect the data loaded by the UF2DSPE0 and UF2DSPEC keywords.
```

UF2DSPE0

Defines in a tabular form the f(x,y) function. The function ID is defined by the preceding UF2DNAME keyword. The function can be used by the keyword DECDEPND if the operation is UF2D.

UF2DSPE0 syntax

```
-- in all sections except RUNSPEC and POST

UF2DSPE0
  nx xmin xmax   ny ymin ymax /
  val11  val21  val31   ...   valnx1  /
  val12  val22  val32   ...   valnx2  /
  val13  val23  val33   ...   valnx3  /
  ...
  val1ny val2ny val3ny  ...   valnxny /
/

================================================================================

  nx      - number of data points uniformly placed along axis X;
  xmin    - the lower interval limit along axis X;
  xmax    - the upper interval limit along axis X;
  ny      - number of data points uniformly placed along axis Y;
  ymin    - the lower interval limit along axis Y;
  ymax    - the upper interval limit along axis Y;
  valij   - the function value at
            x=xmin+(xmax-xmin)*(i-1)/(nx-1) and
            y=ymin+(ymax-ymin)*(j-1)/(ny-1).
```

UF2DSPEC

Defines in a tabular form the f(x,y) function. The function ID is defined by the preceding UF2DNAME keyword. The function can be used by the keyword DECDEPND if the operation is UF2D.

UF2DSPEC syntax

```
-- in all sections except RUNSPEC and POST

UF2DSPEC
      x1     x2     x3            xnx
  y1  val11  val21  val31   ...   valnx1  /
  y2  val12  val22  val32   ...   valnx2  /
  y3  val13  val23  val33   ...   valnx3  /
  ...
  yny val1ny val2ny val3ny  ...   valnxny /
/

================================================================================

  xi      - data points placed along X axis. xi must be ascending;
  yj      - data points placed along Y axis. yi must be ascending;
  valij   - the f(x,y) function value at x=xi and y=yj.
```

WORKDIR

Specifies the name of the directory where the hydrodynamic simulations are performed.

WORKDIR syntax

```
-- in OPTIM-ENDOPTIM subsection

WORKDIR
   workdir /

================================================================================

    workdir      - the working directory where hydrodynamic simulation is
                   performed by the Utilities program.
```

QNT

Applies the quasi-Newton method of optimization. The parameters of the optimization can be adjusted by using QNTPARAM.

QNT syntax

```
-- in OPTIM-ENDOPTIM subsection of the U-program

QNT
  decl1    minlimit1  maxlimit1 /
  decl2    minlimit2  maxlimit2 /
  ...
  declN    minlimitN  maxlimitN /
/

================================================================================

    decl#     - declaration to be changed in the search procefure;
    minlimit# - defines the minimun limit of the parameter;
    maxlimit# - defines the maximum limit of the parameter.
                If OPTIMOPT(11)=0 then
                  mininum value of decl# = minlimit#;
                  maximum value of decl# = maxlimit#.
                If OPTIMOPT(11)=1 then
                  minimum value of decl# = bestfit_of_decl# + minlimit#;
                  maximum value of decl# = bestfit_of_decl# + maxlimit#.
```

QNTPARAM

Changes parameters of the quasi-Newton optimization method.

QNTPARAM syntax

```
-- in OPTIM-ENDOPTIM subsection of the U-program

QNTPARAM
  method tol  miniter maxiter maximp dvarnorm  dvarmult /

================================================================================

    method    - type of the method:
                    BFGS     - Broyden-Fletcher-Goldfarb-Shanno algorithm
                               (default)
                    BROYDEN  - Broyden's method
                    DFP      - Davidon-Fletcher-Powell formula
                    GRADDESC - Gradient descent algorithm
                    NEWTON   - Newton's method
                    SR1      - Symmetric rank-one
    tol       - tolerance (default 0.001)
    miniter   - minimum number of iterations (default 1)
    maxiter   - maximum number of iterations (default 10)
    maximp    - maximum numer of iterations with no improvement in the
                objective function (default 1)
    dvarnorm  - (default 0.005)
    dvarmult  - (default 1e6)
```

1.19. U-program. Section SCHED-ENDSCHED

APPEND

Appends template with current declaration values to the output file.

APPEND syntax

```
-- in OPTIM-ENDOPTIM or SCHED-ENDSCHED section of the U-program

APPEND
```

APPLYDEP

Invokes instant application of the declaration dependencies.

APPLYDEP syntax

```
-- in OPTIM-ENDOPTIM or SCHED-ENDSCHED section of the U-program

APPLYDEP
```

ARITHDCL

Performs an arithmetic operation with scalar variable (i.e. the declarations). This keyword should be regarded a more convinient analogue to OPERDECL and DECDEPND. The operations and functions available for use in the arithmetic expressions are listed in section 1.5. It is recommended to place any expression in single quotes.

ARITHDCL syntax

```
-- within the OPTIM-ENDOPTIM or SCHED-ENDSCHED brackets

ARITHDCL
   'expression1'  /
   'expression2'  /
   'expression3'  /
   ...            /
/

================================================================================

   expression# - an arithmetic expression.
```

AUTO

Affects the interpretation of the ARITHDCL keywords. If AUTO is placed ahead of this keyword, then ARITHDCL is understood as the automatic keyword, i.e. executed before any new simulation.

AUTO syntax

```
-- in the INIT or SCHEDULE section of the H-program
-- within SCHED-ENDSCHED or OPTIM-ENDOPTIM bractkets of the U-program

AUTO
```

DECLARE

Defines current declaration values.

DECLARE syntax

```
-- in OPTIM-ENDOPTIM or SCHED-ENDSCHED section of the U-program

DECLARE
   dname1  dvalue1  dtype1  format1 /
   dname2  dvalue2  dtype2  format2 /
   dname3  dvalue3  dtype3  format3 /
   ...     ...      ...     ...     /
/

================================================================================

   dname#   - declaration name (the name must consist of up to 8 capital
              characters strating with $D# (e.g. $D#MYDEC);
   dvalue#  - the current declaration value;
   dtype#   - the declaration data type. Available options are
                REAL - real number;
                INT  - integer;
                CHAR - string of characters.
              The type influences the data specified in the dvalue# field.
    format#  - output format.

               Possible values of format#:

               1PEw.d (or Ew.d) - transfers real values with E exponent.
               Can be applyied only to the Real data types.
               w-d must be greater than 6 (Example: 1PE12.5).

               Fw.d - transfers real values with no exponents.
               Can be applyied only to the Real data types.
               w-d must be greater than 3 (Example: F9.6).

               Iw - transfers decimal integer values.
               Can be applyied only to the Integer data types.
               (Example: I8).

               Aw - transfers characters.
               Can be applyied only to the Character data types.
               (Example: A8).

               Here,
               w is the field width,
               d is the number of digits to the right of the decimal point.
```

DECDEPND

Defines dependencies between declarations.

DECDEPND syntax

```
-- in OPTIM-ENDOPTIM or SCHED-ENDSCHED section of the U-program

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

DECDEPND
   ... other record of Option 2 ...
   NOTHING /
   ... other record of Option 2 ...
/

            The record NOTHING clears the depency list.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

DECDEPND
  declmod1  decldep1  oper1 par1_1 par2_1 par3_1 /
  declmod2  decldep2  oper2 par1_2 par2_2 par3_2 /
  declmod3  decldep3  oper3 par1_3 par2_3 par3_3 /
   ...
/

================================================================================

   declmod#    - declaration which is modified (M).
   decldep#    - mnemonic of the dependent declaration (X).
   oper#       - mnemonic of operation which is applied (see the list).
   par1_#      - dependent scalar value.
   par2_#      - dependent scalar value.
   par3_#      - dependent scalar value.

================================================================================

   The keyword results in the following:

       declmod1:=Oper1(declmod1,decldep1,par1_1,par2_1,par3_1)
       declmod2:=Oper2(declmod2,decldep2,par1_2,par2_2,par3_2)
       declmod3:=Oper3(declmod3,decldep3,par1_3,par2_3,par3_3)
       ...
   before any Template transformation to the H-program RUN-file.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

List of available operations		
ADD	-	$M := M + par_1$
ADDARR	-	$M := M + X$
ADDX	-	$M := X + par_1$
COPY	-	$M := X$
COS	-	$M := M + par_1 * \cos(par_2 * X + par_3)$
COSSPEC	-	$M := M + par_1 * \cos(2 * \pi * X/par_2 + par_3)$
CRUSTPER	-	$M := min(10^{par_1 - par_2 * \log_{10}(X/10^3)}, par_3)$
CRUSTPOR	-	$M := max(min(10^{par_1 + par_2 * (X/10^3) + par_3 * (X/10^3)^2}, 1.0), 0.001)$
DISTURB	-	$M := M + par_1 + rand(-1, 1) * par_2$, where $rand(-1, 1)$ is a function generating random numbers in the interval $(-1, 1)$
DIVIARR	-	$M := M/X$
DIVIDE	-	$M := M/par_1$
EQUALS	-	$M := par_1$
EXP	-	$M := M + par_1 \exp(par_2 X) + par_3$
EXP2	-	$M := M + par_3 \exp\left(-\frac{(X - par_2)^2}{par_1^2}\right)$
IFGT	-	$M := X$ if $X > 0$
IFLT	-	$M := X$ if $X < 0$
INV	-	$M := 1/X$
LOGE	-	$M := \ln(X)$
LOG10	-	$M := \log_{10}(X)$
MAXARR	-	$M := max(M, X)$
MAXLIM	-	$M := min\{X, par_1\}$
MAXVALUE	-	$M := min\{M, par_1\}$
MINARR	-	$M := min(M, X)$
MINLIM	-	$M := max\{X, par_1\}$
MINVALUE	-	$M := max\{M, par_1\}$
MOD	-	$M := mod(X, par_1)$; the remainder when X is divided by par_1
MULTA	-	$M := par_1 + X * par_2$
MULTARR	-	$M := M * X$
MULTP	-	$M := par_1 * X^{par_2}$
MULTX	-	$M := X * par_1$
MULTIPLY	-	$M := M * par_1$

List of available operations		
NORMDIST	-	$M := M + \frac{par_3}{2\pi par_1^2} \exp\left(-\frac{(X-par_2)^2}{2par_1^2}\right)$
POLY	-	$M := M + par_1 * X^{par_2}$
POW	-	$M := M^{par_1 * X + par_2}$
SCALE	-	$M := par_1 * X + par_2 * (M - par_1)$
SETINT	-	$M := par_1$ if $par_2 \leq X \leq par_3$
SETEXT	-	$M := par_1$ if $X \leq par_2$ or $X \geq par_3$
SIGN	-	$M := par_1 * sign(X)$
SIN	-	$M := M + par_1 * \sin(par_2 * X + par_3)$
SINSPEC	-	$M := M + par_1 * \sin(2 * \pi * X / par_2 + par_3)$
SLOG	-	$M := 10^{par_1 + X * par_2}$
UF2D	-	$M := UF2D_{par_1}(M, X)$, where $UF2D_{par_1}$ is a function supplied by user
USERFUNC	-	$M := UserFunc_{par_1}(X)$, where $UserFunc_{par_1}$ is a function supplied by user

FUNCNAME

Defines the user function ID loaded by the succeeding FUNCSPEC keyword. The function can be used by the keyword DECDEPND if the operation is USERFUNC.

FUNCNAME syntax

```
-- in all sections except RUNSPEC and POST

FUNCNAME
  funcid /

================================================================================

  funcid - a user function id (must be 1,2,3,...,100; default 1). The function
           id affect the data loaded by the FUNCSPEC keyword.
```

FUNCSPEC

Defines in a tabular form the a function. The function ID is defined by the preceding FUNCNAME keyword. The function can be used by the keywords DECDEPND if the operation is USERFUNC.

FUNCSPEC syntax

```
-- in all sections except RUNSPEC and POST

FUNCSPEC
  arg1  value1 /
  arg2  value2 /
  arg3  value3 /
  ...
/

================================================================================

  arg#   - the argument;
  value# - the function value.
```

NOAUTO

Affects the interpretation of the ARITHDCL keywords. If NOAUTO is placed ahead of this keyword, then ARITHDCL is understood as a nonutomatic keyword, i.e. executed before any new simulation.

NOAUTO syntax

```
-- in the INIT or SCHEDULE section of the H-program
-- within SCHED-ENDSCHED or OPTIM-ENDOPTIM bractkets of the U-program

NOAUTO
```

OPERDECL

Applies instantaneous operations with declarations.

OPERDECL syntax

```
-- in OPTIM-ENDOPTIM or SCHED-ENDSCHED section of the U-program

OPERDECL
  declmod1  decldep1  oper1 par1_1 par2_1 par3_1 /
  declmod2  decldep2  oper2 par1_2 par2_2 par3_2 /
  declmod3  decldep3  oper3 par1_3 par2_3 par3_3 /
   ...
/

================================================================================

   declmod#    - declaration which is modified (M).
   decldep#    - mnemonic of the dependent declaration (X).
   oper#       - mnemonic of operation which is applied (see the list).
   par1_#      - dependent scalar value.
   par2_#      - dependent scalar value.
   par3_#      - dependent scalar value.

================================================================================

   The keyword results in the following instantaneous operations:

       declmod1:=Oper1(declmod1,decldep1,par1_1,par2_1,par3_1)
       declmod2:=Oper2(declmod2,decldep2,par1_2,par2_2,par3_2)
       declmod3:=Oper3(declmod3,decldep3,par1_3,par2_3,par3_3)
       ...
```

List of available operations		
ADD	-	$M := M + par_1$
ADDARR	-	$M := M + X$
ADDX	-	$M := X + par_1$
COPY	-	$M := X$
COS	-	$M := M + par_1 * \cos(par_2 * X + par_3)$
COSSPEC	-	$M := M + par_1 * \cos(2 * \pi * X/par_2 + par_3)$
CRUSTPER	-	$M := min(10^{par_1 - par_2 * \log_{10}(X/10^3)}, par_3)$
CRUSTPOR	-	$M := max(min(10^{par_1 + par_2 * (X/10^3) + par_3 * (X/10^3)^2}, 1.0), 0.001)$
DISTURB	-	$M := M + par_1 + rand(-1, 1) * par_2$, where $rand(-1, 1)$ is a function generating random numbers in the interval $(-1, 1)$

List of available operations		
DIVIARR	-	$M := M/X$
DIVIDE	-	$M := M/par_1$
EQUALS	-	$M := par_1$
EXP	-	$M := M + par_1 \exp(par_2 X) + par_3$
EXP2	-	$M := M + par_3 \exp\left(-\frac{(X-par_2)^2}{par_1^2}\right)$
IFGT	-	$M := X$ if $X > 0$
IFLT	-	$M := X$ if $X < 0$
INV	-	$M := 1/X$
LOGE	-	$M := \ln(X)$
LOG10	-	$M := \log_{10}(X)$
MAXARR	-	$M := max(M, X)$
MAXLIM	-	$M := min\{X, par_1\}$
MAXVALUE	-	$M := min\{M, par_1\}$
MINARR	-	$M := min(M, X)$
MINLIM	-	$M := max\{X, par_1\}$
MINVALUE	-	$M := max\{M, par_1\}$
MOD	-	$M := mod(X, par_1)$; the remainder when X is divided by par_1
MULTA	-	$M := par_1 + X * par_2$
MULTARR	-	$M := M * X$
MULTP	-	$M := par_1 * X^{par_2}$
MULTX	-	$M := X * par_1$
MULTIPLY	-	$M := M * par_1$
NORMDIST	-	$M := M + \frac{par_3}{2\pi par_1^2} \exp\left(-\frac{(X-par_2)^2}{2par_1^2}\right)$
POLY	-	$M := M + par_1 * X^{par_2}$
POW	-	$M := M^{par_1 * X + par_2}$
SCALE	-	$M := par_1 * X + par_2 * (M - par_1)$
SETINT	-	$M := par_1$ if $par_2 \leq X \leq par_3$
SETEXT	-	$M := par_1$ if $X \leq par_2$ or $X \geq par_3$
SIGN	-	$M := par_1 * sign(X)$
SIN	-	$M := M + par_1 * \sin(par_2 * X + par_3)$
SINSPEC	-	$M := M + par_1 * \sin(2 * \pi * X/par_2 + par_3)$
SLOG	-	$M := 10^{par_1 + X * par_2}$

List of available operations		
UF2D	-	$M := UF2D_{par_1}(M, X)$, where $UF2D_{par_1}$ is a function supplied by user
USERFUNC	-	$M := UserFunc_{par_1}(X)$, where $UserFunc_{par_1}$ is a function supplied by user

RANDSEED

Initializes the random numbers generator.

RANDSEED syntax

```
-- in sections RNSPEC, GRID, INIT, SCHEDULE, in U-PROGRAM

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 1 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

RANDSEED
   seed1  seed2   /

===============================================================================

    seed# - an integer used for intialization of the random numbers generator.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  OPTION 2 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

-- to set the seed internaly (using time) the following command should be used.

RANDSEED
   INTERNAL   /

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

RPTFUNC

Reports a function previously specified with the FUNCSPEC keyword.

RPTFUNC syntax

```
-- in   - all sections except RUNSPEC and POST
        - OPTIM-ENDOPTIM subsection of the U-program

RPTFUNC
  funcid  n argmin argmax  outfile /

================================================================================

  funcid  - a user function id (must be 1,2,3,...,100);
  n       - number of report items in the range argmin through argmax;
  argmin  - minimal argument of the report interval;
  argmax  - maximal argument of the report interval;
  outfile - output file name.
```

SCHED-ENDSCHED

The SCHED section delimiters. This section aims at preparation of the input data for SCHEDULE section of the H-program.

SCHED-ENDSCHED syntax

```
-- in U-program

SCHED

-- the SCHED-ENDSCHED section keywords

ENDSCHED
```

SCHFILES

Defines the output file to which the data will be added with the APPEND keyword.

SCHFILES syntax

```
-- in SCHED-ENDSCHED section of the U-program

SCHFILES
   output_file   delfile /

================================================================================

  output_file   - output file name;
  delfile       - if T then the current output file content will be eraised;
                - if F (default value) the current content will not be deleted.
```

TEMPLATE

Defines the path to the template file.

TEMPLATE syntax

```
-- in OPTIM-ENDOPTIM subsection

TEMPLATE
    template_file/

================================================================================

   template_file  - the template file name (the path to the template file)
```

UF2DNAME

Defines the f(x,y) function ID loaded by the succeeding UF2DSPEC or UF2DSPE0 keyword. The function can be used by the keyword DECDEPND if the operation is UF2D.

UF2DNAME syntax

```
-- in all sections except RUNSPEC and POST

UF2DNAME
  funcid /

================================================================================

  funcid - a user function id (must be 1,2,3,...,100; default 1). The function
           id affect the data loaded by the UF2DSPE0 and UF2DSPEC keywords.
```

UF2DSPE0

Defines in a tabular form the f(x,y) function. The function ID is defined by the preceding UF2DNAME keyword. The function can be used by the keyword DECDEPND if the operation is UF2D.

UF2DSPE0 syntax

```
-- in all sections except RUNSPEC and POST

UF2DSPE0
  nx xmin xmax   ny ymin ymax /
  val11  val21  val31   ...   valnx1  /
  val12  val22  val32   ...   valnx2  /
  val13  val23  val33   ...   valnx3  /
  ...
  val1ny val2ny val3ny  ...   valnxny /
/

================================================================================

  nx      - number of data points uniformly placed along axis X;
  xmin    - the lower interval limit along axis X;
  xmax    - the upper interval limit along axis X;
  ny      - number of data points uniformly placed along axis Y;
  ymin    - the lower interval limit along axis Y;
  ymax    - the upper interval limit along axis Y;
  valij   - the function value at
            x=xmin+(xmax-xmin)*(i-1)/(nx-1) and
            y=ymin+(ymax-ymin)*(j-1)/(ny-1).
```

UF2DSPEC

Defines in a tabular form the f(x,y) function. The function ID is defined by the preceding UF2DNAME keyword. The function can be used by the keyword DECDEPND if the operation is UF2D.

UF2DSPEC syntax

```
-- in all sections except RUNSPEC and POST

UF2DSPEC
      x1     x2     x3            xnx
  y1  val11  val21  val31   ...   valnx1  /
  y2  val12  val22  val32   ...   valnx2  /
  y3  val13  val23  val33   ...   valnx3  /
  ...
  yny val1ny val2ny val3ny  ...   valnxny /
/

================================================================================

  xi      - data points placed along X axis. xi must be ascending;
  yj      - data points placed along Y axis. yi must be ascending;
  valij   - the f(x,y) function value at x=xi and y=yj.
```

1.20. U-program. Section SATFC-ENDSATFC

BRCOREY

Switches the program to the Brooks-Corey model. Note, that SPECMOD influences the processing of this keyword.

BRCOREY syntax

```
-- in SATFC-ENDSATFC subsection of the U-program

BRCOREY
  lambda /

================================================================================

   lambda  - parameter of the Brooks-Corey model (default 2).
```

COREY

Switches the program to the Corey model. This is the default model. Note, that SPECMOD influences the processing of this keyword.

COREY syntax

```
-- in SATFC-ENDSATFC subsection of the U-program

COREY
  n1  n2 /

================================================================================

   n1 and n2   - the exponents of the Corey functions (default 2 and 2).
```

EPOINTS

Specifies the horizontal and vertical end-points for the relative permeability and capillary pressure curves.

EPOINTS syntax

```
-- in SATFC-ENDSATFC subsection of the U-program

EPOINTS
  smin1 sc1 kmax1 smin2 sc2 kmax2  Pscl /

================================================================================

   smin1    - the minimum (connate) saturation of phase 1 (default 0);
   sc1      - the critical saturation of phase 1 (default 0);
   krmax1   - the maximum relative permeability of phase 1 (default 1);
   smin2    - the minimum (connate) saturation of phase 2 (default 0);
   sc2      - the critical saturation of phase 2 (default 0);
   krmax2   - the maximum relative permeability of phase 2 (default 1);
   Pscl     - the vertical scale factor for the capillary pressure curve
              (default 0).
```

INVERSE

The relative permeability and capillary pressure curves are reported as functions of the second phase saturation. By default they are reported as functions of the first phase saturation.

INVERSE syntax

```
-- in SATFC-ENDSATFC subsection of the U-program

INVERSE
```

PCAPZERO

Switches the program to the zero capillary pressure. Note, that SPECMOD influences the processing of this keyword.

PCAPZERO syntax

```
-- in SATFC-ENDSATFC subsection of the U-program

PCAPZERO
```

PCLIMITS

Specifies the minimum and maximum capillary pressures (the cut-off values).

PCLIMITS syntax

```
-- in SATFC-ENDSATFC subsection of the U-program

PCLIMITS
  PcapMin  PcapMax /

================================================================================

   PcapMin  - the minimum capillary pressure;
   PcapMax  - the maximum capillary pressure.
```

SATFC-ENDSATFC

The SATFC section delimiters. This section aims at preparation of the saturation function tables for the H-program.

SATFC-ENDSATFC syntax

```
-- in U-program

SATFC

-- the SATFC-ENDSATFC section keywords

ENDSATFC
```

SATVAL

Specifies the saturations for which the table is generated.

SATVAL syntax

```
-- in SATFC-ENDSATFC subsection of the U-program

SATVAL
  sat1  sat2   sat3   ... /

================================================================================

   sat#  - the values of the first phase saturation for which the table will
           be generated. Must be sorted in ascending order.
```

SAVESFC

Saves the saturation functions to an external file. Note, that the saturation values must be specified by SATVAL before using this keyword.

SAVESFC syntax

```
-- in SATFC-ENDSATFC subsection of the U-program

SAVESFC
  TableName   Mode   Filename /

================================================================================

   TableName   - name of the output table, e.g. SWOF (default SATTAB)
   Mode        - Available options:
                  NEW - the file is eraised before exportinf the table
                        (default).
                  APPEND - the table is written at the end of the file.
   FileName    - Output file name (default SATFC.INC)
```

SFCOPT

Allows the explicit definition of some program options.

SFCOPT syntax

```
-- in SATFC-ENDSATFC subsection of the U-program

SFCOPT
  SaveCP   InverseOn /

================================================================================

   SaveCP    -  T or F. If T then the capillary pressure is reported in the
                output table (default T).
   InverseOn -  T or F. If T then the saturation table is reported against the
                second phase saturation (default F).
```

SPECMOD

Specifies whether the relative permeability and capillary pressure models can be changed with the BRCOREY, COREY, PCAPZERO, and VGENUCH keywords. For example, using this keyword, one can specify the Brooks-Corey model for the relative permeability and the van Genuchten model for the capillary pressure.

SPECMOD syntax

```
-- in SATFC-ENDSATFC subsection of the U-program

SPECMOD
  mnem1  mnem2 ... /

================================================================================

   if mnem# = RP or RELPERM then the model of the relative permeability can
              be changed.
   if mnem# = CP or CAPPRES then the model of the capillary pressure can
              be changed.
```

VGENUCH

Switches the program to the van Genuchten model. Note, that SPECMOD influences the processing of this keyword.

VGENUCH syntax

```
-- in SATFC-ENDSATFC subsection of the U-program

VGENUCH
  lambda  sls /

================================================================================

   lambda and sls   - parameters of the van Genuchten model (default values
                      are lambda=0.2 and sls=1.0).
```

1.21. PVT program (I-program) keywords

The keywords below are available only with PVT program.

LOADEOS

Loads EOS-file with the properties of binary mixture.

LOADEOS syntax

```
-- in PROPS section

LOADEOS
   filename /

================================================================================

   filename - name of EOS file.
```

PHEQ

Calculates properties of binary mixture for a given pressure, enthalpy and composition. The PVT report is outputted in the LOG-file.

PHEQ syntax

```
PHEQ
  pres1  entht1  comp1t1 comp2t1 /
  pres2  entht2  comp1t2 comp2t2 /
  pres3  entht3  comp1t3 comp2t3 /
  ...
/

================================================================================

  pres#   - pressure;
  entht#  - total molar enthalpy;
  comp1t# - total molar fraction of the 1st component;
  comp2t# - total molar fraction of the 2nd component.
```

PHEQPTX

Calculates properties of binary mixture for a given pressure, temperature and composition. The PVT report is outputted in the LOG-file.

PHEQPTX syntax

```
PHEQPTX
  pres1  temp1  comp1t1 comp2t1 /
  pres2  temp2  comp1t2 comp2t2 /
  pres3  temp3  comp1t3 comp2t3 /
  ...
/

================================================================================

  pres#   - pressure;
  temp#   - temperature (degrees of Kelvin);
  comp1t# - total molar fraction of the 1st component;
  comp2t# - total molar fraction of the 2nd component.
```

2 Mnemonics

2.1. Using Mnemonics

The list below is the list of MUFITS mnemonics describing physical, geological, geometrical, and logical data in a simulation. The column on the left contains names of the available mnemonics. The column on the right contains the description of mnemonics. The data in the simulator can be invoced using the names of mnemonics in RUN-file. Using mnemonics you can

- load data in simulation (e.g., see keywords ACTNUM, PORO, PRES, etc.);
- perform operations on arrays (e.g., see keywords EQUALS, COPY, OPERATE, etc.);
- specify regions for thermophysical properties or automatic fluid-in-place calculation;
- specify initial or boundary conditions;
- specify output data which simulator saves in summary files (e.g., see keyword RPTGRID, RPTSUM).

2.2. Kernel mnemonics

Kernel mnemonics		
ACG%name	–	Gas phase index for additional component 'name'
ACL%name	–	Liquid phase index for additional component 'name'
ACTNUM	–	Activity flag. Available values: 0 – inactive, 1 – active, 2 – fixed parameters
AREA	–	Interface area for logical connection. A two-component vector which defines cell face area in two connected cells in the direction of connection (vector (2))
AREACONN	–	Interface area for logical connection.
C#c$p	–	Component c=1,2,3,... mass fraction in phase p=1,2,3,...
CAP%name	–	Capillary pressure associated with phase 'name'
CBLKROCK	–	Rock compressibility
CELLID	–	Cell ID
CELLVOL	–	Cell geometrical volume
CHARGRP	–	Group character name
CHARNAME	–	Character name
CEG%name	–	Mass concentration of primary components in gas phase for 'name' additional component
CEL%name	–	Mass concentration of primary components in liquid phase for 'name' additional component
CHPT#c	–	Chemical potential of component c=1,2,3,...
CMG%name	–	Mass concentration of additional component 'name' in gas phase
CML%name	–	Mass concentration of additional component 'name' in liquid phase
CMT%name	–	Total mass concentration of additional component 'name' (primary variable)
CNTNAME	–	Connection type. E.g., IFACE – interface, PUMP – pumping device, COMPDATA – well completion, PIPEJUNC – pipe junction
CNTNUM	–	Connection type integer ID
COMPTM#c	–	Total mass fraction of the component c=1,2,3,...
COMPTX#c	–	Total molar fraction of the component c=1,2,3,...

Kernel mnemonics		
CONNID	–	Connection ID. IDs of two connected cells (vector (2))
CORNAUX1	–	An auxiliary array for operations on the corner point depths
CORNAUX2	–	An auxiliary array for operations on the corner point depths
CPERMULT	–	Permeability multiplier (ROCKMULT option)
CPORMULT	–	Porosity multiplier (ROCKMULT option)
CRFLFLAG	–	Cross-flow flag for a well completion
DATE	–	Date
DAY	–	Calendar day
DCTTMULT	–	Minus Log10 of transmissibility multiplier induced by ductile behavior of rocks
DDENGRAV	–	Density change with the respect to the reference state. It is used in the calculation of the gravity changes.
DENT	–	Total density of multiphase fluid
DENX$p	–	Molar density of phase p=1,2,3,...
DEN$p	–	Density of phase p=1,2,3,...
DEN%name	–	Density of phase 'name'
DEPTH	–	Cell center depth
DEPTHOBS	–	Depth of the observation point (map axes)
DEPTHQ	–	Depth of connection at the interface between connected cells
DIFFMI	–	Multiplier of diffusivity along i-index direction (usually X axis)
DIFFMJ	–	Multiplier of diffusivity along j-index direction (usually Y axis)
DIFFMK	–	Multiplier of diffusivity along k-index direction (usually Z axis)
DIFFMMF	–	Multiplier of diffusivity for matrix-fracture connections
DIFFMX	–	Multiplier of diffusivity along X axis
DIFFMY	–	Multiplier of diffusivity along Y axis
DIFFMZ	–	Multiplier of diffusivity along Z axis
DIRNUM	–	Direction ID for a connection. Available values: 0 – undefined, 1 – along i-index axis, 2 – along j-index axis, 3 – along k-index axis
DISTANCE	–	Distance from cell center to interface between two connected cells (vector (2))

Kernel mnemonics		
DPCAP$p	–	The phase p=1,2,3,... pressure deviation from the pressure provided by the PRES mnemonic
DPNUM	–	DPNUM flag. If DPNUM=1 in two connected grid blocks, then both fracture and matrix connections are created in double porosity simulations. If DPNUM=0 in one of two connected grid blocks, then only fracture connections are created. By default, DPNUM=0 if DUALPERM is not active, and DPNUM=1 if DUALPERM is active.
DPRECIP	–	Precipitate phase density.
DTRAN	–	Diffusivity
DVOLDISP	–	Relative volume change used in the calculation of the ground displacement.
DX	–	Grid block length along X-axis
DY	–	Grid block length along Y-axis
DZ	–	Grid block length along Z-axis
DZMTRXV	–	Matrix block length along Z-axis in the gravity drainage model
EOSNUM	–	Number of equation of state region
EPRECIP	–	Precipitate phase specific enthalpy.
EQLNUM	–	Number of resevoir initial equilibration region
ETM$p	–	Specific enthalpy of phase p=1,2,3,...
ETM%name	–	Specific enthalpy of phase 'name'
ETX$p	–	Molar enthalpy of phase p=1,2,3,...
GMIR#c	–	Group mass injection rate of component c=1,2,3,...
GMIRALL	–	Summed over all components mass injection rate (for group)
GMIT#c	–	Group mass injection total of component c=1,2,3,...
GMITALL	–	Summed over all components mass injection total (for group)
GMPI#c	–	Group mass potential injection of component c=1,2,3,...
GMPIALL	–	Summed over all components mass potential injection (for group)
GMPP#c	–	Group mass potential production of component c=1,2,3,...
GMPPALL	–	Summed over all components mass potential production (for group)

Kernel mnemonics		
GMPR#c	–	Group mass production rate of component c=1,2,3,...
GMPRALL	–	Summed over all components mass production rate (for group)
GMPT#c	–	Group mass production total of component c=1,2,3,...
GMPTALL	–	Summed over all components mass production total (for group)
GRUPNAME	–	Group name
FIPCELL	–	Fluid-in-place region number when reporting data
FIPCONN	–	Fluid-in-place connection ID when reporting data. The numbers of two fluid-in-place regions for which the data are reported (vector(2))
FIPNUM	–	Fluid-in-place region number
FLUX#c	–	Component c=1,2,3,... flux between two connected cells
FLUXE	–	Energy flux (defined for connections)
FLUXI#c	–	Estimation of the component c=1,2,3,... flux along i-index direction (defined in cells)
FLUXI#E	–	Estimation of the energy flux along i-index direction (defined in cells)
FLUXI#T	–	Estimation of the total mass flux along i-index direction (defined in cells)
FLUXJ#c	–	Estimation of the component c=1,2,3,... flux along j-index direction (defined in cells)
FLUXJ#E	–	Estimation of the energy flux along j-index direction (defined in cells)
FLUXJ#T	–	Estimation of the total mass flux along j-index direction (defined in cells)
FLUXK#c	–	Estimation of the component c=1,2,3,... flux along k-index direction (defined in cells)
FLUXK#E	–	Estimation of the energy flux along k-index direction (defined in cells)
FLUXK#T	–	Estimation of the total mass flux along k-index direction (defined in cells)
FLUXNUM	–	Fluxnum region number
FLUXDPT	–	Total mass flux from fractures into matrix

Kernel mnemonics		
FLUXT	–	Total mass flux between two connected cells
FMIR#c	–	Field mass injection rate of component c=1,2,3,...
FMIRALL	–	Summed over all components field mass injection rate
FMIT#c	–	Field mass injection total of component c=1,2,3,...
FMITALL	–	Summed over all components field mass injection total
FMPR#c	–	Field mass production rate of component c=1,2,3,...
FMPRALL	–	Summed over all components field mass production rate
FMPT#c	–	Field mass production total of component c=1,2,3,...
FMPTALL	–	Summed over all components field mass production total
FSE#c	–	Field storage efficiency of component c=1,2,3,...
FVIR	–	Field volume injection rate (at the injection tank conditions)
FVIT	–	Field volume injection total (at the injection tank conditions)
GBVMULT	–	Multiplier of the grid block volume
GRAVMT	–	Gravity change (absolute value)
GRAVMTX	–	Gravity change along Ox axis (map axes)
GRAVMTY	–	Gravity change along Oy axis (map axes)
GRAVMTZ	–	Gravity change along Oz axis (positive if gravity increases)
HCONDCF	–	Rock heat conduction coefficient defined for a connection (vector(2)). These are two heat conduction coefficients defined in the connected blocks
HCONDCFI	–	Heat conduction coefficient along i-index direction (usually X axis)
HCONDCFJ	–	Heat conduction coefficient along j-index direction (usually Y axis)
HCONDCFK	–	Heat conduction coefficient along k-index direction (usually Z axis)
HCONDCFX	–	Heat conduction coefficient along X axis
HCONDCFY	–	Heat conduction coefficient along Y axis
HCONDCFZ	–	Heat conduction coefficient along Z axis
I-IJKRES	–	i index in the ijk-indexed box of the parent reservoir
IMBNUM	–	Imbibition saturation functions region number
INCONUM	–	A region number. Does not have predefined meaning

Kernel mnemonics		
ILOC	–	i index in the ijk-indexed box of the current grid
IRES	–	i index in the ijk-indexed box of the parent reservoir
J-IJKRES	–	j index in the ijk-indexed box of the parent reservoir
JLOC	–	j index in the ijk-indexed box of the current grid
JRES	–	j index in the ijk-indexed box of the parent reservoir
K-IJKRES	–	k index in the ijk-indexed box of the parent reservoir
KLOC	–	k index in the ijk-indexed box of the current grid
KRES	–	k index in the ijk-indexed box of the parent reservoir
KVL%name	–	K-value of additional component 'name'
LX	–	Matrix block length along X-axis in double porosity models
LY	–	Matrix block length along Y-axis in double porosity models
LZ	–	Matrix block length along Z-axis in double porosity models
MASS#c	–	Total mass of the component c=1,2,3,...
MFRAC$p	–	Molar fraction of phase p=1,2,3,...
MINCNUM	–	Dual porosity flag. Available values: 0 – fracture block, 1 – matrix block
MINCSIZE	–	Number of matrix multiple interacting continuums
MOBT	–	Total mobility of multiphase fluid
MOB$p	–	Mobility of phase p=1,2,3,...
MOLDEN	–	Molar density
MONTH	–	Calendar month (JAN, FEB, ...)
MPINUM	–	MPI-partition region number
MPIRANK	–	MPI-process rank
MTR#c	–	Component c=1,2,3,... flux out of cell
MTR%name	–	Additional component 'name' flux out of cell
MULTPV	–	User defined pore volume multiplier
MULTX	–	User defined transmissibility multiplier along axis X
MULTX-	–	User defined transmissibility multiplier along axis X in the negative direction
MULTY	–	User defined transmissibility multiplier along axis Y
MULTY-	–	User defined transmissibility multiplier along axis Y in the negative direction
MULTZ	–	User defined transmissibility multiplier along axis Z

Kernel mnemonics		
MULTZ-	–	User defined transmissibility multiplier along axis Z in the negative direction
NDTNAME	–	Cell (node) type. E.g., BLOCK – grid block, PIPESEGM – pipe segment, TANK – stock tank
NDTNUM	–	Cell (node) type integer ID
NEQNUM	–	Number of balance equations solved in cell
NKFACES	–	Number of k interfaces for this grid block
NTG	–	Net-to-gross ratio
OBSID	–	ID of the Observation point
ORIGNUM	–	if 1 – the cell was created using the BOUNDARY keyword. 0 – otherwise
PBLKROCK	–	Reference pressure for rock
PERM	–	Permeability defined for a connection (vector(2)). These are two permeabilities defined in the connected blocks
PERMI	–	Permeability along i-index direction (usually X axis)
PERMIFR	–	Dynamic fractures permeability along i-index direction (usually X axis)
PERMJ	–	Permeability along j-index direction (usually Y axis)
PERMJFR	–	Dynamic fractures permeability along j-index direction (usually Y axis)
PERMK	–	Permeability along k-index direction (usually Z axis)
PERMKFR	–	Dynamic fractures permeability along k-index direction (usually Z axis)
PERMX	–	Permeability along X axis
PERMXFR	–	Dynamic fractures permeability along X axis
PERMY	–	Permeability along Y axis
PERMYFR	–	Dynamic fractures permeability along Y axis
PERMZ	–	Permeability along Z axis
PERMZFR	–	Dynamic fractures permeability along Z axis
PFLDFACT	–	Pore-fluid factor (i.e. pore pressure over the failure pressure)
PHID$p	–	ID of phase p=1,2,3,...
PHST	–	Number of phases

Kernel mnemonics		
PHST0	–	Number of phases (not including phases introduced by AUXPHASE option)
PJDIAM	–	Diameter of pipe junction
PJLENGTH	–	Length of pipe junction
PJROUGH	–	Relative roughness of th pipe junction
POREVOL	–	Pore volume
PORO	–	Initial reservoir porosity
POROAVG	–	Averaged over the fractures and matrix initial reservoir porosity in the dual-porosity simulations
POROSITY	–	Porosity, taking into account the rock compaction
PORVMULT	–	Pore volume multiplier due to rock compaction
PRES	–	Pressure
PRES$p	–	Pressure in phase p=1,2,3,...
PRESDP	–	Pressure in fractures minus pressure in matrix blocks in dual porosity simulations (degrees Kelvin/Celcius)
PRESEFF	–	Fluid pressure - Lithostatic pressure
PRESFAIL	–	Rock failure pressure (in brittle rocks)
PRESFDYN	–	Rock failure pressure (in brittle and ductile rocks)
PRESLITH	–	Lithostatic pressure
PRESSURE	–	Pressure (Synonym to PRES)
PRESREF	–	The reference pressure for the ground displacement computations
PRIID$v	–	ID of the primary variable v=1,2,3,...
PRIME$v	–	Value of the primary variable v=1,2,3,...
PUMPGRP	–	Name of the group to which belongs the pumping device
PUMPMODE	–	Pumping device mode
PUMPNAME	–	Pumping device name
PUMPTARG	–	Pumping device operational target
PVTNUM	–	Pressure-volume-temperature region number
PWVFLAG1	–	Flag for the porosity waves option
PWVLSCL	–	Compaction length, i.e length scale for the POROWAVE option
PWVPSCL	–	Characteristic pressure for the POROWAVE option
PWVTSCL	–	Characteristic time for the POROWAVE option

Kernel mnemonics		
REL$p	–	Relative permeability of phase p=1,2,3,...
REL%name	–	Relative permeability of phase 'name'
ROCKDEN	–	Rock density
ROCKENRG	–	Rock specific energy
ROCKNUM	–	Rock properties region number
SAT	–	Saturations of all phases (vector(P), where P is the maximum number of phases)
SAT$p	–	Saturation of phase p=1,2,3,...
SAT%name	–	Saturation of phase 'name'
SATNUM	–	Saturation functions region number
SHY1	–	Internal use
SHY2	–	Internal use
SHYFLAG1	–	Internal use
SHYFLAG2	–	Internal use
SIGMAV	–	Sigma factor in double porosity models
SKIN	–	Skin factor
SMIR#c	–	Point source mass injection rate of component c=1,2,3,...
SMIRALL	–	Point source mass injection rate, summed over all components
SMIT#c	–	Point source mass injection total of component c=1,2,3,...
SMITALL	–	Point source mass injection total, summed over all components
SMPR#c	–	Point sink mass production rate of component c=1,2,3,...
SMPRALL	–	Point sink mass production rate, summed over all components
SMPT#c	–	Point sink mass production total of component c=1,2,3,...
SMPTALL	–	Point sink mass production total, summed over all components
SPRECIP	–	Precipitate phase saturation.
SRCNAME	–	Point source name
SVIR	–	Source volume injection rate (at the injection tank conditions)
TANKGRP	–	Group to which belongs the stock tank
TANKNAME	–	Stock tank name

Kernel mnemonics		
TEMP	–	Temperature (degrees Kelvin)
TEMPC	–	Temperature (degrees Celsius)
TEMPDP	–	Temperature in fractures minus temperature in matrix blocks in dual porosity simulations (degrees Kelvin/Celcius)
TEMPREF	–	The reference temperature for the ground displacement computations
TEMPREL	–	Temperature (relative). Celcius (METRIC) or Fahrenheit (FIELD)
TIME	–	Time (days)
TIMEKY	–	Time (years*1e3)
TIMESEC	–	Time (seconds)
TIMEY	–	Time (years)
TPR#c	–	Throughput ratio for the component c=1,2,3,...
TPRSCALE	–	Throughput ratio scale (internal use)
TRAN	–	Transmissibility
TRANFRME	–	Fractures transmissibility multiplier (explicit)
TRANFRMT	–	Fractures transmissibility multiplier
TRANFRMX	–	Maximum transmissibility of fractures
TRANMULT	–	Transmissibility multiplier due to rock compaction
TRANX	–	Transmissibility along X axis
TRANY	–	Transmissibility along Y axis
TRANZ	–	Transmissibility along Z axis
TSTEP	–	Timestep (days)
TSTEPSEC	–	Timestep (seconds)
TSTEPY	–	Timestep (years)
TTRAN	–	Thermal transmissibility (for modeling heat conduction)
TYPENUM	–	Cell type ID. Available values for grid blocks: 1 - an ordinary block, 2 - an impermeable grid block in which only heat conduction equation is solved)
UX	–	Ground displacement (X-component)
UY	–	Ground displacement (Y-component)
UZ	–	Ground displacement (Z-component). Positive values correspond to the surface uplift and negative values to its subsidence.

Kernel mnemonics		
UTOT	–	Ground displacement (absolute value)
VELTABS	–	Total velocity (absolute value)
VELTX	–	Total velocity in X,Y or Z direction
VELTY	–	Total velocity in X,Y or Z direction
VELTZ	–	Total velocity in X,Y or Z direction
VELX$p	–	Velocity of phase p in the X direction
VELY$p	–	Velocity of phase p in the Y direction
VELZ$p	–	Velocity of phase p in the Z direction
VERSION	–	Program version
VFLUXN$p	–	Volumetric flux of phase p=1,2,3 in negative direction
VFLUXP$p	–	Volumetric flux of phase p=1,2,3 in positive direction
VFLZN$p	–	Volumetric flux of phase p=1,2,3 in the Z- direction
VFLZP$p	–	Volumetric flux of phase p=1,2,3 in the Z+ direction
VIS$p	–	Viscosity of phase p=1,2,3,...
VIS%name	–	Viscosity of phase 'name'
VISRMULT	–	Rock (i.e. solid phase) viscosity multiplier
VISROCK	–	Rock (i.e. solid phase) viscosity
VISROCKE	–	Bulk (i.e. effective) rock viscosity
VMM%name	–	Maximum volume concentration of additional component 'name'
VMT%name	–	Total volume concentration of additional component 'name' (primary variable)
VOLM$p	–	Specific volume of phase p=1,2,3,...
VOLX$p	–	Molar volume of phase p=1,2,3,...
VPRECIP	–	Precipitate phase viscosity.
WABSLVL	–	Absolute static liquid level in the wellbore
WBHDEPTH	–	Well bottom-hole depth (datum depth)
WBHP	–	Well bottom-hole pressure
WBHPRATG	–	Well rate in gas units
WBHPRATL	–	Well rate in liquid units
WBHT	–	Well bottom-hole temperature (degrees Kelvin)
WBHTC	–	Well bottom-hole temperature (degrees Celcius)
WDENT	–	Total fluid density at the bottom-hole depth
WEFAC	–	Well efficiency factor

Kernel mnemonics		
WELLNAME	–	Well name
WMIR#c	–	Well mass injection rate of component c=1,2,3,...
WMIRALL	–	Summed over all components mass injection rate
WMIT#c	–	Well mass injection total of component c=1,2,3,...
WMITALL	–	Summed over all components mass injection total
WMPI#c	–	Well mass potential injection of component c=1,2,3,...
WMPIALL	–	Summed over all components mass potential injection
WMPP#c	–	Well mass production potential of component c=1,2,3,...
WMPR#c	–	Well mass production rate of component c=1,2,3,...
WMPPALL	–	Summed over all components mass production potential
WMPRALL	–	Summed over all components mass production rate
WMPT#c	–	Well mass production total of component c=1,2,3,...
WMPTALL	–	Summed over all components mass production total
WOSFLAG	–	Well mode
WRELLVL	–	Relative to the tubing-head depth liquid level in the well-bore
WSEPNAME	–	Separator name
WTHDEPTH	–	Well tubing head depth
WTHP	–	Well tubing-head pressure
WTHPLIM	–	Limit (or target) of the well tubing-head pressure
WVFPID	–	Number of the VFP table associated with the well
X#c$p	–	Component c=1,2,3,... molar fraction in phase p=1,2,3,...
XCOORD	–	X coordinate of the cell center
XCORNG	–	Corner point coordinate x (grid axes)
XCORNM	–	Corner point coordinate x (map axes)
XOBS	–	X coordinate of the observation point (map axes)
YCOORD	–	Y coordinate of the cell center
YCORNG	–	Corner point coordinate y (grid axes)
YOBS	–	Y coordinate of the observation point (map axes)
YCORNM	–	Corner point coordinate y (map axes)
YEAR	–	Calendar year
ZCOORD	–	Z coordinate of the cell center
ZCORN	–	Corner point depth

Kernel mnemonics		
ZOBS	–	Z coordinate of the observation point (map axes)

Additional kernel mnemonics for PHASES option		
CMc#name	–	Component c=1,2,3,... mass fraction in phase 'name'
CPc#name	–	Component c=1,2,3,... molar fraction in phase 'name'
DEN#name	–	Density of phase 'name'
ETH#name	–	Molar enthalpy of phase 'name'
MCc#name	–	Mass of connate component c=1,2,3,... in phase 'name'
MFR#name	–	Molar fraction of phase 'name'
MMc#name	–	Mass of mobile component c=1,2,3,... in phase 'name'
MSc#name	–	Mass of component c=1,2,3,... in phase 'name'
SAT#name	–	Saturation of phase 'name'
VIS#name	–	Viscosity of phase 'name'

Table of available options for operations on kernel arrays

Mnemonic	Array	Input					Output				
		GRID	EDIT	PROPS	INIT	SCHEDULE	GRID	EDIT	PROPS	INIT	SCHEDULE
ACTNUM	1	+	+	+	+	+	+	+	+	+	+
AREA	3	+	−	−	−	−	+	−	−	−	−
AREACONN	2	+	−	−	−	−	−	−	−	−	−
C#c$p	1	−	−	−	+	+	−	−	−	−	−
CAP%name	1	−	−	−	−	+	−	−	−	−	−
CBLKROCK	1	−	−	+	+	+	−	−	+	+	+
CELLID	1	+	+	−	+	+	−	−	−	−	−
CELLVOL	1	+	+	−	+	+	+	+	−	+	+
COMPTM#c	1	−	−	−	+	+	−	−	−	−	−
COMPTX#c	1	−	−	−	+	+	−	−	−	−	−
CPERMULT	1	−	−	−	+	+	−	−	−	+	+
CPORMULT	1	−	−	−	+	+	−	−	−	+	+
DDENGRAV	1	−	−	−	−	+	−	−	−	−	−
DENT	1	−	−	−	+	+	−	−	−	−	−
DEPTH	1	+	+	−	+	+	+	+	−	+	+
DEPTHQ	2	+	−	+	+	+	−	−	−	−	−
DIFFMI	1	+	−	−	−	−	+	−	−	−	−
DIFFMJ	1	+	−	−	−	−	+	−	−	−	−
DIFFMK	1	+	−	−	−	−	+	−	−	−	−
DIFFMMF	1	+	−	−	−	−	+	−	−	−	−
DIFFMX	1	+	−	−	−	−	+	−	−	−	−
DIFFMY	1	+	−	−	−	−	+	−	−	−	−
DIFFMZ	1	+	−	−	−	−	+	−	−	−	−
DIRNUM	2	+	+	−	+	+	+	+	−	+	+
DISTANCE	3	+	−	−	−	−	+	−	−	−	−
DTRAN	2	−	+	−	+	+	−	+	−	+	+
DVOLDISP	1	−	−	−	−	+	−	−	−	−	−
DX	1	+	−	−	−	−	−	−	−	−	−

Table of available options for operations on kernel arrays

		Input					Output				
Mnemonic	Array	GRID	EDIT	PROPS	INIT	SCHEDULE	GRID	EDIT	PROPS	INIT	SCHEDULE
DY	1	+	−	−	−	−	−	−	−	−	−
DZ	1	+	−	−	−	−	−	−	−	−	−
DZMTRXV	1	+	−	−	−	−	+	−	−	−	−
EOSNUM	1	+	+	+	+	+	+	+	+	+	+
EQLNUM	1	+	+	−	+	+	+	+	−	+	+
FIPNUM	1	+	+	−	+	+	+	+	−	+	+
FLUXI#c	1	−	−	−	−	+	−	−	−	−	−
FLUXI#E	1	−	−	−	−	+	−	−	−	−	−
FLUXI#T	1	−	−	−	−	+	−	−	−	−	−
FLUXJ#c	1	−	−	−	−	+	−	−	−	−	−
FLUXJ#E	1	−	−	−	−	+	−	−	−	−	−
FLUXJ#T	1	−	−	−	−	+	−	−	−	−	−
FLUXK#c	1	−	−	−	−	+	−	−	−	−	−
FLUXK#E	1	−	−	−	−	+	−	−	−	−	−
FLUXK#T	1	−	−	−	−	+	−	−	−	−	−
FLUXNUM	1	+	+	+	+	+	+	+	+	+	+
FLUXDPT	1	−	−	−	+	+	−	−	−	−	−
HCONDCF	3	+	−	−	−	−	+	−	−	−	−
HCONDCFI	1	+	−	−	−	−	+	−	−	−	−
HCONDCFJ	1	+	−	−	−	−	+	−	−	−	−
HCONDCFK	1	+	−	−	−	−	+	−	−	−	−
HCONDCFX	1	+	−	−	−	−	+	−	−	−	−
HCONDCFY	1	+	−	−	−	−	+	−	−	−	−
HCONDCFZ	1	+	−	−	−	−	+	−	−	−	−
I-IJKRES	1	+	+	−	+	+	−	−	−	−	−
IMBNUM	1	+	+	+	+	+	+	+	+	+	+
INCONUM	1	+	+	+	+	+	+	+	+	+	+
ILOC	1	+	+	−	+	+	−	−	−	−	−

Table of available options for operations on kernel arrays

Mnemonic	Array	Input					Output				
		GRID	EDIT	PROPS	INIT	SCHEDULE	GRID	EDIT	PROPS	INIT	SCHEDULE
IRES	1	+	+	−	+	+	−	−	−	−	−
J-IJKRES	1	+	+	−	+	+	−	−	−	−	−
JLOC	1	+	+	−	+	+	−	−	−	−	−
JRES	1	+	+	−	+	+	−	−	−	−	−
K-IJKRES	1	+	+	−	+	+	−	−	−	−	−
KLOC	1	+	+	−	+	+	−	−	−	−	−
KRES	1	+	+	−	+	+	−	−	−	−	−
LX	1	+	−	−	−	−	+	−	−	−	−
LY	1	+	−	−	−	−	+	−	−	−	−
LZ	1	+	−	−	−	−	+	−	−	−	−
MASS#c	1	−	−	−	−	+	−	−	−	−	−
MINCNUM	1	+	+	+	+	+	−	−	−	−	−
MINCSIZE	1	+	−	−	−	−	+	−	−	−	−
MPINUM	1	+	+	+	+	+	+	+	+	+	+
MPIRANK	1	+	+	+	+	+	−	−	−	−	−
MTR#c	1	−	−	−	−	+	−	−	−	−	+
MTR%name	1	−	−	−	−	+	−	−	−	−	+
MULTPV	1	+	+	−	−	−	+	+	−	−	−
MULTX	1	+	+	−	−	−	+	+	−	+	+
MULTX-	1	+	+	−	−	−	+	+	−	+	+
MULTY	1	+	+	−	−	−	+	+	−	+	+
MULTY-	1	+	+	−	−	−	+	+	−	+	+
MULTZ	1	+	+	−	−	−	+	+	−	+	+
MULTZ-	1	+	+	−	−	−	+	+	−	+	+
NDTNUM	1	+	+	+	+	+	−	−	−	−	−
NKFACES	1	+	+	−	+	+	−	−	−	−	−
NTG	1	+	−	−	−	−	+	−	−	−	−
ORIGNUM	1	+	−	−	−	−	+	−	−	−	−

Table of available options for operations on kernel arrays

		Input					Output				
Mnemonic	Array	GRID	EDIT	PROPS	INIT	SCHEDULE	GRID	EDIT	PROPS	INIT	SCHEDULE
PBLKROCK	1	−	−	+	+	+	−	−	+	+	+
PERM	3	+	−	−	−	−	+	−	−	−	−
PERMI	1	+	−	−	−	−	+	−	−	−	−
PERMIFR	1	+	−	−	−	−	+	−	−	−	−
PERMJ	1	+	−	−	−	−	+	−	−	−	−
PERMJFR	1	+	−	−	−	−	+	−	−	−	−
PERMK	1	+	−	−	−	−	+	−	−	−	−
PERMKFR	1	+	−	−	−	−	+	−	−	−	−
PERMX	1	+	−	−	−	−	+	−	−	−	−
PERMXFR	1	+	−	−	−	−	+	−	−	−	−
PERMY	1	+	−	−	−	−	+	−	−	−	−
PERMYFR	1	+	−	−	−	−	+	−	−	−	−
PERMZ	1	+	−	−	−	−	+	−	−	−	−
PERMZFR	1	+	−	−	−	−	+	−	−	−	−
PHID$p	1	−	−	−	+	+	−	−	−	+	+
PHST	1	−	−	−	+	+	−	−	−	+	+
PHST0	1	−	−	−	+	+	−	−	−	+	+
POREVOL	1	+	+	−	+	+	−	−	−	−	−
PORO	1	+	+	−	+	+	+	+	−	+	+
POROAVG	1	+	+	−	+	+	−	−	−	−	−
PORVMULT	1	−	−	−	+	+	−	−	−	+	+
PRES	1	−	−	−	+	+	−	−	−	+	+
PRES$p	1	−	−	−	−	+	−	−	−	−	+
PRESDP	1	−	−	−	+	+	−	−	−	−	−
PRESEFF	1	+	−	+	+	+	+	−	+	+	+
PRESFAIL	1	+	−	+	+	+	+	−	+	+	+
PRESFDYN	1	−	−	−	−	+	−	−	−	−	−
PRESLITH	1	+	−	+	+	+	+	−	+	+	+

Table of available options for operations on kernel arrays

Mnemonic	Array	Input					Output				
		GRID	EDIT	PROPS	INIT	SCHEDULE	GRID	EDIT	PROPS	INIT	SCHEDULE
PRESSURE	1	−	−	−	+	+	−	−	−	+	+
PRIME$v	1	−	−	−	+	−	−	−	−	+	−
PVTNUM	1	+	+	+	+	+	+	+	+	+	+
ROCKNUM	1	+	+	+	+	+	+	+	+	+	+
SAT$p	1	−	−	−	+	+	−	−	−	+	+
SAT%name	1	−	−	−	+	+	−	−	−	+	+
SATNUM	1	+	+	+	+	+	+	+	+	+	+
SIGMAV	1	+	−	−	−	−	+	−	−	−	−
TEMP	1	−	−	−	+	+	−	−	−	+	+
TEMPC	1	−	−	−	+	+	−	−	−	+	+
TEMPDP	1	−	−	−	+	+	−	−	−	−	−
TIME	1	−	+	+	+	+	−	−	−	−	−
TIMEKY	1	−	+	+	+	+	−	−	−	−	−
TIMESEC	1	−	+	+	+	+	−	−	−	−	−
TIMEY	1	−	+	+	+	+	−	−	−	−	−
TPR#c	1	−	−	−	−	+	−	−	−	−	−
TPRSCALE	1	−	−	−	−	+	−	−	−	−	−
TRAN	2	−	+	−	+	+	−	+	−	+	+
TRANFRME	1	−	−	−	+	+	−	−	−	−	−
TRANFRMT	1	−	−	−	+	+	−	−	−	+	+
TRANFRMX	2	−	+	−	+	+	−	+	−	+	+
TSTEP	1	−	−	−	+	+	−	−	−	−	−
TSTEPSEC	1	−	−	−	+	+	−	−	−	−	−
TSTEPY	1	−	−	−	+	+	−	−	−	−	−
TTRAN	2	−	+	−	+	+	−	+	−	+	+
TYPENUM	1	+	+	+	+	+	−	−	−	−	−
VELTABS	1	−	−	−	−	+	−	−	−	−	−
VELTX	1	−	−	−	−	+	−	−	−	−	−

Table of available options for operations on kernel arrays												
		Input					Output					
Mnemonic	Array	GRID	EDIT	PROPS	INIT	SCHEDULE	GRID	EDIT	PROPS	INIT	SCHEDULE
VELTY	1	−	−	−	−	+	−	−	−	−	−
VELTZ	1	−	−	−	−	+	−	−	−	−	−
VELX$p	1	−	−	−	−	+	−	−	−	−	−
VELY$p	1	−	−	−	−	+	−	−	−	−	−
VELZ$p	1	−	−	−	−	+	−	−	−	−	−
VERSION	1	−	+	+	+	+	−	−	−	−	−
VISRMULT	1	−	−	−	+	+	−	−	−	+	+
VISROCK	1	−	−	−	+	+	−	−	−	+	+
VISROCKE	1	−	−	−	+	+	−	−	−	+	+
VMT%name	1	−	−	−	+	+	−	−	−	+	+
X#c$p	1	−	−	−	+	+	−	−	−	−	−
XCOORD	1	+	+	+	+	−	+	+	+	+	−
YCOORD	1	+	+	+	+	−	+	+	+	+	−
ZCOORD	1	+	+	−	+	+	+	+	−	+	+

Table of available options for output of kernel arrays

Mnemonic	RPTGRID							RPTSUM						
	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
ACG%name	−	−	−	−	−	−	−	+	−	−	−	−	−	−
ACL%name	−	−	−	−	−	−	−	+	−	−	−	−	−	−
ACTNUM	+	−	−	−	−	−	−	+	−	−	−	−	−	−
AREA	−	+	−	−	−	−	−	−	−	−	−	−	−	−
AREACONN	−	+	−	−	−	−	−	−	+	−	−	−	−	−
C#c$p	−	−	−	−	−	−	−	+	−	−	−	−	+	−
CAP%name	−	−	−	−	−	−	−	+	−	−	−	−	+	−
CBLKROCK	−	−	−	−	−	−	−	+	−	−	−	−	−	−
CELLID	+	−	−	−	−	−	−	+	−	−	−	−	−	−
CELLVOL	+	−	−	−	−	−	−	+	−	−	−	−	+	−
CHARGRP	+	+	−	−	−	−	−	+	+	−	−	−	−	−
CHARNAME	+	+	−	−	−	−	−	+	+	−	−	−	−	−
CEG%name	−	−	−	−	−	−	−	+	−	−	−	−	+	−
CEL%name	−	−	−	−	−	−	−	+	−	−	−	−	+	−
CHPT#c	−	−	−	−	−	−	−	+	−	−	−	−	−	−
CMG%name	−	−	−	−	−	−	−	+	−	−	−	−	+	−
CML%name	−	−	−	−	−	−	−	+	−	−	−	−	+	−
CMT%name	−	−	−	−	−	−	−	+	−	−	−	−	+	−
CNTNAME	−	+	−	−	−	−	−	−	+	−	−	−	−	−
CNTNUM	−	+	−	−	−	−	−	−	+	−	−	−	−	−
COMPTM#c	−	−	−	−	−	−	−	+	−	+	−	−	−	−
COMPTX#c	−	−	−	−	−	−	−	+	−	+	−	−	−	−
CONNID	−	+	−	−	−	−	−	−	+	−	−	−	−	−
CORNAUX1	−	−	−	−	−	−	−	−	−	−	−	−	−	−
CORNAUX2	−	−	−	−	−	−	−	−	−	−	−	−	−	−
CPERMULT	−	−	−	−	−	−	−	+	−	−	−	−	−	−
CPORMULT	−	−	−	−	−	−	−	+	−	−	−	−	−	−
CRFLFLAG	−	−	−	−	−	−	−	−	+	−	−	−	−	−

Table of available options for output of kernel arrays

	RPTGRID							RPTSUM						
Mnemonic	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
DATE	–	–	–	–	–	–	–	–	–	–	–	–	–	–
DAY	–	–	–	–	–	–	–	–	–	–	–	–	–	–
DCTTMULT	–	–	–	–	–	–	–	+	–	–	–	–	–	–
DDENGRAV	–	–	–	–	–	–	–	+	–	–	–	–	+	–
DENT	–	–	–	–	–	–	–	+	–	–	–	–	+	–
DENX$p	–	–	–	–	–	–	–	+	–	–	–	–	+	–
DEN$p	–	–	–	–	–	–	–	+	–	–	–	–	+	–
DEN%name	–	–	–	–	–	–	–	+	–	–	–	–	+	–
DEPTH	+	–	–	–	–	–	–	+	–	–	–	–	+	–
DEPTHOBS	–	–	–	–	–	–	–	–	–	–	–	–	–	–
DEPTHQ	–	+	–	–	–	–	–	–	+	–	–	–	–	–
DIFFMI	+	–	–	–	–	–	–	–	–	–	–	–	–	–
DIFFMJ	+	–	–	–	–	–	–	–	–	–	–	–	–	–
DIFFMK	+	–	–	–	–	–	–	–	–	–	–	–	–	–
DIFFMMF	+	–	–	–	–	–	–	–	–	–	–	–	–	–
DIFFMX	+	–	–	–	–	–	–	–	–	–	–	–	–	–
DIFFMY	+	–	–	–	–	–	–	–	–	–	–	–	–	–
DIFFMZ	+	–	–	–	–	–	–	–	–	–	–	–	–	–
DIRNUM	–	+	–	–	–	–	–	–	+	–	–	–	–	–
DISTANCE	–	+	–	–	–	–	–	–	–	–	–	–	–	–
DPCAP$p	–	–	–	–	–	–	–	+	–	–	–	–	+	–
DPNUM	–	–	–	–	–	–	–	–	–	–	–	–	–	–
DPRECIP	–	–	–	–	–	–	–	+	–	–	–	–	–	–
DTRAN	–	–	–	–	–	–	–	–	+	–	–	–	–	–
DVOLDISP	–	–	–	–	–	–	–	+	–	–	–	–	+	–
DX	+	–	–	–	–	–	–	–	–	–	–	–	–	–
DY	+	–	–	–	–	–	–	–	–	–	–	–	–	–
DZ	+	–	–	–	–	–	–	–	–	–	–	–	–	–

Table of available options for output of kernel arrays

Mnemonic	RPTGRID							RPTSUM						
	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
DZMTRXV	+	–	–	–	–	–	–	–	–	–	–	–	–	–
EOSNUM	+	–	–	–	–	–	–	+	–	–	–	–	–	–
EPRECIP	–	–	–	–	–	–	–	+	–	–	–	–	–	–
EQLNUM	+	–	–	–	–	–	–	+	–	–	–	–	–	–
ETM$p	–	–	–	–	–	–	–	+	–	–	–	–	+	–
ETM%name	–	–	–	–	–	–	–	+	–	–	–	–	+	–
ETX$p	–	–	–	–	–	–	–	+	–	–	–	–	+	–
GMIR#c	–	–	–	–	–	–	–	–	–	–	–	–	–	–
GMIRALL	–	–	–	–	–	–	–	–	–	–	–	–	–	–
GMIT#c	–	–	–	–	–	–	–	–	–	–	–	–	–	–
GMITALL	–	–	–	–	–	–	–	–	–	–	–	–	–	–
GMPI#c	–	–	–	–	–	–	–	–	–	–	–	–	–	–
GMPIALL	–	–	–	–	–	–	–	–	–	–	–	–	–	–
GMPP#c	–	–	–	–	–	–	–	–	–	–	–	–	–	–
GMPPALL	–	–	–	–	–	–	–	–	–	–	–	–	–	–
GMPR#c	–	–	–	–	–	–	–	–	–	–	–	–	–	–
GMPRALL	–	–	–	–	–	–	–	–	–	–	–	–	–	–
GMPT#c	–	–	–	–	–	–	–	–	–	–	–	–	–	–
GMPTALL	–	–	–	–	–	–	–	–	–	–	–	–	–	–
GRUPNAME	–	–	–	–	–	–	–	–	–	–	–	–	–	–
FIPCELL	–	–	–	–	–	–	–	–	–	–	–	–	+	–
FIPCONN	–	–	–	–	–	–	–	–	–	–	–	–	–	+
FIPNUM	+	–	–	–	–	–	–	+	–	–	–	–	–	–
FLUX#c	–	–	–	–	–	–	–	–	+	–	–	–	–	+
FLUXE	–	–	–	–	–	–	–	–	+	–	–	–	–	+
FLUXI#c	–	–	–	–	–	–	–	+	–	–	–	–	–	–
FLUXI#E	–	–	–	–	–	–	–	+	–	–	–	–	–	–
FLUXI#T	–	–	–	–	–	–	–	+	–	–	–	–	–	–

Table of available options for output of kernel arrays

	RPTGRID							RPTSUM						
Mnemonic	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
FLUXJ#c	–	–	–	–	–	–	–	+	–	–	–	–	–	–
FLUXJ#E	–	–	–	–	–	–	–	+	–	–	–	–	–	–
FLUXJ#T	–	–	–	–	–	–	–	+	–	–	–	–	–	–
FLUXK#c	–	–	–	–	–	–	–	+	–	–	–	–	–	–
FLUXK#E	–	–	–	–	–	–	–	+	–	–	–	–	–	–
FLUXK#T	–	–	–	–	–	–	–	+	–	–	–	–	–	–
FLUXNUM	+	–	–	–	–	–	–	+	–	–	–	–	–	–
FLUXDPT	–	–	–	–	–	–	–	+	–	–	–	–	–	–
FLUXT	–	–	–	–	–	–	–	–	+	–	–	–	–	+
FMIR#c	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FMIRALL	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FMIT#c	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FMITALL	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FMPR#c	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FMPRALL	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FMPT#c	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FMPTALL	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FSE#c	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FVIR	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FVIT	–	–	–	–	–	–	–	–	–	–	–	+	–	–
GBVMULT	–	–	–	–	–	–	–	+	–	–	–	–	–	–
GRAVMT	–	–	–	–	–	–	–	–	–	–	–	–	–	–
GRAVMTX	–	–	–	–	–	–	–	–	–	–	–	–	–	–
GRAVMTY	–	–	–	–	–	–	–	–	–	–	–	–	–	–
GRAVMTZ	–	–	–	–	–	–	–	–	–	–	–	–	–	–
HCONDCF	–	+	–	–	–	–	–	–	–	–	–	–	–	–
HCONDCFI	+	–	–	–	–	–	–	–	–	–	–	–	–	–
HCONDCFJ	+	–	–	–	–	–	–	–	–	–	–	–	–	–

Table of available options for output of kernel arrays														
	RPTGRID							RPTSUM						
Mnemonic	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
HCONDCFK	+	−	−	−	−	−	−	−	−	−	−	−	−	−
HCONDCFX	+	−	−	−	−	−	−	−	−	−	−	−	−	−
HCONDCFY	+	−	−	−	−	−	−	−	−	−	−	−	−	−
HCONDCFZ	+	−	−	−	−	−	−	−	−	−	−	−	−	−
I-IJKRES	+	−	−	−	−	−	−	+	−	−	−	−	−	−
IMBNUM	+	−	−	−	−	−	−	+	−	−	−	−	−	−
INCONUM	+	−	−	−	−	−	−	+	−	−	−	−	−	−
ILOC	+	−	−	−	−	−	−	+	−	−	−	−	−	−
IRES	+	−	−	−	−	−	−	+	−	−	−	−	−	−
J-IJKRES	+	−	−	−	−	−	−	+	−	−	−	−	−	−
JLOC	+	−	−	−	−	−	−	+	−	−	−	−	−	−
JRES	+	−	−	−	−	−	−	+	−	−	−	−	−	−
K-IJKRES	+	−	−	−	−	−	−	+	−	−	−	−	−	−
KLOC	+	−	−	−	−	−	−	+	−	−	−	−	−	−
KRES	+	−	−	−	−	−	−	+	−	−	−	−	−	−
KVL%name	−	−	−	−	−	−	−	+	−	−	−	−	+	−
LX	+	−	−	−	−	−	−	−	−	−	−	−	−	−
LY	+	−	−	−	−	−	−	−	−	−	−	−	−	−
LZ	+	−	−	−	−	−	−	−	−	−	−	−	−	−
MASS#c	−	−	−	−	−	−	−	+	−	−	−	+	+	−
MFRAC$p	−	−	−	−	−	−	−	+	−	−	−	−	+	−
MINCNUM	+	−	−	−	−	−	−	+	−	−	−	−	−	−
MINCSIZE	+	−	−	−	−	−	−	−	−	−	−	−	−	−
MOBT	−	−	−	−	−	−	−	+	−	−	−	−	+	−
MOB$p	−	−	−	−	−	−	−	+	−	−	−	−	+	−
MOLDEN	−	−	−	−	−	−	−	−	−	−	−	−	−	−
MONTH	−	−	−	−	−	−	−	−	−	−	−	−	−	−
MPINUM	+	−	−	−	−	−	−	+	−	−	−	−	−	−

Table of available options for output of kernel arrays														
	RPTGRID							RPTSUM						
Mnemonic	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
MPIRANK	+	−	−	−	−	−	−	+	−	−	−	−	−	−
MTR#c	−	−	−	−	−	−	−	+	−	−	−	−	+	−
MTR%name	−	−	−	−	−	−	−	+	−	−	−	−	+	−
MULTPV	+	−	−	−	−	−	−	−	−	−	−	−	−	−
MULTX	+	−	−	−	−	−	−	+	−	−	−	−	−	−
MULTX-	+	−	−	−	−	−	−	+	−	−	−	−	−	−
MULTY	+	−	−	−	−	−	−	+	−	−	−	−	−	−
MULTY-	+	−	−	−	−	−	−	+	−	−	−	−	−	−
MULTZ	+	−	−	−	−	−	−	+	−	−	−	−	−	−
MULTZ-	+	−	−	−	−	−	−	+	−	−	−	−	−	−
NDTNAME	+	−	−	−	−	−	−	+	−	−	−	−	−	−
NDTNUM	+	−	−	−	−	−	−	+	−	−	−	−	−	−
NEQNUM	+	−	−	−	−	−	−	+	−	−	−	−	−	−
NKFACES	+	−	−	−	−	−	−	+	−	−	−	−	−	−
NTG	+	−	−	−	−	−	−	+	−	−	−	−	−	−
OBSID	−	−	−	−	−	−	−	−	−	−	−	−	−	−
ORIGNUM	+	−	−	−	−	−	−	+	−	−	−	−	−	−
PBLKROCK	−	−	−	−	−	−	−	+	−	−	−	−	−	−
PERM	−	+	−	−	−	−	−	−	−	−	−	−	−	−
PERMI	+	−	−	−	−	−	−	−	−	−	−	−	−	−
PERMIFR	+	−	−	−	−	−	−	−	−	−	−	−	−	−
PERMJ	+	−	−	−	−	−	−	−	−	−	−	−	−	−
PERMJFR	+	−	−	−	−	−	−	−	−	−	−	−	−	−
PERMK	+	−	−	−	−	−	−	−	−	−	−	−	−	−
PERMKFR	+	−	−	−	−	−	−	−	−	−	−	−	−	−
PERMX	+	−	−	−	−	−	−	−	−	−	−	−	−	−
PERMXFR	+	−	−	−	−	−	−	−	−	−	−	−	−	−
PERMY	+	−	−	−	−	−	−	−	−	−	−	−	−	−

Table of available options for output of kernel arrays

Mnemonic	RPTGRID CELLDATA	RPTGRID CONNDATA	RPTGRID SRCDATA	RPTGRID WELLDATA	RPTGRID TOTDATA	RPTGRID FPCEDATA	RPTGRID FPCODATA	RPTSUM CELLDATA	RPTSUM CONNDATA	RPTSUM SRCDATA	RPTSUM WELLDATA	RPTSUM TOTDATA	RPTSUM FPCEDATA	RPTSUM FPCODATA
PERMYFR	+	−	−	−	−	−	−	−	−	−	−	−	−	−
PERMZ	+	−	−	−	−	−	−	−	−	−	−	−	−	−
PERMZFR	+	−	−	−	−	−	−	−	−	−	−	−	−	−
PFLDFACT	−	−	−	−	−	−	−	+	−	−	−	−	−	−
PHID$p	−	−	−	−	−	−	−	+	−	−	−	−	−	−
PHST	−	−	−	−	−	−	−	+	−	−	−	−	−	−
PHST0	−	−	−	−	−	−	−	+	−	−	−	−	−	−
PJDIAM	−	+	−	−	−	−	−	−	+	−	−	−	−	−
PJLENGTH	−	+	−	−	−	−	−	−	+	−	−	−	−	−
PJROUGH	−	+	−	−	−	−	−	−	+	−	−	−	−	−
POREVOL	+	−	−	−	−	−	−	+	−	−	−	−	+	−
PORO	+	−	−	−	−	−	−	+	−	−	−	−	−	−
POROAVG	+	−	−	−	−	−	−	+	−	−	−	−	−	−
POROSITY	−	−	−	−	−	−	−	+	−	−	−	−	+	−
PORVMULT	−	−	−	−	−	−	−	+	−	−	−	−	+	−
PRES	−	−	−	−	−	−	−	+	−	−	+	−	+	−
PRES$p	−	−	−	−	−	−	−	+	−	−	−	−	+	−
PRESDP	−	−	−	−	−	−	−	+	−	−	−	−	+	−
PRESEFF	+	−	−	−	−	−	−	+	−	−	−	−	−	−
PRESFAIL	+	−	−	−	−	−	−	+	−	−	−	−	−	−
PRESFDYN	−	−	−	−	−	−	−	+	−	−	−	−	−	−
PRESLITH	+	−	−	−	−	−	−	+	−	−	−	−	−	−
PRESSURE	−	−	−	−	−	−	−	+	−	−	+	−	+	−
PRESREF	−	−	−	−	−	−	−	+	−	−	−	−	−	−
PRIID$v	−	−	−	−	−	−	−	+	−	−	−	−	−	−
PRIME$v	−	−	−	−	−	−	−	+	−	−	−	−	−	−
PUMPGRP	−	+	−	−	−	−	−	−	+	−	−	−	−	−
PUMPMODE	−	+	−	−	−	−	−	−	+	−	−	−	−	−

Table of available options for output of kernel arrays

	RPTGRID							RPTSUM						
Mnemonic	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
PUMPNAME	–	+	–	–	–	–	–	–	+	–	–	–	–	–
PUMPTARG	–	+	–	–	–	–	–	–	+	–	–	–	–	–
PVTNUM	+	–	–	–	–	–	–	+	–	–	–	–	–	–
PWVFLAG1	–	–	–	–	–	–	–	+	–	–	–	–	–	–
PWVLSCL	–	–	–	–	–	–	–	+	–	–	–	–	–	–
PWVPSCL	–	–	–	–	–	–	–	+	–	–	–	–	–	–
PWVTSCL	–	–	–	–	–	–	–	+	–	–	–	–	–	–
REL$p	–	–	–	–	–	–	–	+	–	–	–	–	+	–
REL%name	–	–	–	–	–	–	–	+	–	–	–	–	+	–
ROCKDEN	–	–	–	–	–	–	–	+	–	–	–	–	–	–
ROCKENRG	–	–	–	–	–	–	–	+	–	–	–	–	–	–
ROCKNUM	+	–	–	–	–	–	–	+	–	–	–	–	–	–
SAT	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SAT$p	–	–	–	–	–	–	–	+	–	–	–	–	+	–
SAT%name	–	–	–	–	–	–	–	+	–	–	–	–	+	–
SATNUM	+	–	–	–	–	–	–	+	–	–	–	–	–	–
SHY1	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SHY2	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SHYFLAG1	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SHYFLAG2	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SIGMAV	+	–	–	–	–	–	–	–	–	–	–	–	–	–
SKIN	–	–	–	–	–	–	–	–	+	–	–	–	–	–
SMIR#c	–	–	–	–	–	–	–	–	–	+	–	–	–	–
SMIRALL	–	–	–	–	–	–	–	–	–	+	–	–	–	–
SMIT#c	–	–	–	–	–	–	–	–	–	+	–	–	–	–
SMITALL	–	–	–	–	–	–	–	–	–	+	–	–	–	–
SMPR#c	–	–	–	–	–	–	–	–	–	+	–	–	–	–
SMPRALL	–	–	–	–	–	–	–	–	–	+	–	–	–	–

Table of available options for output of kernel arrays

Mnemonic	RPTGRID							RPTSUM						
	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
SMPT#c	−	−	−	−	−	−	−	−	−	+	−	−	−	−
SMPTALL	−	−	−	−	−	−	−	−	−	+	−	−	−	−
SPRECIP	−	−	−	−	−	−	−	+	−	−	−	−	−	−
SRCNAME	−	−	+	−	−	−	−	−	−	+	−	−	−	−
SVIR	−	−	−	−	−	−	−	−	−	+	−	−	−	−
TANKGRP	+	−	−	−	−	−	−	+	−	−	−	−	−	−
TANKNAME	+	−	−	−	−	−	−	+	−	−	−	−	−	−
TEMP	−	−	−	−	−	−	−	+	−	−	−	−	+	−
TEMPC	−	−	−	−	−	−	−	+	−	−	−	−	+	−
TEMPDP	−	−	−	−	−	−	−	+	−	−	−	−	+	−
TEMPREF	−	−	−	−	−	−	−	+	−	−	−	−	−	−
TEMPREL	−	−	−	−	−	−	−	+	−	−	−	−	−	−
TIME	−	−	−	−	−	−	−	+	+	−	−	−	−	−
TIMEKY	−	−	−	−	−	−	−	+	+	−	−	−	−	−
TIMESEC	−	−	−	−	−	−	−	+	+	−	−	−	−	−
TIMEY	−	−	−	−	−	−	−	+	+	−	−	−	−	−
TPR#c	−	−	−	−	−	−	−	+	−	−	−	−	+	−
TPRSCALE	−	−	−	−	−	−	−	+	−	−	−	−	−	−
TRAN	−	−	−	−	−	−	−	−	+	−	−	−	−	−
TRANFRME	−	−	−	−	−	−	−	+	−	−	−	−	−	−
TRANFRMT	−	−	−	−	−	−	−	+	−	−	−	−	−	−
TRANFRMX	−	−	−	−	−	−	−	−	+	−	−	−	−	−
TRANMULT	−	−	−	−	−	−	−	+	−	−	−	−	−	−
TRANX	−	−	−	−	−	−	−	+	−	−	−	−	−	−
TRANY	−	−	−	−	−	−	−	+	−	−	−	−	−	−
TRANZ	−	−	−	−	−	−	−	+	−	−	−	−	−	−
TSTEP	−	−	−	−	−	−	−	+	+	−	−	+	−	−
TSTEPSEC	−	−	−	−	−	−	−	+	+	−	−	+	−	−

Table of available options for output of kernel arrays

	RPTGRID							RPTSUM						
Mnemonic	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
TSTEPY	–	–	–	–	–	–	–	+	+	–	–	+	–	–
TTRAN	–	–	–	–	–	–	–	–	+	–	–	–	–	–
TYPENUM	+	–	–	–	–	–	–	+	–	–	–	–	–	–
UX	–	–	–	–	–	–	–	–	–	–	–	–	–	–
UY	–	–	–	–	–	–	–	–	–	–	–	–	–	–
UZ	–	–	–	–	–	–	–	–	–	–	–	–	–	–
UTOT	–	–	–	–	–	–	–	–	–	–	–	–	–	–
VELTABS	–	–	–	–	–	–	–	+	–	–	–	–	–	–
VELTX	–	–	–	–	–	–	–	+	–	–	–	–	–	–
VELTY	–	–	–	–	–	–	–	+	–	–	–	–	–	–
VELTZ	–	–	–	–	–	–	–	+	–	–	–	–	–	–
VELX$p	–	–	–	–	–	–	–	+	–	–	–	–	–	–
VELY$p	–	–	–	–	–	–	–	+	–	–	–	–	–	–
VELZ$p	–	–	–	–	–	–	–	+	–	–	–	–	–	–
VERSION	–	–	–	–	–	–	–	+	+	–	–	–	–	–
VFLUXN$p	–	–	–	–	–	–	–	–	+	–	–	–	–	–
VFLUXP$p	–	–	–	–	–	–	–	–	+	–	–	–	–	–
VFLZN$p	–	–	–	–	–	–	–	+	–	–	–	–	–	–
VFLZP$p	–	–	–	–	–	–	–	+	–	–	–	–	–	–
VIS$p	–	–	–	–	–	–	–	+	–	–	–	–	+	–
VIS%name	–	–	–	–	–	–	–	+	–	–	–	–	+	–
VISRMULT	–	–	–	–	–	–	–	+	–	–	–	–	–	–
VISROCK	–	–	–	–	–	–	–	+	–	–	–	–	–	–
VISROCKE	–	–	–	–	–	–	–	+	–	–	–	–	–	–
VMM%name	–	–	–	–	–	–	–	+	–	–	–	–	+	–
VMT%name	–	–	–	–	–	–	–	+	–	–	–	–	+	–
VOLM$p	–	–	–	–	–	–	–	+	–	–	–	–	+	–
VOLX$p	–	–	–	–	–	–	–	+	–	–	–	–	+	–

Table of available options for output of kernel arrays

Mnemonic	RPTGRID							RPTSUM						
	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
VPRECIP	–	–	–	–	–	–	–	+	–	–	–	–	–	–
WABSLVL	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WBHDEPTH	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WBHP	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WBHPRATG	–	–	–	–	–	–	–	–	+	–	–	–	–	–
WBHPRATL	–	–	–	–	–	–	–	–	+	–	–	–	–	–
WBHT	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WBHTC	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WDENT	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WEFAC	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WELLNAME	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WMIR#c	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WMIRALL	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WMIT#c	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WMITALL	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WMPI#c	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WMPIALL	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WMPP#c	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WMPR#c	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WMPPALL	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WMPRALL	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WMPT#c	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WMPTALL	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WOSFLAG	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WRELLVL	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WSEPNAME	–	–	–	–	–	–	–	–	–	+	+	–	–	–
WTHDEPTH	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WTHP	–	–	–	–	–	–	–	–	–	–	+	–	–	–

Table of available options for output of kernel arrays														
	RPTGRID							RPTSUM						
Mnemonic	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
WTHPLIM	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WVFPID	–	–	–	–	–	–	–	–	–	–	+	–	–	–
X#c$p	–	–	–	–	–	–	–	+	–	–	–	–	+	–
XCOORD	+	–	–	–	–	–	–	–	–	–	–	–	–	–
XCORNG	–	–	–	–	–	–	–	–	–	–	–	–	–	–
XCORNM	–	–	–	–	–	–	–	–	–	–	–	–	–	–
XOBS	–	–	–	–	–	–	–	–	–	–	–	–	–	–
YCOORD	+	–	–	–	–	–	–	–	–	–	–	–	–	–
YCORNG	–	–	–	–	–	–	–	–	–	–	–	–	–	–
YOBS	–	–	–	–	–	–	–	–	–	–	–	–	–	–
YCORNM	–	–	–	–	–	–	–	–	–	–	–	–	–	–
YEAR	–	–	–	–	–	–	–	–	–	–	–	–	–	–
ZCOORD	+	–	–	–	–	–	–	+	–	–	–	–	+	–
ZCORN	–	–	–	–	–	–	–	–	–	–	–	–	–	–
ZOBS	–	–	–	–	–	–	–	–	–	–	–	–	–	–

2.3. BINMIXT mnemonics

Additional mnemonics for BINMIXT module		
CMP1$p	–	Molar fraction of the 1st component in phase p=1,2,3,...
COMPT#c	–	Total molar fraction of the component c=1,2,3,...
COMP1T	–	Total molar fraction of the 1st component.
COMP1TF	–	Total molar fraction of the 1st component in liquid and gas phases, multiplied by porosity.
COMP1TMF	–	Total mass fraction of the 1st component in liquid and gas phases, multiplied by porosity.
COMP2T	–	Total molar fraction of the 2nd component.
COMP2TF	–	Total molar fraction of the 2nd component in liquid and gas phases, multiplied by porosity.
COMP2TMF	–	Total mass fraction of the 2nd component in liquid and gas phases, multiplied by porosity.
CUMFLUX1	–	Cumulative flux of the 1st component.
CUMFLUX2	–	Cumulative flux of the 2nd component.
CUMFLUXE	–	Cumulative flux of energy.
DIST12	–	Distance in thermodynamic space between phases 1 and 2
DIST13	–	Distance in thermodynamic space between phases 1 and 3
DIST23	–	Distance in thermodynamic space between phases 2 and 3
ECNDROCK	–	Electrical conductivity of saturated rock.
ENRGT	–	Total molar energy of the reservoir fluid.
ENTHT	–	Total molar enthalpy of the reservoir fluid.
ERESROCK	–	Electrical resisitivity of saturated rock.
FLUX1	–	Flux of the 1st component (defined for connections).
FLUX1I	–	Estimation of the 1st component flux along i-index direction (defined in cells).
FLUX1J	–	Estimation of the 1st component flux along j-index direction (defined in cells).
FLUX1K	–	Estimation of the 1st component flux along k-index direction (defined in cells).
FLUX2	–	Flux of the 2nd component (defined for connections).
FLUX2I	–	Estimation of the 2nd component flux along i-index direction (defined in cells).

Additional mnemonics for BINMIXT module		
FLUX2J	–	Estimation of the 2nd component flux along j-index direction (defined in cells).
FLUX2K	–	Estimation of the 2nd component flux along k-index direction (defined in cells).
FLXNTRCc	–	Tracer c=1,2,3,... flux between two connected cells in negative direction
FLXPTRCc	–	Tracer c=1,2,3,... flux between two connected cells in positive direction
HNORM	–	Normalized molar enthalpy
LPNORM	–	Normalized logarithm of pressure.
MASS1F	–	Total mass of the 1st component in liquid and gas phases
MASS2F	–	Total mass of the 2nd component in liquid and gas phases
SGASINIT	–	Initial gas phase saturation.
SLIQINIT	–	Initial liquid phase saturation.
SSOL	–	Solid phase saturation.
TNORM	–	Normalized temperature.
X1NORM	–	Normalized molar fraction of the 1st component
VFRAC$p	–	Molar fraction of phase p=1,2,3,...
VOLT	–	Total molar volume of the reservoir fluid.

Table of available options for operations on BINMIXT arrays

		Input					Output				
Mnemonic	Array	GRID	EDIT	PROPS	INIT	SCHEDULE	GRID	EDIT	PROPS	INIT	SCHEDULE
COMPT#c	1	−	−	−	+	+	−	−	−	−	−
COMP1T	1	−	−	−	+	+	−	−	−	+	+
COMP2T	1	−	−	−	+	+	−	−	−	+	+
ENRGT	1	−	−	−	−	+	−	−	−	−	−
ENTHT	1	−	−	−	+	+	−	−	−	+	+
FLUX1I	1	−	−	−	+	+	−	−	−	−	−
FLUX1J	1	−	−	−	+	+	−	−	−	−	−
FLUX1K	1	−	−	−	+	+	−	−	−	−	−
FLUX2I	1	−	−	−	+	+	−	−	−	−	−
FLUX2J	1	−	−	−	+	+	−	−	−	−	−
FLUX2K	1	−	−	−	+	+	−	−	−	−	−
SGASINIT	1	−	−	−	+	−	−	−	−	+	−
SLIQINIT	1	−	−	−	+	−	−	−	−	+	−
VOLT	1	−	−	−	−	+	−	−	−	−	−

Table of available options for output of BINMIXT arrays

	RPTGRID							RPTSUM						
Mnemonic	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
CMP1$p	−	−	−	−	−	−	−	+	−	−	−	−	+	−
COMPT#c	−	−	−	−	−	−	−	+	−	+	−	−	−	−
COMP1T	−	−	−	−	−	−	−	+	−	+	−	−	+	−
COMP1TF	−	−	−	−	−	−	−	+	−	+	−	−	+	−
COMP1TMF	−	−	−	−	−	−	−	+	−	+	−	−	+	−
COMP2T	−	−	−	−	−	−	−	+	−	+	−	−	+	−
COMP2TF	−	−	−	−	−	−	−	+	−	+	−	−	+	−
COMP2TMF	−	−	−	−	−	−	−	+	−	+	−	−	+	−
CUMFLUX1	−	−	−	−	−	−	−	−	−	+	−	−	−	+
CUMFLUX2	−	−	−	−	−	−	−	−	−	+	−	−	−	+
CUMFLUXE	−	−	−	−	−	−	−	−	−	+	−	−	−	+
DIST12	−	−	−	−	−	−	−	+	−	−	−	−	−	−
DIST13	−	−	−	−	−	−	−	+	−	−	−	−	−	−
DIST23	−	−	−	−	−	−	−	+	−	−	−	−	−	−
ECNDROCK	−	−	−	−	−	−	−	+	−	−	−	−	−	−
ENRGT	−	−	−	−	−	−	−	+	−	+	−	−	+	−
ENTHT	−	−	−	−	−	−	−	+	−	+	−	−	+	−
ERESROCK	−	−	−	−	−	−	−	+	−	−	−	−	−	−
FLUX1	−	−	−	−	−	−	−	−	+	+	−	−	−	+
FLUX1I	−	−	−	−	−	−	−	+	−	−	−	−	−	−
FLUX1J	−	−	−	−	−	−	−	+	−	−	−	−	−	−
FLUX1K	−	−	−	−	−	−	−	+	−	−	−	−	−	−
FLUX2	−	−	−	−	−	−	−	−	+	+	−	−	−	+
FLUX2I	−	−	−	−	−	−	−	+	−	−	−	−	−	−
FLUX2J	−	−	−	−	−	−	−	+	−	−	−	−	−	−
FLUX2K	−	−	−	−	−	−	−	+	−	−	−	−	−	−
FLXNTRCc	−	−	−	−	−	−	−	−	+	−	−	−	−	+
FLXPTRCc	−	−	−	−	−	−	−	−	+	−	−	−	−	+

Table of available options for output of BINMIXT arrays														
	RPTGRID							RPTSUM						
Mnemonic	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
HNORM	−	−	−	−	−	−	−	+	−	−	−	−	−	−
LPNORM	−	−	−	−	−	−	−	+	−	−	−	−	−	−
MASS1F	−	−	−	−	−	−	−	+	−	−	−	−	+	−
MASS2F	−	−	−	−	−	−	−	+	−	−	−	−	+	−
SGASINIT	−	−	−	−	−	−	−	−	−	−	−	−	−	−
SLIQINIT	−	−	−	−	−	−	−	−	−	−	−	−	−	−
SSOL	−	−	−	−	−	−	−	+	−	−	−	−	−	−
TNORM	−	−	−	−	−	−	−	+	−	−	−	−	−	−
X1NORM	−	−	−	−	−	−	−	+	−	−	−	−	−	−
VFRAC$p	−	−	−	−	−	−	−	+	−	−	−	−	+	−
VOLT	−	−	−	−	−	−	−	+	−	+	−	−	+	−

2.4. BLACKOIL mnemonics

Additional mnemonics for BLACKOIL module		
BGAS	–	Gas volume factor
BOIL	–	Oil volume factor
BWAT	–	Water volume factor
DGAS	–	Gas density
DOIL	–	Oil density
DWAT	–	Water density
FGIP	–	Gas in place
FGIPG	–	Gas in place (in gas phase)
FGIPGM	–	Gas in place (in gas phase; mass units)
FGIPMG	–	Gas in place (in gas phase); Accounts only for the mobile fraction of gas
FGIPMGM	–	Gas in place (in gas phase); Accounts only for the mobile fraction of gas (mass units)
FGIPRG	–	Gas in place (in gas phase); Accounts only for the trapped gas, i.e. the residual fraction of gas
FGIPRGM	–	Gas in place (in gas phase); Accounts only for the trapped gas, i.e. the residual fraction of gas (mass units)
FGIPL	–	Gas in place (in liquid phase)
FGIPLM	–	Gas in place (in liquid phase; mass units)
FGIPM	–	Gas in place (mass units)
FGIR	–	Gas injection rate
FGIRM	–	Gas injection rate (mass units)
FGIT	–	Gas injection total
FGITM	–	Gas injection total (mass units)
FGOR	–	Gas oil ratio
FGPR	–	Gas production rate
FGPRM	–	Gas production rate (mass units)
FGPT	–	Gas production total
FGPTM	–	Gas production total (mass units)
FLPR	–	Liquid production rate
FLPT	–	Liquid production total
FOE	–	Oil recovery factor

Additional mnemonics for BLACKOIL module		
FOIP	–	Oil in place
FOIPG	–	Oil in place (in gas phase)
FOIPGM	–	Oil in place (in gas phase; mass units)
FOIPL	–	Oil in place (in liquid phase)
FOIPLM	–	Oil in place (in liquid phase; mass units)
FOIPM	–	Oil in place (mass units)
FOIR	–	Oil injection rate
FOIRM	–	Oil injection rate (mass units)
FOIT	–	Oil injection total
FOITM	–	Oil injection total (mass units)
FOPR	–	Oil production rate
FOPRM	–	Oil production rate (mass units)
FOPT	–	Oil production total
FOPTM	–	Oil production total (mass units)
FWCT	–	Water cut
FWOR	–	Water-oil ratio
FWIP	–	Water in place
FWIPM	–	Water in place (mass units)
FWIR	–	Water injection rate
FWIRM	–	Water injection rate (mass units)
FWIT	–	Water injection total
FWITM	–	Water injection total (mass units)
FWPR	–	Water production rate
FWPRM	–	Water production rate (mass units)
FWPT	–	Water production total
FWPTM	–	Water production total (mass units)
GIPG	–	Gas-in-place in the gas phase (in the grid block)
GIPGM	–	Gas mass in the gas phase (in the grid block)
GIPL	–	Gas-in-place in the liquid phase (in the grid block)
GIPLM	–	Gas mass in the liquid phase (in the grid block)
GIP	–	Gas-in-place (in the grid block)
GIPM	–	Gas mass (in the grid block)
GIPMG	–	Mobile gas-in-place (in the grid block)
GIPMGM	–	Mass of the mobile gas in the gas phase (in the grid block)

Additional mnemonics for BLACKOIL module		
GIPRG	–	Residual gas-in-place (in the grid block)
GIPRGM	–	Mass of the residual gas in the gas phase (in the grid block)
GGIR	–	Group gas injection rate
GGIRM	–	Group gas injection rate (mass units)
GGPI	–	Group gas potential injection
GGPIM	–	Group gas potential injection (mass units)
IKRG	–	Scaled end-point gas relative permeability (imbibition curve)
IKRO	–	Scaled end-point oil relative permeability (imbibition curve)
IKROW	–	Scaled end-point oil relative permeability (imbibition curve)
IKRW	–	Scaled end-point water relative permeability (imbibition curve)
IPCG	–	Scaled maximum gas-oil capillary pressure (imbibition curve)
IPCW	–	Scaled maximum water-oil capillary pressure (imbibition curve)
ISGCR	–	Scaled critical gas saturation (imbibition curve)
ISGL	–	Scaled connate gas saturation (imbibition curve)
ISGLPC	–	Scaled connate gas saturation for capillary pressure (imbibition curve)
ISGU	–	Scaled saturation table maximum gas saturation (imbibition curve)
ISOGCR	–	Scaled oil-in-gas saturation (imbibition curve)
ISOWCR	–	Scaled critical oil-in-water saturation (imbibition curve)
ISOWL	–	Scaled saturation table minimum oil-in-water saturation (imbibition curve)
ISOWU	–	Scaled saturation table maximum oil-in-water saturation (imbibition curve)
ISWCR	–	Scaled critical water saturation (imbibition curve)
ISWL	–	Scaled connate water saturation (imbibition curve)
ISWLPC	–	Scaled connate water saturation for capillary pressure (imbibition curve)
ISWU	–	Scaled saturation table maximum water saturation (imbibition curve)

Additional mnemonics for BLACKOIL module		
ISWUPC	–	Scaled saturation table maximum water saturation for capillary pressure (imbibition curve)
KRG	–	Scaled end-point gas relative permeability
KRO	–	Scaled end-point oil relative permeability
KROW	–	Scaled end-point oil relative permeability
KRW	–	Scaled end-point water relative permeability
MGIG	–	Mass of the gas component in the gas phase
MGIL	–	Mass of the gas component in the liquid phase
MGIMG	–	Mass of the gas component in the gas phase (mobile fraction of gas)
MGIRG	–	Mass of the gas component in the gas phase (immobile fraction of gas)
PBUB	–	Bubble point pressure
PCOG	–	Oil-gas capillary pressure
PCG	–	Scaled maximum gas capillary pressure
PCW	–	Scaled maximum water capillary pressure
PCWO	–	Water-oil capillary pressure
PDEW	–	Dew point pressure
PGAS	–	Pressure in gas phase
POIL	–	Pressure in oil phase
PWAT	–	Pressure in water phase
RELGAS	–	Gas phase relative permeability
RELOIL	–	Oil phase relative permeability
RELWAT	–	Water phase relative permeability
RPMGAS	–	Gas phase relative permeability (internal variable)
RPMOIL	–	Oil phase relative permeability (internal variable)
RPMWAT	–	Water phase relative permeability (internal variable)
RS	–	Gas in oil ratio
RV	–	Oil in gas ratio
SGAS	–	Gas saturation
SGASAVG	–	Averaged over the fractures and matrix gas saturation in dual-porosity simulations
SGCR	–	Scaled critical gas saturation
SGINI	–	Initial gas saturation (for gravity drainage)

Additional mnemonics for BLACKOIL module		
SGL	–	Scaled connate gas saturation
SGLPC	–	Scaled connate gas saturation for capillary pressure
SGU	–	Scaled saturation table maximum gas saturation
SOGCR	–	Scaled oil-in-gas saturation
SOIL	–	Oil saturation
SOILAVG	–	Averaged over the fractures and matrix oil saturation in dual-porosity simulations
SOWCR	–	Scaled critical oil-in-water saturation
SOWL	–	Scaled saturation table minimum oil-in-water saturation
SOWU	–	Scaled saturation table maximum oil-in-water saturation
SWAT	–	Water saturation
SWATAVG	–	Averaged over the fractires and matrix water saturation in dual-porosity simulations
SWATINIT	–	Initial water saturation
SWCR	–	Scaled critical water saturation
SWINI	–	Initial water saturation (for gravity drainage)
SWL	–	Scaled connate water saturation
SWLPC	–	Scaled connate water saturation for capillary pressure
SWU	–	Scaled saturation table maximum water saturation
SWUPC	–	Scaled saturation table maximum water saturation for capillary pressure
VISGAS	–	Gas viscosity
VISOIL	–	Oil viscosity
VISWAT	–	Water viscosity
WGIR	–	Well gas injection rate
WGIRM	–	Well gas injection rate (mass units)
WGIT	–	Well gas injection total
WGITM	–	Well gas injection total (mass units)
WGOR	–	Well gas-oil ratio
WGPI	–	Well gas potential injection
WGPIM	–	Well gas potential injection (mass units)
WGPR	–	Well gas production rate
WGPRM	–	Well gas production rate (mass units)
WGPT	–	Well gas production total

Additional mnemonics for BLACKOIL module		
WGPTM	–	Well gas production total (mass units)
WLPR	–	Well liquid production rate
WLPT	–	Well liquid production total
WOIR	–	Well oil injection rate
WOIRM	–	Well oil injection rate (mass units)
WOIT	–	Well oil injection total
WOITM	–	Well oil injection total (mass units)
WOPR	–	Well oil production rate
WOPRM	–	Well oil production rate (mass units)
WOPT	–	Well oil production total
WOPTM	–	Well oil production total (mass units)
WWCT	–	Well water cut
WWIR	–	Well water injection rate
WWIRM	–	Well water injection rate (mass units)
WWIT	–	Well water injection total
WWITM	–	Well water injection total (mass units)
WWOR	–	Well water-oil ratio
WWPR	–	Well water producion rate
WWPRM	–	Well water producion rate (mass units)
WWPT	–	Well water production total
WWPTM	–	Well water production total (mass units)

Table of available options for operations on BLACKOIL arrays

Mnemonic	Array	Input					Output				
		GRID	EDIT	PROPS	INIT	SCHEDULE	GRID	EDIT	PROPS	INIT	SCHEDULE
BGAS	1	–	–	–	+	+	–	–	–	–	–
BOIL	1	–	–	–	+	+	–	–	–	–	–
BWAT	1	–	–	–	+	+	–	–	–	–	–
DGAS	1	–	–	–	+	+	–	–	–	–	–
DOIL	1	–	–	–	+	+	–	–	–	–	–
DWAT	1	–	–	–	+	+	–	–	–	–	–
GIPG	1	–	–	–	–	+	–	–	–	–	–
GIPGM	1	–	–	–	–	+	–	–	–	–	–
GIPL	1	–	–	–	–	+	–	–	–	–	–
GIPLM	1	–	–	–	–	+	–	–	–	–	–
GIP	1	–	–	–	–	+	–	–	–	–	–
GIPM	1	–	–	–	–	+	–	–	–	–	–
GIPMG	1	–	–	–	–	+	–	–	–	–	–
GIPMGM	1	–	–	–	–	+	–	–	–	–	–
GIPRG	1	–	–	–	–	+	–	–	–	–	–
GIPRGM	1	–	–	–	–	+	–	–	–	–	–
IKRG	1	–	–	+	+	+	–	–	+	–	–
IKRO	1	–	–	+	+	+	–	–	+	–	–
IKRW	1	–	–	+	+	+	–	–	+	–	–
IPCG	1	–	–	+	+	+	–	–	+	–	–
IPCW	1	–	–	+	+	+	–	–	+	–	–
ISGCR	1	–	–	+	+	+	–	–	+	–	–
ISGL	1	–	–	+	+	+	–	–	+	–	–
ISGLPC	1	–	–	+	+	+	–	–	+	–	–
ISGU	1	–	–	+	+	+	–	–	+	–	–
ISOGCR	1	–	–	+	+	+	–	–	+	–	–
ISOWCR	1	–	–	+	+	+	–	–	+	–	–
ISOWL	1	–	–	–	+	+	–	–	–	–	–

Table of available options for operations on BLACKOIL arrays

Mnemonic	Array	Input					Output				
		GRID	EDIT	PROPS	INIT	SCHEDULE	GRID	EDIT	PROPS	INIT	SCHEDULE
ISOWU	1	−	−	−	+	+	−	−	−	−	−
ISWCR	1	−	−	+	+	+	−	−	+	−	−
ISWL	1	−	−	+	+	+	−	−	+	−	−
ISWLPC	1	−	−	+	+	+	−	−	+	−	−
ISWU	1	−	−	+	+	+	−	−	+	−	−
KRG	1	−	−	+	+	+	−	−	+	−	−
KRO	1	−	−	+	+	+	−	−	+	−	−
KRW	1	−	−	+	+	+	−	−	+	−	−
MGIG	1	−	−	−	−	+	−	−	−	−	−
MGIL	1	−	−	−	−	+	−	−	−	−	−
MGIMG	1	−	−	−	−	+	−	−	−	−	−
MGIRG	1	−	−	−	−	+	−	−	−	−	−
PBUB	1	−	−	−	+	+	−	−	−	+	+
PCG	1	−	−	+	+	+	−	−	+	−	−
PCW	1	−	−	+	+	+	−	−	+	−	−
PDEW	1	−	−	−	+	+	−	−	−	+	+
PGAS	1	−	−	−	+	+	−	−	−	−	−
POIL	1	−	−	−	+	+	−	−	−	−	−
PWAT	1	−	−	−	+	+	−	−	−	−	−
RELGAS	1	−	−	−	+	+	−	−	−	+	+
RELOIL	1	−	−	−	+	+	−	−	−	+	+
RELWAT	1	−	−	−	+	+	−	−	−	+	+
RPMGAS	1	−	−	−	−	+	−	−	−	−	−
RPMOIL	1	−	−	−	−	+	−	−	−	−	−
RPMWAT	1	−	−	−	−	+	−	−	−	−	−
RS	1	−	−	−	+	+	−	−	−	+	+
RV	1	−	−	−	+	+	−	−	−	+	+
SGAS	1	−	−	−	+	+	−	−	−	+	+

Table of available options for operations on BLACKOIL arrays

Mnemonic	Array	Input					Output				
		GRID	EDIT	PROPS	INIT	SCHEDULE	GRID	EDIT	PROPS	INIT	SCHEDULE
SGASAVG	1	−	−	−	−	+	−	−	−	−	−
SGCR	1	−	−	+	+	+	−	−	+	−	−
SGL	1	−	−	+	+	+	−	−	+	−	−
SGLPC	1	−	−	+	+	+	−	−	+	−	−
SGU	1	−	−	+	+	+	−	−	+	−	−
SOGCR	1	−	−	+	+	+	−	−	+	−	−
SOIL	1	−	−	−	+	+	−	−	−	+	+
SOILAVG	1	−	−	−	−	+	−	−	−	−	+
SOWCR	1	−	−	+	+	+	−	−	+	−	−
SWAT	1	−	−	−	+	+	−	−	−	+	+
SWATAVG	1	−	−	−	−	+	−	−	−	−	−
SWATINIT	1	−	−	+	−	−	−	−	+	−	−
SWCR	1	−	−	+	+	+	−	−	+	−	−
SWL	1	−	−	+	+	+	−	−	+	−	−
SWLPC	1	−	−	+	+	+	−	−	+	−	−
SWU	1	−	−	+	+	+	−	−	+	−	−
VISGAS	1	−	−	−	+	+	−	−	−	+	+
VISOIL	1	−	−	−	+	+	−	−	−	+	+
VISWAT	1	−	−	−	+	+	−	−	−	+	+

Table of available options for output of BLACKOIL arrays

Mnemonic	RPTGRID							RPTSUM						
	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
BGAS	–	–	–	–	–	–	–	+	–	–	–	–	–	–
BOIL	–	–	–	–	–	–	–	+	–	–	–	–	–	–
BWAT	–	–	–	–	–	–	–	+	–	–	–	–	–	–
DGAS	–	–	–	–	–	–	–	+	–	–	–	–	–	–
DOIL	–	–	–	–	–	–	–	+	–	–	–	–	–	–
DWAT	–	–	–	–	–	–	–	+	–	–	–	–	–	–
FGIP	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIPG	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIPGM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIPMG	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIPMGM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIPRG	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIPRGM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIPL	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIPLM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIPM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIR	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIRM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIT	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGITM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGOR	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGPR	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGPRM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGPT	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGPTM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FLPR	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FLPT	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FOE	–	–	–	–	–	–	–	–	–	–	–	+	–	–

Table of available options for output of BLACKOIL arrays

	RPTGRID							RPTSUM						
Mnemonic	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
FOIP	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FOIPG	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FOIPGM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FOIPL	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FOIPLM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FOIPM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FOIR	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FOIRM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FOIT	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FOITM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FOPR	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FOPRM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FOPT	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FOPTM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FWCT	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FWOR	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FWIP	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FWIPM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FWIR	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FWIRM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FWIT	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FWITM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FWPR	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FWPRM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FWPT	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FWPTM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
GIPG	–	–	–	–	–	–	–	+	–	–	–	–	+	–
GIPGM	–	–	–	–	–	–	–	+	–	–	–	–	+	–

Table of available options for output of BLACKOIL arrays

Mnemonic	RPTGRID							RPTSUM						
	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
GIPL	−	−	−	−	−	−	−	+	−	−	−	−	+	−
GIPLM	−	−	−	−	−	−	−	+	−	−	−	−	+	−
GIP	−	−	−	−	−	−	−	+	−	−	−	−	+	−
GIPM	−	−	−	−	−	−	−	+	−	−	−	−	+	−
GIPMG	−	−	−	−	−	−	−	+	−	−	−	−	+	−
GIPMGM	−	−	−	−	−	−	−	+	−	−	−	−	+	−
GIPRG	−	−	−	−	−	−	−	+	−	−	−	−	+	−
GIPRGM	−	−	−	−	−	−	−	+	−	−	−	−	+	−
GGIR	−	−	−	−	−	−	−	−	−	−	−	−	−	−
GGIRM	−	−	−	−	−	−	−	−	−	−	−	−	−	−
GGPI	−	−	−	−	−	−	−	−	−	−	−	−	−	−
GGPIM	−	−	−	−	−	−	−	−	−	−	−	−	−	−
IKRG	−	−	−	−	−	−	−	+	−	−	−	−	−	−
IKRO	−	−	−	−	−	−	−	+	−	−	−	−	−	−
IKROW	−	−	−	−	−	−	−	+	−	−	−	−	−	−
IKRW	−	−	−	−	−	−	−	+	−	−	−	−	−	−
IPCG	−	−	−	−	−	−	−	+	−	−	−	−	−	−
IPCW	−	−	−	−	−	−	−	+	−	−	−	−	−	−
ISGCR	−	−	−	−	−	−	−	+	−	−	−	−	−	−
ISGL	−	−	−	−	−	−	−	+	−	−	−	−	−	−
ISGLPC	−	−	−	−	−	−	−	+	−	−	−	−	−	−
ISGU	−	−	−	−	−	−	−	+	−	−	−	−	−	−
ISOGCR	−	−	−	−	−	−	−	+	−	−	−	−	−	−
ISOWCR	−	−	−	−	−	−	−	+	−	−	−	−	−	−
ISOWL	−	−	−	−	−	−	−	+	−	−	−	−	−	−
ISOWU	−	−	−	−	−	−	−	+	−	−	−	−	−	−
ISWCR	−	−	−	−	−	−	−	+	−	−	−	−	−	−
ISWL	−	−	−	−	−	−	−	+	−	−	−	−	−	−

Table of available options for output of BLACKOIL arrays

	RPTGRID							RPTSUM						
Mnemonic	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
ISWLPC	–	–	–	–	–	–	–	+	–	–	–	–	–	–
ISWU	–	–	–	–	–	–	–	+	–	–	–	–	–	–
ISWUPC	–	–	–	–	–	–	–	+	–	–	–	–	–	–
KRG	–	–	–	–	–	–	–	+	–	–	–	–	–	–
KRO	–	–	–	–	–	–	–	+	–	–	–	–	–	–
KROW	–	–	–	–	–	–	–	+	–	–	–	–	–	–
KRW	–	–	–	–	–	–	–	+	–	–	–	–	–	–
MGIG	–	–	–	–	–	–	–	+	–	–	–	–	+	–
MGIL	–	–	–	–	–	–	–	+	–	–	–	–	+	–
MGIMG	–	–	–	–	–	–	–	+	–	–	–	–	+	–
MGIRG	–	–	–	–	–	–	–	+	–	–	–	–	+	–
PBUB	–	–	–	–	–	–	–	+	–	–	–	–	–	–
PCOG	–	–	–	–	–	–	–	+	–	–	–	–	–	–
PCG	–	–	–	–	–	–	–	+	–	–	–	–	–	–
PCW	–	–	–	–	–	–	–	+	–	–	–	–	–	–
PCWO	–	–	–	–	–	–	–	+	–	–	–	–	–	–
PDEW	–	–	–	–	–	–	–	+	–	–	–	–	–	–
PGAS	–	–	–	–	–	–	–	+	–	–	–	–	–	–
POIL	–	–	–	–	–	–	–	+	–	–	–	–	–	–
PWAT	–	–	–	–	–	–	–	+	–	–	–	–	–	–
RELGAS	–	–	–	–	–	–	–	+	–	–	–	–	–	–
RELOIL	–	–	–	–	–	–	–	+	–	–	–	–	–	–
RELWAT	–	–	–	–	–	–	–	+	–	–	–	–	–	–
RPMGAS	–	–	–	–	–	–	–	+	–	–	–	–	–	–
RPMOIL	–	–	–	–	–	–	–	+	–	–	–	–	–	–
RPMWAT	–	–	–	–	–	–	–	+	–	–	–	–	–	–
RS	–	–	–	–	–	–	–	+	–	–	–	–	–	–
RV	–	–	–	–	–	–	–	+	–	–	–	–	–	–

Table of available options for output of BLACKOIL arrays

Mnemonic	RPTGRID							RPTSUM						
	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
SGAS	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SGASAVG	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SGCR	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SGINI	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SGL	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SGLPC	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SGU	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SOGCR	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SOIL	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SOILAVG	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SOWCR	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SOWL	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SOWU	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SWAT	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SWATAVG	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SWATINIT	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SWCR	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SWINI	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SWL	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SWLPC	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SWU	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SWUPC	–	–	–	–	–	–	–	+	–	–	–	–	–	–
VISGAS	–	–	–	–	–	–	–	+	–	–	–	–	–	–
VISOIL	–	–	–	–	–	–	–	+	–	–	–	–	–	–
VISWAT	–	–	–	–	–	–	–	+	–	–	–	–	–	–
WGIR	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WGIRM	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WGIT	–	–	–	–	–	–	–	–	–	–	+	–	–	–

Table of available options for output of BLACKOIL arrays															
	RPTGRID							RPTSUM							
Mnemonic	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
WGITM	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WGOR	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WGPI	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WGPIM	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WGPR	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WGPRM	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WGPT	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WGPTM	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WLPR	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WLPT	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WOIR	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WOIRM	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WOIT	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WOITM	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WOPR	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WOPRM	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WOPT	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WOPTM	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WWCT	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WWIR	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WWIRM	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WWIT	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WWITM	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WWOR	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WWPR	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WWPRM	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WWPT	–	–	–	–	–	–	–	–	–	–	+	–	–	–

Table of available options for output of BLACKOIL arrays

Mnemonic	RPTGRID							RPTSUM						
	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
WWPTM	–	–	–	–	–	–	–	–	–	–	+	–	–	–

2.5. COMPS mnemonics

Additional mnemonics for COMPS module		
AMF$c	–	Molar concentraion of the component c=1,2,3,... in the aqueous phase
AMFM$c	–	Mass concentraion of the component c=1,2,3,... in the aqueous phase
BGAS	–	Gas volume factor
BGIP	–	Gas in place (in a grid block)
BOIL	–	Oil volume factor
BOIP	–	Oil in place (in a grid block)
BWAT	–	Water volume factor
DGAS	–	Gas density
DOIL	–	Oil density
DWAT	–	Water density
EOSERR	–	Error of the flash calculation
EOSITER	–	Number of iterations (EoS) used for properties prediction (both the stability and flash calculation)
FDGAS	–	Field vapor density
FDGASG	–	Field vapor density (production summed then separated)
FDOIL	–	Field stock tank oil density
FDOILG	–	Field stock tank oil density (production summed then separated)
FEE	–	Energy (natural gas) recovery factor
FEIP	–	Energy in place (only natural gas)
FEPR	–	Field energy production rate
FEPT	–	Field energy production total
FGCV	–	Field gas calorific value
FGE	–	Gas recovery factor
FGIP	–	Gas in place
FGIPG	–	Gas in place (in gas phase)
FGIPGM	–	Gas in place (in gas phase; mass units)
FGIPMG	–	Gas in place (in gas phase); Accounts only for the mobile fraction of gas

Additional mnemonics for COMPS module		
FGIPMGM	–	Gas in place (in gas phase); Accounts only for the mobile fraction of gas (mass units)
FGIPRG	–	Gas in place (in gas phase); Accounts only for the trapped gas, i.e. the residual fraction of gas
FGIPRGM	–	Gas in place (in gas phase); Accounts only for the trapped gas, i.e. the residual fraction of gas (mass units)
FGIPL	–	Gas in place (in liquid phase)
FGIPLM	–	Gas in place (in liquid phase; mass units)
FGIPM	–	Gas in place (mass units)
FGIR	–	Gas injection rate
FGIRM	–	Gas injection rate (mass units)
FGIT	–	Gas injection total
FGITM	–	Gas injection total (mass units)
FGOR	–	Gas oil ratio
FGPR	–	Gas production rate
FGPRM	–	Gas production rate (mass units)
FGPT	–	Gas production total
FGPTM	–	Gas production total (mass units)
FOE	–	Oil recovery factor
FOILAPI	–	Field stock tank oil API gravity
FOILAPIG	–	Field stock tank oil API gravity (production summed then separated)
FOIP	–	Oil in place
FOIPG	–	Oil in place (in gas phase)
FOIPGM	–	Oil in place (in gas phase; mass units)
FOIPL	–	Oil in place (in liquid phase)
FOIPLM	–	Oil in place (in liquid phase; mass units)
FOIPM	–	Oil in place (mass units)
FOPR	–	Oil production rate
FOPRM	–	Oil production rate (mass units)
FOPT	–	Oil production total
FOPTM	–	Oil production total (mass units)
FVISGAS	–	Field gas viscosity
FVISGASG	–	Field gas viscosity (production summed then separated)

Additional mnemonics for COMPS module		
FVISOIL	–	Field viscosity of the stock tank oil
FVISOILG	–	Field viscosity of the stock tank oil (production summed then separated)
FWCT	–	Field water cut
FWIP	–	Water in place
FWIPM	–	Water in place (mass units)
FWIR	–	Water injection rate
FWIRM	–	Water injection rate (mass units)
FWIT	–	Water injection total
FWITM	–	Water injection total (mass units)
FWPR	–	Water production rate
FWPRM	–	Water production rate (mass units)
FWPT	–	Water production total
FWPTM	–	Water production total (mass units)
FAIPM$c	–	Component c=1,2,3,... in place (in the aqueous phase; mass units)
FXIPM$c	–	Component c=1,2,3,... in place (in oil phase; mass units)
FXMF$c	–	Field molar concentration of the component c=1,2,3,... in the stock tank oil
FXMFG$c	–	Field molar concentration of the component c=1,2,3,... in the stock tank oil (production summed then separated)
FYIPM$c	–	Component c=1,2,3,... in place (in gas phase; mass units)
FYMF$c	–	Field molar concentration of the component c=1,2,3,... in the vapor
FYMFG$c	–	Field molar concentration of the component c=1,2,3,... in the vapor (production summed then separated)
FYMM$c	–	Component c=1,2,3,... in place (in gas phase; mass units); Accounts only for the mobile fraction of gas
FYRM$c	–	Component c=1,2,3,... in place (in gas phase; mass units); Accounts only for the trapped gas, i.e. the residual fraction of gas
FZIPM$c	–	Component c=1,2,3,... in place (mass units)
FZMF$c	–	Field total molar concentration of the component c=1,2,3,... in the produced fluid

Additional mnemonics for COMPS module		
GGIR	–	Group gas injection rate
GGIRM	–	Group gas injection rate (mass units)
GGPI	–	Group gas potential injection
GGPIM	–	Group gas potential injection (mass units)
IKRG	–	Scaled end-point gas relative permeability (imbibition curve)
IKRO	–	Scaled end-point oil relative permeability (imbibition curve)
IKROW	–	Scaled end-point oil relative permeability (imbibition curve)
IKRW	–	Scaled end-point water relative permeability (imbibition curve)
IPCG	–	Scaled maximum gas-oil capillary pressure (imbibition curve)
IPCW	–	Scaled maximum water-oil capillary pressure (imbibition curve)
ISGCR	–	Scaled critical gas saturation (imbibition curve)
ISGL	–	Scaled connate gas saturation (imbibition curve)
ISGLPC	–	Scaled connate gas saturation for capillary pressure (imbibition curve)
ISGU	–	Scaled saturation table maximum gas saturation (imbibition curve)
ISOGCR	–	Scaled oil-in-gas saturation (imbibition curve)
ISOWCR	–	Scaled critical oil-in-water saturation (imbibition curve)
ISOWL	–	Scaled saturation table minimum oil-in-water saturation (imbibition curve)
ISOWU	–	Scaled saturation table maximum oil-in-water saturation (imbibition curve)
ISWCR	–	Scaled critical water saturation (imbibition curve)
ISWL	–	Scaled connate water saturation (imbibition curve)
ISWLPC	–	Scaled connate water saturation for capillary pressure (imbibition curve)
ISWU	–	Scaled saturation table maximum water saturation (imbibition curve)
ISWUPC	–	Scaled saturation table maximum water saturation for capillary pressure (imbibition curve)

Additional mnemonics for COMPS module		
KRG	–	Scaled end-point gas relative permeability
KRO	–	Scaled end-point oil relative permeability
KROW	–	Scaled end-point oil relative permeability
KRW	–	Scaled end-point water relative permeability
KVAL$c	–	K-value of the component c=1,2,3,...
KVEXP$c	–	K-value of the component c=1,2,3,... in the explicit level of the scheme (at the previous time step)
MWGAS	–	Molar density of the gas phase
MWOIL	–	Molar density of the oil phase
OILAPI	–	Oil API gravity
OMF	–	Molar fraction of oil phase
PCG	–	Scaled maximum gas capillary pressure
PCOG	–	oil-gas capillary pressure (i.e. Pgas-Poil)
PCW	–	Scaled maximum water capillary pressure
PCWO	–	Water-oil capillary pressure (i.e. Poil-Pwat)
PGAS	–	Pressure in gas phase
POIL	–	Pressure in oil phase
PWAT	–	Pressure in water phase
RELGAS	–	Gas phase relative permeability
RELOIL	–	Oil phase relative permeability
RELWAT	–	Water phase relative permeability
RPMGAS	–	Gas phase relative permeability (internal variable)
RPMOIL	–	Oil phase relative permeability (internal variable)
RPMWAT	–	Water phase relative permeability (internal variable)
RS	–	Gas in oil ratio
RSW	–	Gas in water ratio
RV	–	Oil in gas ratio
SDGAS	–	Gas density at surface conditions (all phases separated simulataneously)
SDOIL	–	Oil density at surface conditions (all phases separated simulataneously)
SGAS	–	Gas saturation
SGASAVG	–	Averaged over the fractures and matrix gas saturation in dual-porosity simulations

Additional mnemonics for COMPS module		
SGASSCL	–	Internal use
SGCR	–	Scaled critical gas saturation
SGL	–	Scaled connate gas saturation
SGLPC	–	Scaled connate gas saturation for capillary pressure
SGU	–	Scaled saturation table maximum gas saturation
SOGCR	–	Scaled oil-in-gas saturation
SOIL	–	Oil saturation
SOILAVG	–	Averaged over the fractures and matrix oil saturation in dual-porosity simulations
SOILINIT	–	Initial oil saturation
SOWCR	–	Scaled critical oil-in-water saturation
SOWL	–	Scaled saturation table minimum oil-in-water saturation
SOWU	–	Scaled saturation table maximum oil-in-water saturation
SWAT	–	Water saturation
SWATAVG	–	Averaged over the fractires and matrix water saturation in dual-porosity simulations
SWATINIT	–	Initial water saturation
SWCR	–	Scaled critical water saturation
SWL	–	Scaled connate water saturation
SWLPC	–	Scaled connate water saturation for capillary pressure
SWU	–	Scaled saturation table maximum water saturation
SWUPC	–	Scaled saturation table maximum water saturation for capillary pressure
TRELAX	–	Characteristic relaxation time to flash in non-equilibrium model
VISGAS	–	Gas viscosity
VISOIL	–	Oil viscosity
VISWAT	–	Water viscosity
VMF	–	Molar fraction of vapor phase
VOLGAS	–	Molar volume of the gas phase
VOLOIL	–	Molar volume of the oil phase
WBHDGAS	–	Well bottom-hole density of gas
WDGAS	–	Gas density at the output of separator
WDOIL	–	Oil density at the output of separator

Additional mnemonics for COMPS module		
WEPR	–	Well energy production rate
WGCV	–	Well gas calorific value
WGIR	–	Well gas injection rate
WGIRM	–	Well gas injection rate (mass units)
WGOR	–	Well gas-oil ratio
WGPI	–	Well gas potential injection
WGPIM	–	Well gas potential injection (mass units)
WGPP	–	Well gas potential production
WGPR	–	Well gas production rate
WGPRM	–	Well gas production rate (mass units)
WLPR	–	Well liquid production rate
WOILAPI	–	Oil API gravity at the output of the separator
WOGR	–	Well oil-gas ratio
WOPP	–	Well oil potential production
WOPR	–	Well oil production rate
WOPRM	–	Well oil production rate (mass units)
WSGAS	–	Volume fraction of the separator gas (excludes volume of condensate)
WSOIL	–	Volume fraction of oil in stock tank
WVISGAS	–	Gas viscosity at the output of separator
WVISOIL	–	Oil viscosity at the output of separator
WWCT	–	Well water cut
WWGR	–	Well water-gas ratio
WWIR	–	Well water injection rate
WWIRM	–	Well water injection rate (mass units)
WWIT	–	Well water injection total
WWITM	–	Well water injection total (mass units)
WWOR	–	Well water-oil ratio
WWPR	–	Well water producion rate
WWPRM	–	Well water producion rate (mass units)
WWPT	–	Well water production total
WWPTM	–	Well water production total (mass units)
WXMF$c	–	Molar concentraion of the component c=1,2,3,... in the stock tank oil

Additional mnemonics for COMPS module		
WYMF$c	–	Molar concentraion of the component c=1,2,3,... in the stock tank vapor
XIPM$c	–	Mass of component c=1,2,3,... (in oil phase)
XMF$c	–	Molar concentraion of the component c=1,2,3,... in liquid phase
XMF0$c	–	Molar concentraion of the component c=1,2,3,... in liquid phase (on explicit level)
XMFM$c	–	Mass concentraion of the component c=1,2,3,... in liquid phase
YIPM$c	–	Mass of component c=1,2,3,... (in gas phase)
YMF$c	–	Molar concentraion of the component c=1,2,3,... in gas phase
YMF0$c	–	Molar concentraion of the component c=1,2,3,... in gas phase (on explicit level)
YMFM$c	–	Mass concentraion of the component c=1,2,3,... in gas phase
YMM$c	–	Mass of component c=1,2,3,... (in gas phase); Accounts only for the mobile fraction of gas
YRM$c	–	Mass of component c=1,2,3,... (in gas phase); Accounts only for the trapped gas, i.e. the residual fraction of gas
ZIPM$c	–	Mass of component c=1,2,3,...
ZMF$c	–	Total molar composition of the component c=1,2,3,...

Table of available options for operations on COMPS arrays

		Input					Output				
Mnemonic	Array	GRID	EDIT	PROPS	INIT	SCHEDULE	GRID	EDIT	PROPS	INIT	SCHEDULE
AMF$c	1	–	–	–	–	+	–	–	–	–	–
AMFM$c	1	–	–	–	–	+	–	–	–	–	–
BGAS	1	–	–	–	+	+	–	–	–	–	–
BOIL	1	–	–	–	+	+	–	–	–	–	–
BWAT	1	–	–	–	+	+	–	–	–	–	–
DGAS	1	–	–	–	+	+	–	–	–	–	–
DOIL	1	–	–	–	+	+	–	–	–	–	–
DWAT	1	–	–	–	+	+	–	–	–	–	–
IKRG	1	–	–	+	+	+	–	–	+	–	–
IKRO	1	–	–	+	+	+	–	–	+	–	–
IKROW	1	–	–	+	+	+	–	–	–	–	–
IKRW	1	–	–	+	+	+	–	–	+	–	–
IPCG	1	–	–	+	+	+	–	–	+	–	–
IPCW	1	–	–	+	+	+	–	–	+	–	–
ISGCR	1	–	–	+	+	+	–	–	+	–	–
ISGL	1	–	–	+	+	+	–	–	+	–	–
ISGLPC	1	–	–	+	+	+	–	–	+	–	–
ISGU	1	–	–	+	+	+	–	–	+	–	–
ISOGCR	1	–	–	+	+	+	–	–	+	–	–
ISOWCR	1	–	–	+	+	+	–	–	+	–	–
ISOWL	1	–	–	+	+	+	–	–	–	–	–
ISOWU	1	–	–	+	+	+	–	–	–	–	–
ISWCR	1	–	–	+	+	+	–	–	+	–	–
ISWL	1	–	–	+	+	+	–	–	+	–	–
ISWLPC	1	–	–	+	+	+	–	–	+	–	–
ISWU	1	–	–	+	+	+	–	–	+	–	–
ISWUPC	1	–	–	+	+	+	–	–	–	–	–
KRG	1	–	–	+	+	+	–	–	+	–	–

Table of available options for operations on COMPS arrays

Mnemonic	Array	Input					Output				
		GRID	EDIT	PROPS	INIT	SCHEDULE	GRID	EDIT	PROPS	INIT	SCHEDULE
KRO	1	−	−	+	+	+	−	−	+	−	−
KROW	1	−	−	+	+	+	−	−	−	−	−
KRW	1	−	−	+	+	+	−	−	+	−	−
KVAL$c	1	−	−	−	+	+	−	−	−	−	−
KVEXP$c	1	−	−	−	+	+	−	−	−	−	−
MWGAS	1	−	−	−	+	+	−	−	−	−	−
MWOIL	1	−	−	−	+	+	−	−	−	−	−
OILAPI	1	−	−	−	+	+	−	−	−	−	−
PCG	1	−	−	+	+	+	−	−	+	−	−
PCW	1	−	−	+	+	+	−	−	+	−	−
PGAS	1	−	−	−	+	+	−	−	−	−	−
POIL	1	−	−	−	+	+	−	−	−	−	−
PWAT	1	−	−	−	+	+	−	−	−	−	−
RELGAS	1	−	−	−	+	+	−	−	−	+	+
RELOIL	1	−	−	−	+	+	−	−	−	+	+
RELWAT	1	−	−	−	+	+	−	−	−	+	+
RPMGAS	1	−	−	−	−	+	−	−	−	−	−
RPMOIL	1	−	−	−	−	+	−	−	−	−	−
RPMWAT	1	−	−	−	−	+	−	−	−	−	−
RSW	1	−	−	−	+	+	−	−	−	+	+
SGAS	1	−	−	−	+	+	−	−	−	−	−
SGASAVG	1	−	−	−	−	+	−	−	−	−	−
SGCR	1	−	−	+	+	+	−	−	+	−	−
SGL	1	−	−	+	+	+	−	−	+	−	−
SGLPC	1	−	−	+	+	+	−	−	+	−	−
SGU	1	−	−	+	+	+	−	−	+	−	−
SOGCR	1	−	−	+	+	+	−	−	+	−	−
SOIL	1	−	−	−	+	+	−	−	−	−	−

Table of available options for operations on COMPS arrays

		Input					Output				
Mnemonic	Array	GRID	EDIT	PROPS	INIT	SCHEDULE	GRID	EDIT	PROPS	INIT	SCHEDULE
SOILAVG	1	−	−	−	−	+	−	−	−	−	−
SOILINIT	1	−	−	−	+	−	−	−	−	+	−
SOWCR	1	−	−	+	+	+	−	−	+	−	−
SOWL	1	−	−	−	+	+	−	−	−	−	−
SOWU	1	−	−	−	+	+	−	−	−	−	−
SWAT	1	−	−	−	+	+	−	−	−	+	+
SWATAVG	1	−	−	−	−	+	−	−	−	−	−
SWATINIT	1	−	−	+	−	−	−	−	+	−	−
SWCR	1	−	−	+	−	−	−	−	+	−	−
SWL	1	−	−	+	−	−	−	−	+	−	−
SWLPC	1	−	−	+	−	−	−	−	+	−	−
SWU	1	−	−	+	−	−	−	−	+	−	−
TRELAX	1	−	−	+	+	+	−	−	+	+	+
VISGAS	1	−	−	−	+	+	−	−	−	−	−
VISOIL	1	−	−	−	+	+	−	−	−	−	−
VISWAT	1	−	−	−	+	+	−	−	−	+	+
VOLGAS	1	−	−	−	+	+	−	−	−	−	−
VOLOIL	1	−	−	−	+	+	−	−	−	−	−
XIPM$c	1	−	−	−	−	+	−	−	−	−	−
XMF$c	1	−	−	−	−	+	−	−	−	−	−
XMF0$c	1	−	−	−	−	+	−	−	−	−	−
XMFM$c	1	−	−	−	−	+	−	−	−	−	−
YIPM$c	1	−	−	−	−	+	−	−	−	−	−
YMF$c	1	−	−	−	−	+	−	−	−	−	−
YMF0$c	1	−	−	−	−	+	−	−	−	−	−
YMFM$c	1	−	−	−	−	+	−	−	−	−	−
YMM$c	1	−	−	−	−	+	−	−	−	−	−
YRM$c	1	−	−	−	−	+	−	−	−	−	−

Table of available options for operations on COMPS arrays

Mnemonic	Array	Input					Output				
		GRID	EDIT	PROPS	INIT	SCHEDULE	GRID	EDIT	PROPS	INIT	SCHEDULE
ZIPM$c	1	−	−	−	−	+	−	−	−	−	−
ZMF$c	1	−	−	−	+	+	−	−	−	+	+

Table of available options for output of COMPS arrays

	RPTGRID							RPTSUM						
Mnemonic	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
AMF$c	–	–	–	–	–	–	–	+	–	–	–	–	+	–
AMFM$c	–	–	–	–	–	–	–	+	–	–	–	–	+	–
BGAS	–	–	–	–	–	–	–	+	–	–	–	–	–	–
BGIP	–	–	–	–	–	–	–	+	–	–	–	–	–	–
BOIL	–	–	–	–	–	–	–	+	–	–	–	–	–	–
BOIP	–	–	–	–	–	–	–	+	–	–	–	–	–	–
BWAT	–	–	–	–	–	–	–	+	–	–	–	–	–	–
DGAS	–	–	–	–	–	–	–	+	–	–	–	–	–	–
DOIL	–	–	–	–	–	–	–	+	–	–	–	–	–	–
DWAT	–	–	–	–	–	–	–	+	–	–	–	–	–	–
EOSERR	–	–	–	–	–	–	–	+	–	–	–	–	–	–
EOSITER	–	–	–	–	–	–	–	+	–	–	–	–	–	–
FDGAS	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FDGASG	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FDOIL	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FDOILG	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FEE	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FEIP	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FEPR	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FEPT	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGCV	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGE	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIP	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIPG	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIPGM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIPMG	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIPMGM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIPRG	–	–	–	–	–	–	–	–	–	–	–	+	–	–

Table of available options for output of COMPS arrays

Mnemonic	RPTGRID							RPTSUM						
	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
FGIPRGM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIPL	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIPLM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIPM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIR	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIRM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIT	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGITM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGOR	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGPR	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGPRM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGPT	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGPTM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FOE	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FOILAPI	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FOILAPIG	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FOIP	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FOIPG	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FOIPGM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FOIPL	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FOIPLM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FOIPM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FOPR	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FOPRM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FOPT	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FOPTM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FVISGAS	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FVISGASG	–	–	–	–	–	–	–	–	–	–	–	+	–	–

Table of available options for output of COMPS arrays

	RPTGRID							RPTSUM						
Mnemonic	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
FVISOIL	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FVISOILG	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FWCT	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FWIP	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FWIPM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FWIR	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FWIRM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FWIT	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FWITM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FWPR	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FWPRM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FWPT	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FWPTM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FAIPM$c	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FXIPM$c	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FXMF$c	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FXMFG$c	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FYIPM$c	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FYMF$c	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FYMFG$c	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FYMM$c	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FYRM$c	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FZIPM$c	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FZMF$c	–	–	–	–	–	–	–	–	–	–	–	+	–	–
GGIR	–	–	–	–	–	–	–	–	–	–	–	–	–	–
GGIRM	–	–	–	–	–	–	–	–	–	–	–	–	–	–
GGPI	–	–	–	–	–	–	–	–	–	–	–	–	–	–
GGPIM	–	–	–	–	–	–	–	–	–	–	–	–	–	–

Table of available options for output of COMPS arrays

Mnemonic	RPTGRID							RPTSUM						
	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
IKRG	–	–	–	–	–	–	–	+	–	–	–	–	–	–
IKRO	–	–	–	–	–	–	–	+	–	–	–	–	–	–
IKROW	–	–	–	–	–	–	–	+	–	–	–	–	–	–
IKRW	–	–	–	–	–	–	–	+	–	–	–	–	–	–
IPCG	–	–	–	–	–	–	–	+	–	–	–	–	–	–
IPCW	–	–	–	–	–	–	–	+	–	–	–	–	–	–
ISGCR	–	–	–	–	–	–	–	+	–	–	–	–	–	–
ISGL	–	–	–	–	–	–	–	+	–	–	–	–	–	–
ISGLPC	–	–	–	–	–	–	–	+	–	–	–	–	–	–
ISGU	–	–	–	–	–	–	–	+	–	–	–	–	–	–
ISOGCR	–	–	–	–	–	–	–	+	–	–	–	–	–	–
ISOWCR	–	–	–	–	–	–	–	+	–	–	–	–	–	–
ISOWL	–	–	–	–	–	–	–	+	–	–	–	–	–	–
ISOWU	–	–	–	–	–	–	–	+	–	–	–	–	–	–
ISWCR	–	–	–	–	–	–	–	+	–	–	–	–	–	–
ISWL	–	–	–	–	–	–	–	+	–	–	–	–	–	–
ISWLPC	–	–	–	–	–	–	–	+	–	–	–	–	–	–
ISWU	–	–	–	–	–	–	–	+	–	–	–	–	–	–
ISWUPC	–	–	–	–	–	–	–	+	–	–	–	–	–	–
KRG	–	–	–	–	–	–	–	+	–	–	–	–	–	–
KRO	–	–	–	–	–	–	–	+	–	–	–	–	–	–
KROW	–	–	–	–	–	–	–	+	–	–	–	–	–	–
KRW	–	–	–	–	–	–	–	+	–	–	–	–	–	–
KVAL$c	–	–	–	–	–	–	–	+	–	–	–	–	–	–
KVEXP$c	–	–	–	–	–	–	–	+	–	–	–	–	–	–
MWGAS	–	–	–	–	–	–	–	+	–	–	–	–	–	–
MWOIL	–	–	–	–	–	–	–	+	–	–	–	–	–	–
OILAPI	–	–	–	–	–	–	–	+	–	–	–	–	–	–

Table of available options for output of COMPS arrays

	RPTGRID							RPTSUM						
Mnemonic	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
OMF	–	–	–	–	–	–	–	+	–	–	–	–	+	–
PCG	–	–	–	–	–	–	–	+	–	–	–	–	–	–
PCOG	–	–	–	–	–	–	–	+	–	–	–	–	–	–
PCW	–	–	–	–	–	–	–	+	–	–	–	–	–	–
PCWO	–	–	–	–	–	–	–	+	–	–	–	–	–	–
PGAS	–	–	–	–	–	–	–	+	–	–	–	–	–	–
POIL	–	–	–	–	–	–	–	+	–	–	–	–	–	–
PWAT	–	–	–	–	–	–	–	+	–	–	–	–	–	–
RELGAS	–	–	–	–	–	–	–	+	–	–	–	–	–	–
RELOIL	–	–	–	–	–	–	–	+	–	–	–	–	–	–
RELWAT	–	–	–	–	–	–	–	+	–	–	–	–	–	–
RPMGAS	–	–	–	–	–	–	–	+	–	–	–	–	–	–
RPMOIL	–	–	–	–	–	–	–	+	–	–	–	–	–	–
RPMWAT	–	–	–	–	–	–	–	+	–	–	–	–	–	–
RS	–	–	–	–	–	–	–	+	–	–	–	–	–	–
RSW	–	–	–	–	–	–	–	+	–	–	–	–	–	–
RV	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SDGAS	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SDOIL	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SGAS	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SGASAVG	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SGASSCL	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SGCR	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SGL	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SGLPC	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SGU	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SOGCR	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SOIL	–	–	–	–	–	–	–	+	–	–	–	–	–	–

Table of available options for output of COMPS arrays

Mnemonic	RPTGRID							RPTSUM						
	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
SOILAVG	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SOILINIT	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SOWCR	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SOWL	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SOWU	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SWAT	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SWATAVG	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SWATINIT	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SWCR	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SWL	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SWLPC	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SWU	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SWUPC	–	–	–	–	–	–	–	+	–	–	–	–	–	–
TRELAX	–	–	–	–	–	–	–	+	–	–	–	–	–	–
VISGAS	–	–	–	–	–	–	–	+	–	–	–	–	–	–
VISOIL	–	–	–	–	–	–	–	+	–	–	–	–	–	–
VISWAT	–	–	–	–	–	–	–	+	–	–	–	–	–	–
VMF	–	–	–	–	–	–	–	+	–	–	–	–	+	–
VOLGAS	–	–	–	–	–	–	–	+	–	–	–	–	–	–
VOLOIL	–	–	–	–	–	–	–	+	–	–	–	–	–	–
WBHDGAS	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WDGAS	–	–	–	–	–	–	–	–	–	+	+	–	–	–
WDOIL	–	–	–	–	–	–	–	–	–	+	+	–	–	–
WEPR	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WGCV	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WGIR	–	–	–	–	–	–	–	–	–	+	+	–	–	–
WGIRM	–	–	–	–	–	–	–	–	–	+	+	–	–	–
WGOR	–	–	–	–	–	–	–	–	–	+	+	–	–	–

Table of available options for output of COMPS arrays

	RPTGRID							RPTSUM						
Mnemonic	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
WGPI	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WGPIM	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WGPP	–	–	–	–	–	–	–	–	–	+	+	–	–	–
WGPR	–	–	–	–	–	–	–	–	–	+	+	–	–	–
WGPRM	–	–	–	–	–	–	–	–	–	+	+	–	–	–
WLPR	–	–	–	–	–	–	–	–	–	+	+	–	–	–
WOILAPI	–	–	–	–	–	–	–	–	–	+	+	–	–	–
WOGR	–	–	–	–	–	–	–	–	–	+	+	–	–	–
WOPP	–	–	–	–	–	–	–	–	–	+	+	–	–	–
WOPR	–	–	–	–	–	–	–	–	–	+	+	–	–	–
WOPRM	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WSGAS	–	–	–	–	–	–	–	–	–	+	+	–	–	–
WSOIL	–	–	–	–	–	–	–	–	–	+	+	–	–	–
WVISGAS	–	–	–	–	–	–	–	–	–	+	+	–	–	–
WVISOIL	–	–	–	–	–	–	–	–	–	+	+	–	–	–
WWCT	–	–	–	–	–	–	–	–	–	+	+	–	–	–
WWGR	–	–	–	–	–	–	–	–	–	+	+	–	–	–
WWIR	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WWIRM	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WWIT	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WWITM	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WWOR	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WWPR	–	–	–	–	–	–	–	–	–	+	+	–	–	–
WWPRM	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WWPT	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WWPTM	–	–	–	–	–	–	–	–	–	–	+	–	–	–
WXMF$c	–	–	–	–	–	–	–	–	–	+	+	–	–	–
WYMF$c	–	–	–	–	–	–	–	–	–	+	+	–	–	–

Table of available options for output of COMPS arrays														
	RPTGRID							RPTSUM						
Mnemonic	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
XIPM$c	−	−	−	−	−	−	−	+	−	−	−	−	+	−
XMF$c	−	−	−	−	−	−	−	+	−	−	−	−	+	−
XMF0$c	−	−	−	−	−	−	−	+	−	−	−	−	+	−
XMFM$c	−	−	−	−	−	−	−	+	−	−	−	−	+	−
YIPM$c	−	−	−	−	−	−	−	+	−	−	−	−	+	−
YMF$c	−	−	−	−	−	−	−	+	−	−	−	−	+	−
YMF0$c	−	−	−	−	−	−	−	+	−	−	−	−	+	−
YMFM$c	−	−	−	−	−	−	−	+	−	−	−	−	+	−
YMM$c	−	−	−	−	−	−	−	+	−	−	−	−	+	−
YRM$c	−	−	−	−	−	−	−	+	−	−	−	−	+	−
ZIPM$c	−	−	−	−	−	−	−	+	−	−	−	−	+	−
ZMF$c	−	−	−	−	−	−	−	+	−	−	−	−	+	−

2.6. GASSTORE mnemonics

Additional mnemonics for GASSTORE module		
DGAS	–	Gas density
DLIQ	–	Liquid density
DSOL	–	Solid phase density
ENTHMGAS	–	Specific enthalpy of gas
ENTHMLIQ	–	Specific enthalpy of liquid
ENTHMSOL	–	Specific enthalpy of solid phase
FGIP	–	Gas in place
FGIPG	–	Gas in place (in gas phase)
FGIPMG	–	Gas in place (in gas phase); Accounts only for the mobile fraction of gas
FGIPMGM	–	Gas in place (in gas phase); Accounts only for the mobile fraction of gas (mass units)
FGIPRG	–	Gas in place (in gas phase); Accounts only for the trapped gas, i.e. the residual fraction of gas
FGIPRGM	–	Gas in place (in gas phase); Accounts only for the trapped gas, i.e. the residual fraction of gas (mass units)
FGIPGM	–	Gas in place (in gas phase; mass units)
FGIPL	–	Gas in place (in liquid phase)
FGIPLM	–	Gas in place (in liquid phase; mass units)
FGIPM	–	Gas in place (mass units)
FGIR	–	Gas injection rate
FGIRM	–	Gas injection rate (mass units)
FGIT	–	Gas injection total
FGITM	–	Gas injection total (mass units)
FGPR	–	Gas production rate
FGPRM	–	Gas production rate (mass units)
FGPT	–	Gas production total
FGPTM	–	Gas production total (mass units)
FWGR	–	Water gas ratio
FWIP	–	Water in place
FWIPM	–	Water in place (mass units)
FWIR	–	Water injection rate

Additional mnemonics for GASSTORE module		
FWIRM	–	Water injection rate (mass units)
FWIT	–	Water injection total
FWITM	–	Water injection total (mass units)
FWPR	–	Water production rate
FWPRM	–	Water production rate (mass units)
FWPT	–	Water production total
FWPTM	–	Water production total (mass units)
GIPG	–	Gas-in-place in the gas phase (in the grid block)
GIPGM	–	Gas mass in the gas phase (in the grid block)
GIPL	–	Gas-in-place in the liquid phase (in the grid block)
GIPLM	–	Gas mass in the liquid phase (in the grid block)
GIP	–	Gas-in-place (in the grid block)
GIPM	–	Gas mass (in the grid block)
GIPMG	–	Mobile gas-in-place (in the grid block)
GIPMGM	–	Mass of the mobile gas in the gas phase (in the grid block)
GIPRG	–	Residual gas-in-place (in the grid block)
GIPRGM	–	Mass of the residual gas in the gas phase (in the grid block)
GGIR	–	Group gas injection rate
GGIRM	–	Group gas injection rate (mass units)
GGPI	–	Group gas potential injection
GGPIM	–	Group gas potential injection (mass units)
GWPR	–	Group water producion rate
GWPT	–	Group water production total
IKRG	–	Scaled end-point gas relative permeability (imbibition curve)
IKRL	–	Scaled end-point liquid relative permeability (imbibition curve)
IPCG	–	Scaled maximum gas capillary pressure (imbibition curve)
ISGCR	–	Scaled critical gas saturation (imbibition curve)
ISGL	–	Scaled connate gas saturation (imbibition curve)
ISGLPC	–	Scaled connate gas saturation for capillary pressure (imbibition curve)
ISGU	–	Scaled saturation table maximum gas saturation (imbibition curve)

Additional mnemonics for GASSTORE module		
ISLCR	–	Scaled critical liquid saturation (imbibition curve)
ISSCR	–	Scaled critical saturation of the solid phase (imbibition curve)
KRG	–	Scaled end-point gas relative permeability
KRFS	–	Permeability multiplier due to precipitation of salt
KRL	–	Scaled end-point liquid relative permeability
PCG	–	Scaled maximum gas capillary pressure
PCWG	–	Water-gas capillary pressure
PGAS	–	Pressure in gas phase
PLIQ	–	Pressure in liquid phase
RELGAS	–	Gas phase relative permeability
RELLIQ	–	Liquid phase relative permeability
RPMGAS	–	Gas phase relative permeability (internal variable)
RPMLIQ	–	Liquid phase relative permeability (internal variable)
SGAS	–	Gas saturation
SGCR	–	Scaled critical gas saturation
SGL	–	Scaled connate gas saturation
SGLPC	–	Scaled connate gas saturation for capillary pressure
SGU	–	Scaled saturation table maximum gas saturation
SLCR	–	Scaled critical liquid saturation
SLIQ	–	Liquid saturation
SMOL	–	Salt molality in liquid (brine) phase
SMOLT	–	Total salt molality of water-salt two-phase solution
SSCR	–	Scaled critical saturation of the solid phase
SSOL	–	Solid saturation
VISGAS	–	Gas viscosity
VISLIQ	–	Liquid viscosity
WGIR	–	Well gas injection rate
WGIRM	–	Well gas injection rate (mass units
WGIT	–	Well gas injection total
WGITM	–	Well gas injection total (mass units)
WGPI	–	Well gas potential injection
WGPIM	–	Well gas potential injection (mass units)
WGPR	–	Well gas production rate

Additional mnemonics for GASSTORE module		
WGPRM	–	Well gas production rate (mass units)
WGPT	–	Well gas production total
WGPTM	–	Well gas production total (mass units)
WWGR	–	Well water-gas ratio
WWIR	–	Well water injection rate
WWIRM	–	Well water injection rate (mass units)
WWIT	–	Well water injection total
WWITM	–	Well water injection total (mass units)
WWPR	–	Well water producion rate
WWPRM	–	Well water producion rate (mass units)
WWPT	–	Well water production total
WWPTM	–	Well water production total (mass units)
XG	–	Gas molar fraction in brine
XGM	–	Gas mass fraction in brine
XS	–	Salt molar fraction in brine
XSM	–	Salt mass fraction in brine (in two-component water-salt system)
XSMR	–	Salt mass fraction in brine (in three-component water-salt-gas system)
XSMT	–	Total salt mass fraction of water-salt two-phase solution
XW	–	Water molar fraction in brine
XWM	–	Water mass fraction in brine
YG	–	Gas molar fraction in gas phase
YGM	–	Gas mass fraction in gas phase
YW	–	Water molar fraction in gas phase
YWM	–	Water mass fraction in gas phase

Table of available options for operations on GASSTORE arrays

		Input					Output				
Mnemonic	Array	GRID	EDIT	PROPS	INIT	SCHEDULE	GRID	EDIT	PROPS	INIT	SCHEDULE
DGAS	1	–	–	–	–	+	–	–	–	–	–
DLIQ	1	–	–	–	–	+	–	–	–	–	–
DSOL	1	–	–	–	–	+	–	–	–	–	–
ENTHMGAS	1	–	–	–	–	+	–	–	–	–	–
ENTHMLIQ	1	–	–	–	–	+	–	–	–	–	–
ENTHMSOL	1	–	–	–	–	+	–	–	–	–	–
GIPG	1	–	–	–	–	+	–	–	–	–	–
GIPGM	1	–	–	–	–	+	–	–	–	–	–
GIPL	1	–	–	–	–	+	–	–	–	–	–
GIPLM	1	–	–	–	–	+	–	–	–	–	–
GIP	1	–	–	–	–	+	–	–	–	–	–
GIPM	1	–	–	–	–	+	–	–	–	–	–
GIPMG	1	–	–	–	–	+	–	–	–	–	–
GIPMGM	1	–	–	–	–	+	–	–	–	–	–
GIPRG	1	–	–	–	–	+	–	–	–	–	–
GIPRGM	1	–	–	–	–	+	–	–	–	–	–
IKRG	1	–	–	+	+	+	–	–	+	–	–
IKRL	1	–	–	+	+	+	–	–	+	–	–
IPCG	1	–	–	+	+	+	–	–	+	–	–
ISGCR	1	–	–	+	+	+	–	–	+	–	–
ISGL	1	–	–	+	+	+	–	–	+	–	–
ISGLPC	1	–	–	+	+	+	–	–	+	–	–
ISGU	1	–	–	+	+	+	–	–	+	–	–
ISLCR	1	–	–	+	+	+	–	–	+	–	–
ISSCR	1	–	–	+	+	+	–	–	+	–	–
KRG	1	–	–	+	+	+	–	–	+	–	–
KRFS	1	–	–	–	–	+	–	–	–	–	–
KRL	1	–	–	+	+	+	–	–	+	–	–

Table of available options for operations on GASSTORE arrays												
		Input					Output					
Mnemonic	Array	GRID	EDIT	PROPS	INIT	SCHEDULE	GRID	EDIT	PROPS	INIT	SCHEDULE
PCG	1	−	−	+	+	+	−	−	+	−	−
PGAS	1	−	−	−	−	+	−	−	−	−	−
PLIQ	1	−	−	−	−	+	−	−	−	−	−
RELGAS	1	−	−	−	−	+	−	−	−	−	−
RELLIQ	1	−	−	−	−	+	−	−	−	−	−
RPMGAS	1	−	−	−	−	+	−	−	−	−	−
RPMLIQ	1	−	−	−	−	+	−	−	−	−	−
SGAS	1	−	−	−	+	+	−	−	−	+	+
SGCR	1	−	−	+	+	+	−	−	+	−	−
SGL	1	−	−	+	+	+	−	−	+	−	−
SGLPC	1	−	−	+	+	+	−	−	+	−	−
SGU	1	−	−	+	+	+	−	−	+	−	−
SLCR	1	−	−	+	+	+	−	−	+	−	−
SLIQ	1	−	−	−	+	+	−	−	−	+	+
SMOL	1	−	−	−	+	+	−	−	−	+	+
SSCR	1	−	−	+	+	+	−	−	+	−	−
SSOL	1	−	−	−	+	+	−	−	−	+	+
VISGAS	1	−	−	−	−	+	−	−	−	−	−
VISLIQ	1	−	−	−	−	+	−	−	−	−	−
XGM	1	−	−	−	+	+	−	−	−	+	+
XS	1	−	−	−	−	+	−	−	−	−	−
XSM	1	−	−	−	+	+	−	−	−	+	+
YWM	1	−	−	−	+	+	−	−	−	+	+

Table of available options for output of GASSTORE arrays

	RPTGRID							RPTSUM						
Mnemonic	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
DGAS	–	–	–	–	–	–	–	+	–	–	–	–	+	–
DLIQ	–	–	–	–	–	–	–	+	–	–	–	–	+	–
DSOL	–	–	–	–	–	–	–	+	–	–	–	–	+	–
ENTHMGAS	–	–	–	–	–	–	–	+	–	–	–	–	+	–
ENTHMLIQ	–	–	–	–	–	–	–	+	–	–	–	–	+	–
ENTHMSOL	–	–	–	–	–	–	–	+	–	–	–	–	+	–
FGIP	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIPG	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIPMG	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIPMGM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIPRG	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIPRGM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIPGM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIPL	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIPLM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIPM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIR	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIRM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGIT	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGITM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGPR	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGPRM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGPT	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FGPTM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FWGR	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FWIP	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FWIPM	–	–	–	–	–	–	–	–	–	–	–	+	–	–
FWIR	–	–	–	–	–	–	–	–	–	–	–	+	–	–

Table of available options for output of GASSTORE arrays

Mnemonic	RPTGRID							RPTSUM						
	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
FWIRM	−	−	−	−	−	−	−	−	−	−	−	+	−	−
FWIT	−	−	−	−	−	−	−	−	−	−	−	+	−	−
FWITM	−	−	−	−	−	−	−	−	−	−	−	+	−	−
FWPR	−	−	−	−	−	−	−	−	−	−	−	+	−	−
FWPRM	−	−	−	−	−	−	−	−	−	−	−	+	−	−
FWPT	−	−	−	−	−	−	−	−	−	−	−	+	−	−
FWPTM	−	−	−	−	−	−	−	−	−	−	−	+	−	−
GIPG	−	−	−	−	−	−	−	+	−	−	−	−	+	−
GIPGM	−	−	−	−	−	−	−	+	−	−	−	−	+	−
GIPL	−	−	−	−	−	−	−	+	−	−	−	−	+	−
GIPLM	−	−	−	−	−	−	−	+	−	−	−	−	+	−
GIP	−	−	−	−	−	−	−	+	−	−	−	−	+	−
GIPM	−	−	−	−	−	−	−	+	−	−	−	−	+	−
GIPMG	−	−	−	−	−	−	−	+	−	−	−	−	+	−
GIPMGM	−	−	−	−	−	−	−	+	−	−	−	−	+	−
GIPRG	−	−	−	−	−	−	−	+	−	−	−	−	+	−
GIPRGM	−	−	−	−	−	−	−	+	−	−	−	−	+	−
GGIR	−	−	−	−	−	−	−	−	−	−	−	−	−	−
GGIRM	−	−	−	−	−	−	−	−	−	−	−	−	−	−
GGPI	−	−	−	−	−	−	−	−	−	−	−	−	−	−
GGPIM	−	−	−	−	−	−	−	−	−	−	−	−	−	−
GWPR	−	−	−	−	−	−	−	−	−	−	−	−	−	−
GWPT	−	−	−	−	−	−	−	−	−	−	−	−	−	−
IKRG	−	−	−	−	−	−	−	+	−	−	−	−	−	−
IKRL	−	−	−	−	−	−	−	+	−	−	−	−	−	−
IPCG	−	−	−	−	−	−	−	+	−	−	−	−	−	−
ISGCR	−	−	−	−	−	−	−	+	−	−	−	−	−	−
ISGL	−	−	−	−	−	−	−	+	−	−	−	−	−	−

Table of available options for output of GASSTORE arrays

	RPTGRID							RPTSUM						
Mnemonic	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
ISGLPC	–	–	–	–	–	–	–	+	–	–	–	–	–	–
ISGU	–	–	–	–	–	–	–	+	–	–	–	–	–	–
ISLCR	–	–	–	–	–	–	–	+	–	–	–	–	–	–
ISSCR	–	–	–	–	–	–	–	+	–	–	–	–	–	–
KRG	–	–	–	–	–	–	–	+	–	–	–	–	–	–
KRFS	–	–	–	–	–	–	–	+	–	–	–	–	+	–
KRL	–	–	–	–	–	–	–	+	–	–	–	–	–	–
PCG	–	–	–	–	–	–	–	+	–	–	–	–	–	–
PCWG	–	–	–	–	–	–	–	+	–	–	–	–	–	–
PGAS	–	–	–	–	–	–	–	+	–	–	–	–	+	–
PLIQ	–	–	–	–	–	–	–	+	–	–	–	–	+	–
RELGAS	–	–	–	–	–	–	–	+	–	–	–	–	–	–
RELLIQ	–	–	–	–	–	–	–	+	–	–	–	–	–	–
RPMGAS	–	–	–	–	–	–	–	+	–	–	–	–	–	–
RPMLIQ	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SGAS	–	–	–	–	–	–	–	+	–	–	–	–	+	–
SGCR	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SGL	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SGLPC	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SGU	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SLCR	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SLIQ	–	–	–	–	–	–	–	+	–	–	–	–	+	–
SMOL	–	–	–	–	–	–	–	+	–	–	–	–	+	–
SMOLT	–	–	–	–	–	–	–	+	–	–	–	–	+	–
SSCR	–	–	–	–	–	–	–	+	–	–	–	–	–	–
SSOL	–	–	–	–	–	–	–	+	–	–	–	–	+	–
VISGAS	–	–	–	–	–	–	–	+	–	–	–	–	+	–
VISLIQ	–	–	–	–	–	–	–	+	–	–	–	–	+	–

Table of available options for output of GASSTORE arrays

Mnemonic	RPTGRID							RPTSUM						
	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
WGIR	−	−	−	−	−	−	−	−	−	−	+	−	−	−
WGIRM	−	−	−	−	−	−	−	−	−	−	+	−	−	−
WGIT	−	−	−	−	−	−	−	−	−	−	+	−	−	−
WGITM	−	−	−	−	−	−	−	−	−	−	+	−	−	−
WGPI	−	−	−	−	−	−	−	−	−	−	+	−	−	−
WGPIM	−	−	−	−	−	−	−	−	−	−	+	−	−	−
WGPR	−	−	−	−	−	−	−	−	−	−	+	−	−	−
WGPRM	−	−	−	−	−	−	−	−	−	−	+	−	−	−
WGPT	−	−	−	−	−	−	−	−	−	−	+	−	−	−
WGPTM	−	−	−	−	−	−	−	−	−	−	+	−	−	−
WWGR	−	−	−	−	−	−	−	−	−	−	+	−	−	−
WWIR	−	−	−	−	−	−	−	−	−	−	+	−	−	−
WWIRM	−	−	−	−	−	−	−	−	−	−	+	−	−	−
WWIT	−	−	−	−	−	−	−	−	−	−	+	−	−	−
WWITM	−	−	−	−	−	−	−	−	−	−	+	−	−	−
WWPR	−	−	−	−	−	−	−	−	−	−	+	−	−	−
WWPRM	−	−	−	−	−	−	−	−	−	−	+	−	−	−
WWPT	−	−	−	−	−	−	−	−	−	−	+	−	−	−
WWPTM	−	−	−	−	−	−	−	−	−	−	+	−	−	−
XG	−	−	−	−	−	−	−	+	−	−	−	−	+	−
XGM	−	−	−	−	−	−	−	+	−	−	−	−	+	−
XS	−	−	−	−	−	−	−	+	−	−	−	−	+	−
XSM	−	−	−	−	−	−	−	+	−	−	−	−	+	−
XSMR	−	−	−	−	−	−	−	+	−	−	−	−	+	−
XSMT	−	−	−	−	−	−	−	+	−	−	−	−	+	−
XW	−	−	−	−	−	−	−	+	−	−	−	−	+	−
XWM	−	−	−	−	−	−	−	+	−	−	−	−	+	−
YG	−	−	−	−	−	−	−	+	−	−	−	−	+	−

Table of available options for output of GASSTORE arrays														
	RPTGRID							RPTSUM						
Mnemonic	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
YGM	−	−	−	−	−	−	−	+	−	−	−	−	+	−
YW	−	−	−	−	−	−	−	+	−	−	−	−	+	−
YWM	−	−	−	−	−	−	−	+	−	−	−	−	+	−

2.7. SIMPLMOD mnemonics

There are no additional mnemonics for SIMPLMOD module.

2.8. T2EOS1 mnemonics

Additional mnemonics for T2EOS1 module		
DVAP	–	Vapor density
DWAT	–	Water density
ENTHMTOT	–	Total specific enthalpy of the two-phase fluid
ENTHMVAP	–	Specific enthalpy of vapor
ENTHMWAT	–	Specific enthalpy of water
HCVAP	–	Vapor heat conduction coefficient
HCWAT	–	Water heat conduction coefficient
PCWV	–	Water-vapor capillary pressure
PVAP	–	Pressure in vapor phase
PWAT	–	Pressure in water phase
RELVAP	–	Relative permeability of vapor
RELWAT	–	Relative permeability of water
SVAP	–	Vapor saturation
SWAT	–	Water saturation
TEQUIL	–	Equilibrium temperature (degrees Kelvin)
TEQUILC	–	Equilibrium temperature (degrees Celsius)
VISVAP	–	Vapor viscosity
VISWAT	–	Water viscosity
WENTHM	–	Total enthalpy of the fluid flowing through borehole

Table of available options for operations on T2EOS1 arrays

Mnemonic	Array	Input					Output				
		GRID	EDIT	PROPS	INIT	SCHEDULE	GRID	EDIT	PROPS	INIT	SCHEDULE
ENTHMTOT	1	−	−	−	+	+	−	−	−	+	+
PVAP	1	−	−	−	−	+	−	−	−	−	−
PWAT	1	−	−	−	−	+	−	−	−	−	−
SVAP	1	−	−	−	+	+	−	−	−	+	+
SWAT	1	−	−	−	+	+	−	−	−	+	+
TEQUIL	1	−	−	−	+	+	−	−	−	−	−
TEQUILC	1	−	−	−	+	+	−	−	−	−	−

Table of available options for output of T2EOS1 arrays														
	RPTGRID							RPTSUM						
Mnemonic	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
DVAP	−	−	−	−	−	−	−	+	−	−	−	−	+	−
DWAT	−	−	−	−	−	−	−	+	−	−	−	−	+	−
ENTHMTOT	−	−	−	−	−	−	−	+	−	−	−	−	+	−
ENTHMVAP	−	−	−	−	−	−	−	+	−	−	−	−	+	−
ENTHMWAT	−	−	−	−	−	−	−	+	−	−	−	−	+	−
HCVAP	−	−	−	−	−	−	−	+	−	−	−	−	+	−
HCWAT	−	−	−	−	−	−	−	+	−	−	−	−	+	−
PCWV	−	−	−	−	−	−	−	+	−	−	−	−	−	−
PVAP	−	−	−	−	−	−	−	+	−	−	−	−	+	−
PWAT	−	−	−	−	−	−	−	+	−	−	−	−	+	−
RELVAP	−	−	−	−	−	−	−	+	−	−	−	−	+	−
RELWAT	−	−	−	−	−	−	−	+	−	−	−	−	+	−
SVAP	−	−	−	−	−	−	−	+	−	−	−	−	+	−
SWAT	−	−	−	−	−	−	−	+	−	−	−	−	+	−
TEQUIL	−	−	−	−	−	−	−	+	−	−	−	−	+	−
TEQUILC	−	−	−	−	−	−	−	+	−	−	−	−	+	−
VISVAP	−	−	−	−	−	−	−	+	−	−	−	−	+	−
VISWAT	−	−	−	−	−	−	−	+	−	−	−	−	+	−
WENTHM	−	−	−	−	−	−	−	−	−	−	+	−	−	−

2.9. USEREOS mnemonics

Additional mnemonics for USEREOS module		
Cnamec-namep	–	Mass concentration of the component ‘namec’ in the phase ‘namep’
Dname	–	Density of the phase ‘name’
ENTHMname	–	Enthalpy of the phase ‘name’
Pname	–	Pressure in the phase ‘name’
RELname	–	Relative permeability of the phase ‘name’
Sname	–	Saturation of the phase ‘name’
VISname	–	Viscosity of the phase ‘name’
ZFname	–	Total mass fraction of the component ‘name’

Table of available options for operations on USEREOS arrays

Mnemonic	Array	Input					Output				
		GRID	EDIT	PROPS	INIT	SCHEDULE	GRID	EDIT	PROPS	INIT	SCHEDULE
Cnamec-namep	1	–	–	–	–	+	–	–	–	–	–
Dname	1	–	–	–	–	+	–	–	–	–	–
ENTHMname	1	–	–	–	–	+	–	–	–	–	–
RELname	1	–	–	–	–	+	–	–	–	–	–
Sname	1	–	–	–	–	+	–	–	–	–	–
VISname	1	–	–	–	–	+	–	–	–	–	–
ZFname	1	–	–	–	+	+	–	–	–	+	+

Table of available options for output of USEREOS arrays

	RPTGRID							RPTSUM						
Mnemonic	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA	CELLDATA	CONNDATA	SRCDATA	WELLDATA	TOTDATA	FPCEDATA	FPCODATA
Cnamec-namep	−	−	−	−	−	−	−	+	−	−	−	−	−	−
Dname	−	−	−	−	−	−	−	+	−	−	−	−	−	−
ENTHMname	−	−	−	−	−	−	−	+	−	−	−	−	−	−
Pname	−	−	−	−	−	−	−	+	−	−	−	−	+	−
RELname	−	−	−	−	−	−	−	+	−	−	−	−	−	−
Sname	−	−	−	−	−	−	−	+	−	−	−	−	−	−
VISname	−	−	−	−	−	−	−	+	−	−	−	−	−	−
ZFname	−	−	−	−	−	−	−	+	−	−	−	−	+	−

3 Options vs EOS modules

The simulator includes a set of modelling options activated with specific keywords in the RUNSPEC section. In general, every option is supposed to be available with every EOS module. However some parts of the code where not developed and not validated against benchmark problems yet. An extended number of available EOS modules and options makes it almost impossible to validate the simulator against benchmark problems with all possible combinations of options and EOS modules. Therefore, some EOS modules do not allow all options or should be used with precautions.

It should be noted that due to module architecture of the code, a particular option activate one and the same module with any EOS module. Actually, the modules are tested using the benchmarks . Therefore, if a module is accurate with a particular EOS module, it should also be accurate with another EOS module.

The tables below demonstrate which option can be used with a particular EOS module. The tables contain the following short tags

- DEVELOPED AND TESTED means that option can be activated with EOS module. This tag indicates that the simulator was validated against some benchmarks with this particular option providing enough confidence in that simulation results obtained with this particular EOS module and this particular option are correct.

- USE WITH PRECAUTIONS means that option can be activated with the EOS module but the simulation results should be used with precautions. In general, no problems are expected when the option is activated

but it should be taken into account that the simulator was not tested when the option is used with the EOS module. Therefore, some validation of the simulation results might be necessary in order to achieve a required degree of confidence in the results.

- NOT APPLICABLE means that the option in general cannot be applied with EOS module.
- DISABLED means that the option cannot be activated with EOS module. If it is activated the simulation will be terminated with an error in the RUNSPEC section. Further, code developments are necessary in order to make the option available.

3.1. BINMIXT

Option		Availability
CAPPRES	-	DISABLED
DUALPERM	-	USE WITH PRECAUTIONS
DUALPORO	-	USE WITH PRECAUTIONS
DUCTILE	-	DEVELOPED AND TESTED
DYNFRACK	-	DEVELOPED AND TESTED
ENDSCALE	-	DISABLED
GRAVDR	-	DISABLED
HCFLUID	-	DISABLED
HCROCK	-	DEVELOPED AND TESTED
ISOTHERM	-	DISABLED
MECHDISP	-	USE WITH PRECAUTIONS
MINC	-	USE WITH PRECAUTIONS
MUSEGWEL	-	USE WITH PRECAUTIONS
ROCKCOMP	-	DEVELOPED AND TESTED
SURFTENS	-	DISABLED
DIFFCHEM	-	DISABLED

3.2. BLACKOIL

Option		Availability
CAPPRES	-	DEVELOPED AND TESTED
DUALPERM	-	DEVELOPED AND TESTED
DUALPORO	-	DEVELOPED AND TESTED
DUCTILE	-	NOT APPLICABLE
DYNFRACK	-	DEVELOPED AND TESTED
ENDSCALE	-	DEVELOPED AND TESTED
GRAVDR	-	DEVELOPED AND TESTED
HCFLUID	-	NOT APPLICABLE
HCROCK	-	NOT APPLICABLE
ISOTHERM	-	NOT APPLICABLE
MECHDISP	-	USE WITH PRECAUTIONS
MINC	-	DEVELOPED AND TESTED
MUSEGWEL	-	DEVELOPED AND TESTED
ROCKCOMP	-	DEVELOPED AND TESTED
SURFTENS	-	DEVELOPED AND TESTED
DIFFCHEM	-	NOT APPLICABLE

3.3. COMPS

Option		Availability
CAPPRES	-	DEVELOPED AND TESTED
DUALPERM	-	USE WITH PRECAUTIONS
DUALPORO	-	DEVELOPED AND TESTED
DUCTILE	-	USE WITH PRECAUTIONS
DYNFRACK	-	USE WITH PRECAUTIONS
ENDSCALE	-	DEVELOPED AND TESTED
GRAVDR	-	USE WITH PRECAUTIONS
HCFLUID	-	USE WITH PRECAUTIONS
HCROCK	-	USE WITH PRECAUTIONS
ISOTHERM	-	DEVELOPED AND TESTED
MECHDISP	-	USE WITH PRECAUTIONS
MINC	-	USE WITH PRECAUTIONS
MUSEGWEL	-	USE WITH PRECAUTIONS
ROCKCOMP	-	DEVELOPED AND TESTED
SURFTENS	-	DISABLED
DIFFCHEM	-	DEVELOPED AND TESTED

3.4. GASSTORE

Option		Availability
CAPPRES	-	DEVELOPED AND TESTED
DUALPERM	-	USE WITH PRECAUTIONS
DUALPORO	-	USE WITH PRECAUTIONS
DUCTILE	-	USE WITH PRECAUTIONS
DYNFRACK	-	USE WITH PRECAUTIONS
ENDSCALE	-	DEVELOPED AND TESTED
GRAVDR	-	USE WITH PRECAUTIONS
HCFLUID	-	DISABLED
HCROCK	-	DEVELOPED AND TESTED
ISOTHERM	-	DEVELOPED AND TESTED
MECHDISP	-	DEVELOPED AND TESTED
MINC	-	USE WITH PRECAUTIONS
MUSEGWEL	-	USE WITH PRECAUTIONS
ROCKCOMP	-	DEVELOPED AND TESTED
SURFTENS	-	DISABLED
DIFFCHEM	-	NOT APPLICABLE

3.5. SIMPLMOD

Option		Availability
CAPPRES	-	DEVELOPED AND TESTED
DUALPERM	-	USE WITH PRECAUTIONS
DUALPORO	-	USE WITH PRECAUTIONS
DUCTILE	-	DEVELOPED AND TESTED
DYNFRACK	-	DEVELOPED AND TESTED
ENDSCALE	-	DISABLED
GRAVDR	-	NOT APPLICABLE
HCFLUID	-	DISABLED
HCROCK	-	DEVELOPED AND TESTED
ISOTHERM	-	DEVELOPED AND TESTED
MECHDISP	-	USE WITH PRECAUTIONS
MINC	-	USE WITH PRECAUTIONS
MUSEGWEL	-	USE WITH PRECAUTIONS
ROCKCOMP	-	DEVELOPED AND TESTED
SURFTENS	-	NOT APPLICABLE
DIFFCHEM	-	NOT APPLICABLE

3.6. T2EOS1

Option		Availability
CAPPRES	-	DEVELOPED AND TESTED
DUALPERM	-	DEVELOPED AND TESTED
DUALPORO	-	DEVELOPED AND TESTED
DUCTILE	-	USE WITH PRECAUTIONS
DYNFRACK	-	USE WITH PRECAUTIONS
ENDSCALE	-	DISABLED
GRAVDR	-	USE WITH PRECAUTIONS
HCFLUID	-	DEVELOPED AND TESTED
HCROCK	-	DEVELOPED AND TESTED
ISOTHERM	-	DEVELOPED AND TESTED
MECHDISP	-	USE WITH PRECAUTIONS
MINC	-	DEVELOPED AND TESTED
MUSEGWEL	-	USE WITH PRECAUTIONS
ROCKCOMP	-	DEVELOPED AND TESTED
SURFTENS	-	DISABLED
DIFFCHEM	-	NOT APPLICABLE

3.7. USEREOS

Option		Availability
CAPPRES	-	USE WITH PRECAUTIONS
DUALPERM	-	USE WITH PRECAUTIONS
DUALPORO	-	USE WITH PRECAUTIONS
DUCTILE	-	USE WITH PRECAUTIONS
DYNFRACK	-	USE WITH PRECAUTIONS
ENDSCALE	-	DISABLED
GRAVDR	-	USE WITH PRECAUTIONS
HCFLUID	-	USE WITH PRECAUTIONS
HCROCK	-	DEVELOPED AND TESTED
ISOTHERM	-	DEVELOPED AND TESTED
MECHDISP	-	USE WITH PRECAUTIONS
MINC	-	USE WITH PRECAUTIONS
MUSEGWEL	-	USE WITH PRECAUTIONS
ROCKCOMP	-	DEVELOPED AND TESTED
SURFTENS	-	DISABLED
DIFFCHEM	-	NOT APPLICABLE

4 Thermodynamic states of fluid

4.1. Overview

There are several precalculated states of the fluid which can be used to define parameters of the injected fluid using the WELLINJE or SRCINJE keywords. The name of the fluid state (as given in the tables below) can be input using the 8th (WELLINJE) or 3rd (SRCINJE) argument of the keywords in order to redefine parameters in the stock tank associated with the well of point source.

If parameters in a stock tank or a pipe segment are not defined by the end of the INIT section then the default fluid state is used to initialise parameters of the fluid.

The default stock tank pressure is 1 atm. This value can be redefined using the TANKPRES keyword.

4.2. BINMIXT

Predefined states of the fluid		
Name		**Description**
COMP2	-	Pure second component at default stock tank pressure and temperature = 20C=293.15K (default fluid state)

4.3. BLACKOIL

Predefined states of the fluid		
Name		**Description**
WATER	-	Water at default stock tank pressure (default fluid state)
OIL	-	Dead oil (no dissolved gas) at default stock tank pressure
GAS	-	Dry gas (no vaporised oil) at default stock tank pressure

4.4. COMPS

Predefined states of the fluid		
Name		**Description**
WATER	-	Water at default stock tank pressure (default fluid state if WATER is enabled)
COMPi	-	Component #i at default stock tank pressure (default fluid state is COMP1, if WATER is disabled)

4.5. GASSTORE

Predefined states of the fluid		
Name		**Description**
WATER	-	Pure H2O at default stock tank pressure and temperature = 20C=293.15K (default fluid state)
GAS	-	Pure gas at default stock tank pressure and temperature = 20C=293.15K

4.6. SIMPLMOD

Predefined states of the fluid		
Name		**Description**
DEFAULT	-	Fluid at default stock tank pressure and temperature = 20C=293.15K (default fluid state)

4.7. T2EOS1

Predefined states of the fluid		
Name		**Description**
WATER	-	H2O at default stock tank pressure and temperature = 20C=293.15K (default fluid state)

4.8. USEREOS

Predefined states of the fluid		
Name		**Description**
DEFAULT	-	1st component at default stock tank pressure and temperature = 20C=293.15K (default fluid state)

5 Well rates

5.1. Kernel

Available well rates		
Name		**Description**
MASS	-	Mass rate of the fluid at the inlet of the pumping device
RATE	-	Volumetric rate of the fluid at surface conditions (defined for wells). RATE=RATEIN for injection wells, RATE=RATEOUT for production wells
GRUP	-	Well is under group control
RESV	-	Volumetric rate of the fluid at reservoir conditions (defined for wells). RESV=RATEOUT for injection wells, RESV=RATEIN for production wells
BHP	-	Constant bottom-hole pressure (for wells)
RATEIN	-	Volumetric rate of fluid at the inlet of pumping device (define for point sources and pumping devices)
RATEOUT	-	Volumetric rate of fluid at the outlet of pumping device (define for point sources and pumping devices). This target is not recommended for use.

5.2. BLACKOIL

Available well rates		
Name		**Description**
WRAT	-	Volumetric flow rate of water. The volume is associated with the parameters defined by the DENSITY keyword
ORAT	-	Volumetric flow rate of oil. The volume is associated with the parameters defined by the DENSITY keyword
GRAT	-	Volumetric flow rate of gas. The volume is associated with the parameters defined by the DENSITY keyword

Available well rates		
Name		**Description**
LRAT	-	Volumetric flow rate of liquid (water+oil). The volume is associated with the parameters defined by the DENSITY keyword

5.3. COMPS

Available well rates		
Name		**Description**
WRAT	-	Volumetric flow rate of water. The volume is associated with the parameters defined by the DENSITY keyword
ORAT	-	Volumetric flow rate of oil.
GRAT	-	Volumetric flow rate of gas.
LRAT	-	Volumetric flow rate of liquid (water+oil).

5.4. GASSTORE

Available well rates		
Name		**Description**
WRAT	-	Volumetric flow rate of water. The volume is associated with the parameters defined by the DENSTCND keyword
GRAT	-	Volumetric flow rate of gas. The volume is associated with the parameters defined by the DENSTCND keyword

6 Units

At present there are four systems of units in MUFITS, namely DEFUNITS (default option), METRIC, FIELD, and LAB systems. A particular system can be selected by using the DEFUNITS, METRIC, FIELD, and LAB keywords, respectively. Some of the units used for particular physical parameters are given in the table below. If a parameter is not in the table then its units are absolute for all systems of units.

List of units				
Description	**DEFUNITS**	**METRIC**	**FIELD**	**LAB**
Pressure	MPA	BAR	PSIA	ATM
Viscosity	KG/M/S	CP	CP	CP
Formation volume factor of liquid phase	RM3/SM3	RM3/SM3	RB/STB	RCC/SCC
Formation volume factor of gas phase	RM3/SM3	RM3/SM3	RB/MSCF	RCC/SCC
Density	KG/M3	KG/M3	LBM/CF	G/CC
Gas-oil ratio	NODIM	NODIM	MSCF/STB	NODIM
Oil-gas ratio	NODIM	NODIM	STB/MSCF	NODIM
Isothermal compressibility	1/MPA	1/BAR	1/PSIA	1/ATM
Viscosibility	1/MPA	1/BAR	1/PSIA	1/ATM
Distance; Length; Coordinate	M	M	FT	CM
Liquid volume at surface conditions	M3	M3	STB	SCC
Gas volume at surface conditions	M3	M3	MSCF	SCC
Liquid volumetric flow rate at surface conditions	M3/DAY	M3/DAY	STB/DAY	SCC/HR

List of units				
Description	**DEFUNITS**	**METRIC**	**FIELD**	**LAB**
Gas volumetric flow rate at surface conditions	M3/DAY	M3/DAY	MSCF/DAY	SCC/HR
Transmissibility	DEFUNITS	METRIC	FIELD	LAB
Dimensionless parameter (internal use)	NODIM	NODIM	NODIM	NODIM
Temperature (associated with the TEMP mnemonic)	K	K	R	K
Specific enthalpy (specific energy)	MJ/KG	MJ/KG	BTU/LBM	J/G
Molar enthalpy (molar energy)	KJ/MOL	KJ/MOL	KJ/MOL	KJ/MOL
Distance; Length; Coordinate (on the map)	METERS	METRES	FEET	CM
Mass flow rate	TON/DAY	TON/DAY	LBM/DAY	G/HR
Mass	TON	TON	LBM	G
Time	DAYS	DAYS	DAYS	HRS
Energy	MW*DAY	MJ	BTU	J
Energy flow rate	MW	MJ/DAY	BTU/DAY	J/HR
Molar density	G/MOL	G/MOL	G/MOL	G/MOL
Specific volume	CC/G	M3/KG	CF/LBM	CC/G
Sigma-factor	1/M2	1/M2	1/FT2	1/CM2
Surface tension	N/M	N/M	LB/IN	DYNE/CM
Velocity	M/DAY	M/DAY	FT/DAY	CM/HRS
Dispersivity	M2/DAY	M2/DAY	FT2/DAY	CM2/HRS
Molar volume	CM3/MOL	CM3/MOL	CM3/MOL	CM3/MOL
Specific heat capacity	KJ/KG/K	KJ/KG/K	BTU/CF/F	J/G3/K
Molar volume (for particular keywords)	M3/KG-M	M3/KG-M	CF/LB-M	CM3/G-M
Relative temperature	C	C	F	C
Gravity change	MCGAL	MCGAL	MCGAL	MCGAL
Elastic modulus	GPA	GPA	GPA	GPA
Ground displacement	CM	CM	IN	CM
Parachor	PARUNITS	PARUNITS	PARUNITS	PARUNITS

List of units				
Description	**DEFUNITS**	**METRIC**	**FIELD**	**LAB**
Diffusivity	M2/DAY	M2/DAY	FT2/DAY	CM2/DAY
k*h map	MD*M	MD*M	MD*FT	MD*CM

The transcripts for some abbreviations are as follows:

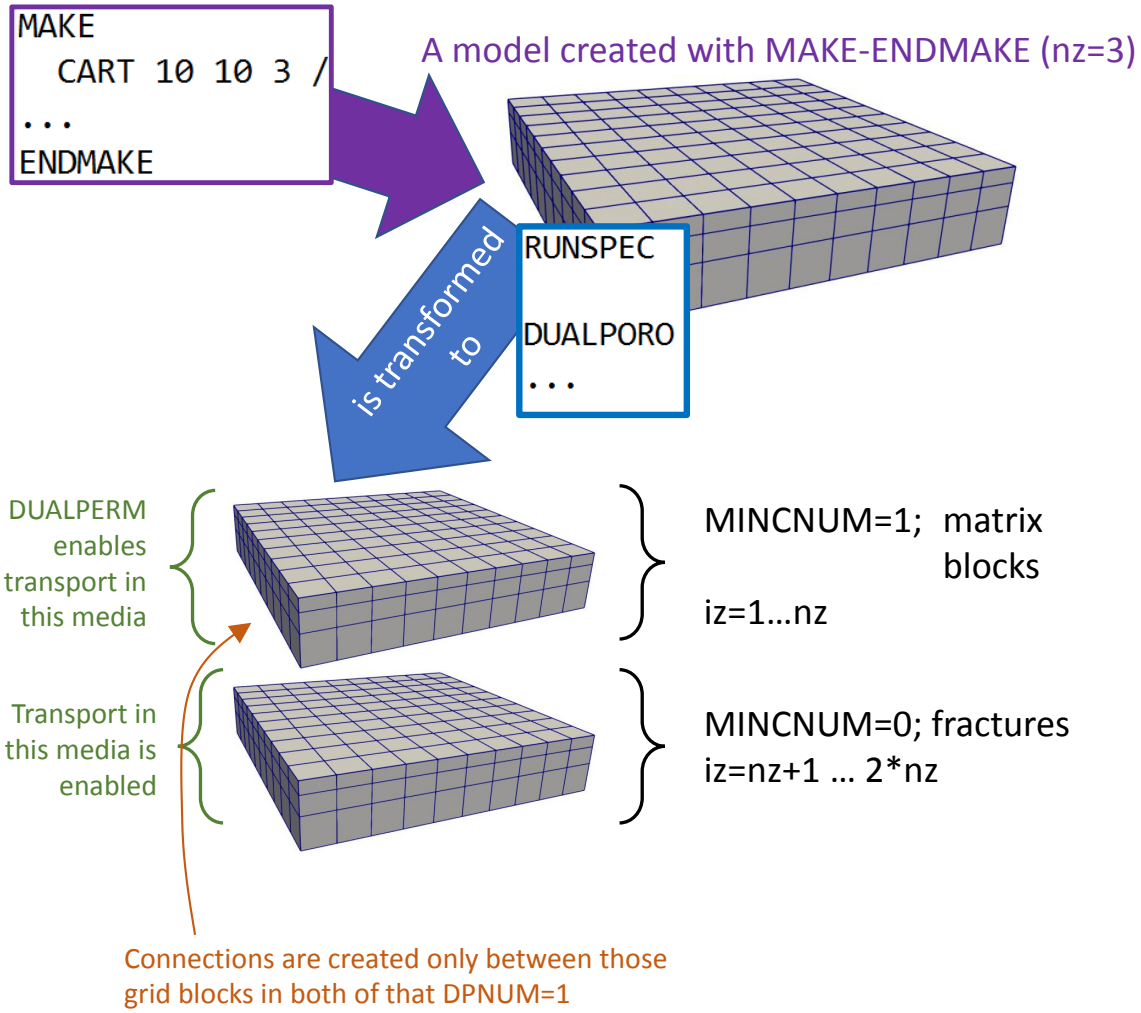
- ATM – atmosphere;
- BAR – bar;
- BTU – British thermal unit;
- CC – cubic centimeter;
- CF – cubic feet;
- CM – centimeter;
- CP – centipoise;
- DAY – day;
- DAYS – days;
- F – degree Fahrenheit;
- G – gram;
- GPA – gigapascal;
- HR – hour;
- HRS – hours;
- J – joule;
- K – degree Kelvin;

- KG – kilogram;
- KJ – kilojoule;
- KW – kilowatt;
- LBM – pound;
- M – meter;
- M3 – cubic meter;
- MCGAL – microGal;
- MJ – megajoule;
- MOL – mole;
- MPA – megapascal;
- MSCF – 10^3 cubic feet (at surface conditions);
- MW – megawatt;
- NODIM – dimensionless parameter;
- PSIA – pounds per square inch;
- R – degree Rankine;
- RB – barrel at reservoir conditions;
- RCC – cubic centimeter (at reservoir conditions);
- RM3 – cubic meter (at reservoir conditions);
- S – second;
- SCC – cubic centimeter (at surface conditions);

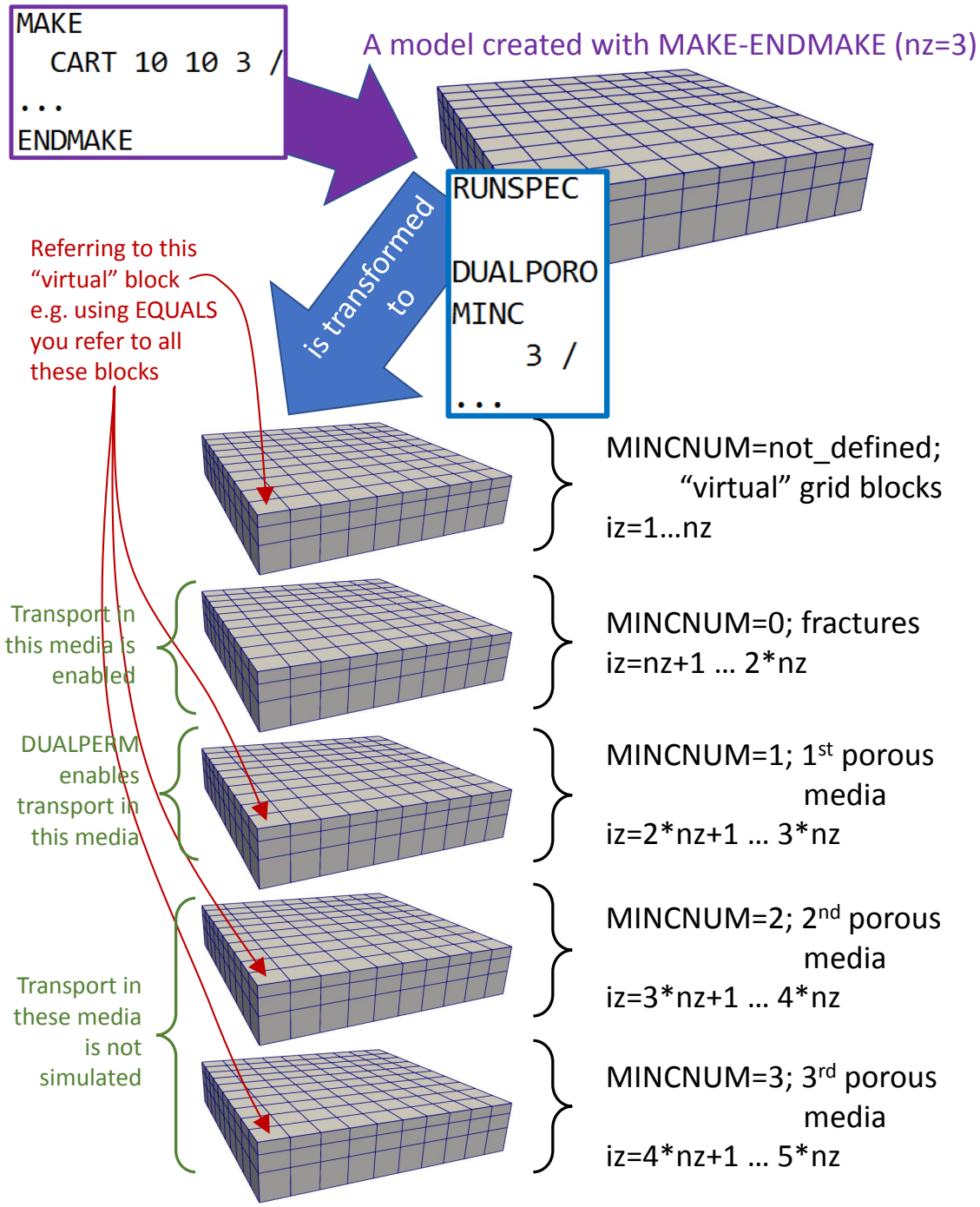
- SM3 – cubic meter (at surface conditions);
- STB – stock tank barrel;
- TON – metric ton.

7 Useful Notes

7.1. Grid blocks indexation in dual-porosity simulations

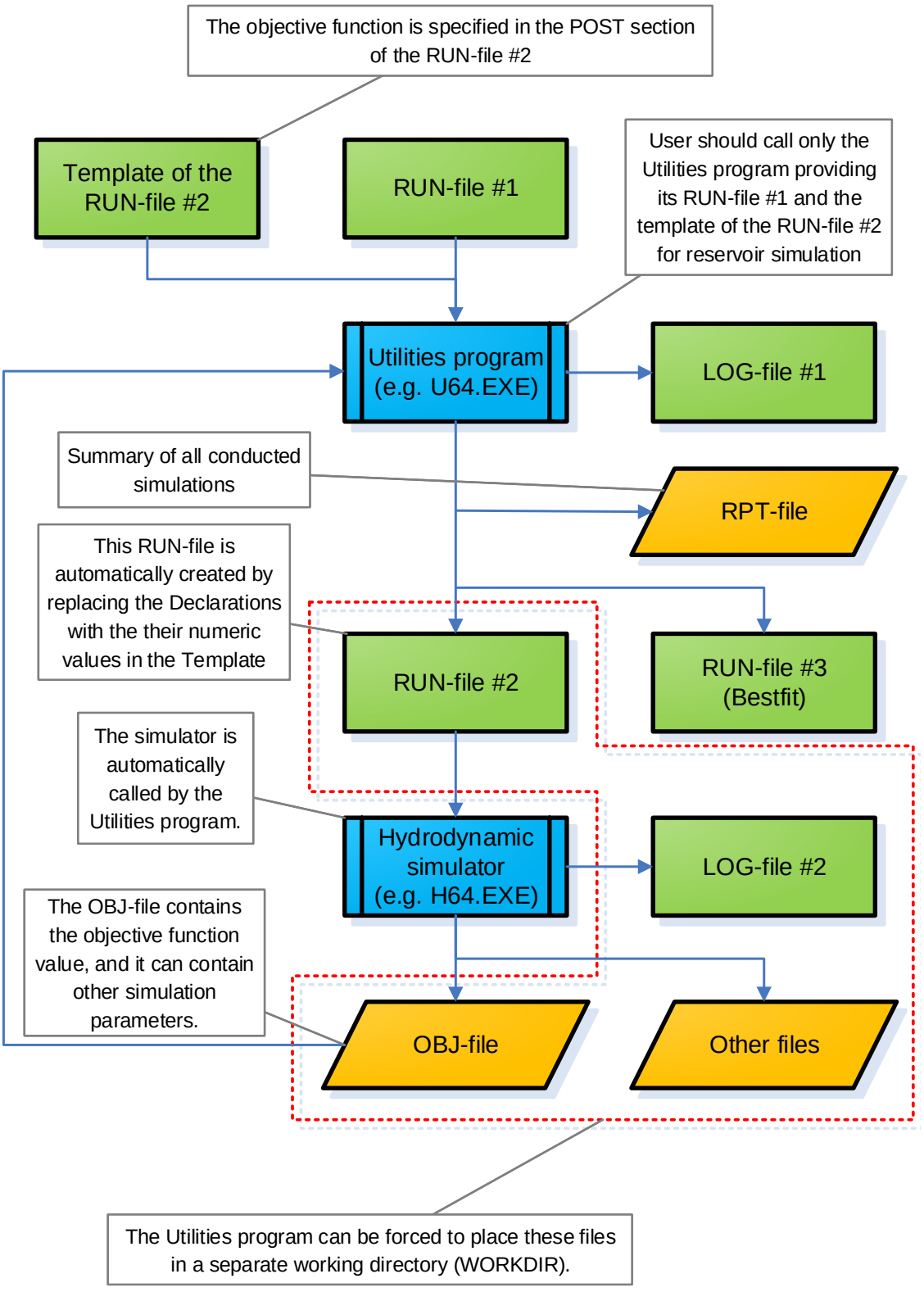


A particular grid block can be referred to by the iz index or the whole media can be referred to by using MINCNUM region in e.g. the EQUALREG keyword.



A particular grid block can be referred to by the iz index or the whole media can be referred to by using MINCNUM region in e.g. the EQUALREG keyword.

7.2. Inverse modelling



7.3. Library of components for the COMPS module

At present the built-in library of EoS coefficients contains the following sections:

- BR3 – Brusilovskii EoS
- PR3 – Peng-Robinson EoS with shift correction
- SRK3 – Soave-Redlich-Kwong EoS with shift correction
- SW – Søreide-Whitson EoS for modelling aqueous solutions

Some transcripts used in the build-in library

- H2O - water
- N2 - nitrogen
- H2 - hydrogen
- O2 - oxygen
- H2S - hydrogen sulfide
- CO2 - carbon dioxide
- CO - carbon monoxide
- C1 - methane
- C2 - ethane
- C3 - propane
- IC4 - i-butane (2-methylpropane)
- C4 - butanes

- NC4 - n-butane
- IC5 - i-pentane (2-methylbutane)
- C5 - pentanes
- NC5 - n-pentane
- BEN - benzene
- C6 - hexanes
- TOL - methylbenzene
- C7 - heptanes
- C8 - octanes
- C9 - nonanes
- C10 - decanes
- C11 - undecanes
- C12 - dodecanes
- C13 - tridecanes
- C14 - tetradecanes
- C15 - pentadecanes
- C16 - hexadecanes
- C17 - heptadecanes
- C18 - octadecanes
- C19 - nonadecanes

- C20 - eicosanes
- ...

7.4. Options for modeling edge-water drive

Generally, we can distinguish four options for modeling the edge-water drive, which can be used with MUFITS. The sketch drawings in Figures 7.1–7.4 demonstrate the ideas behind each such option.

The option 1 utilizes the standard approach that is based on the pore volume multipliers (Figure 7.1). The volume of the boundary grid blocks can be multiplied by specific factors to account for the discarded sectors of the reservoir. It is convenient to use arithmetic operations to specify MULTPV in the GRID section. By default MULTPV=1 in all grid blocks. Alternatively, one can multiply the entire cell volume by applying a factor to CELLVOL. If extremely large MULTPV are required, then it is better to set ACTNUM=2 in all boundary grid blocks, which results in fixing the parameters in these blocks over time. In specific cases, the option 1 is not convenient, because the boundary grid blocks are utilized for modeling the aquifer rather than the flow in the domain.

The option 2 is an improvement of the option 1. By using the BOUNDARY keyword, the virtual (ghost) grid blocks can automatically be created by the software (Figure 7.2). If the SAMESIZE option of BOUNDARY is used, then the dimensions of the virtual grid blocks equal those of the adjacent internal grid blocks of the domain. Each virtual grid block is connected only to a single internal grid block. The connections between the virtual grid blocks are not created by MUFITS. After the virtual blocks are created, their pore volumes can be changed by MULTPV to model a specific aquifer.

The option 3 is a further development of the option 2 (Figure 7.3). One

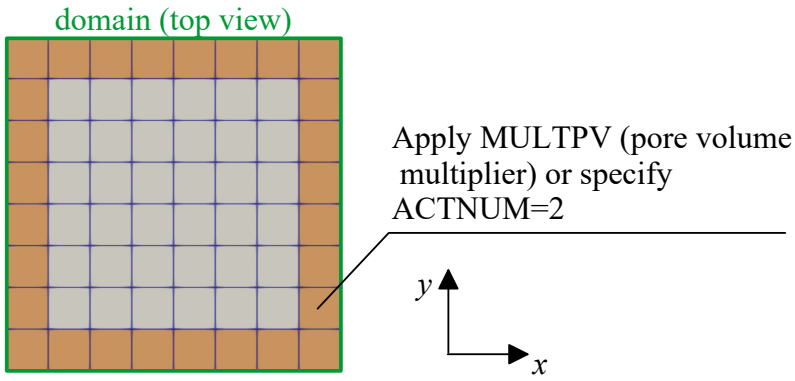


Figure 7.1: This schematic demonstrates Option 1 for modeling an aquifer in the case of the edge-water drive. It is a standard method based on the pore-volume multiplier MULTPV applied to the boundary grid blocks.

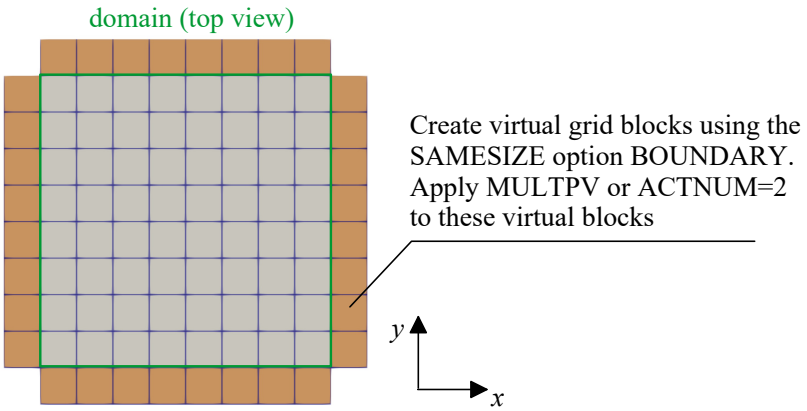


Figure 7.2: This schematic demonstrates Option 2 for modeling an aquifer in the case of the edge-water drive. It is an improved method based on creating virtual (ghost) grid blocks connected to the lateral boundaries.

can automatically adjust the volumes of the virtual grid blocks by using the option CYLINTYP of BOUNDARY. By specifying the center of the aquifer and its radius, the volumes of the virtual grid blocks are created in a fashion as it is shown in Figure 7.3. So, it is similar to the option 2, but the virtual grid blocks dimensions are different. Again, the virtual grid blocks are not

connected with each other.

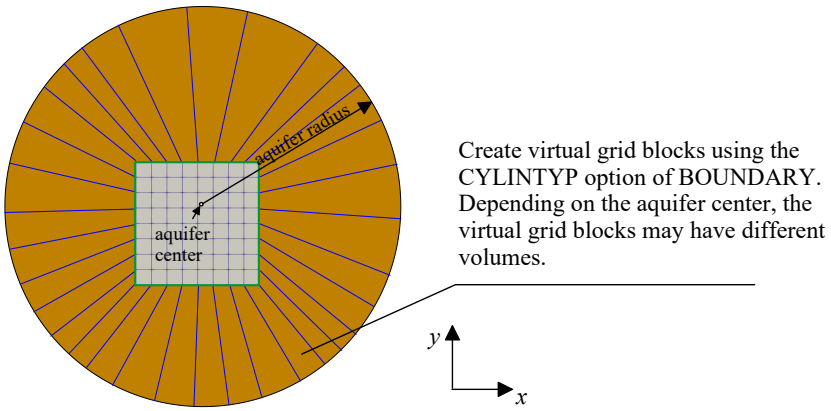


Figure 7.3: This schematic demonstrates Option 3 for modeling an aquifer. This method allows an automated scaling of the boundary grid blocks' volumes by introducing the concept of the center and the radius of the aquifer. Note that the virtual grid blocks are connected with each other.

The option 4 is based on the conventional implementation of the 1D numerical aquifer (Figure 7.4). It is assumed that the aquifer encompasses the simulation domain. The aquifer grid has a resolution only in the radial direction (which is in the direction away from the domain ''center"). Note that the simulator creates virtual (ghost) grid blocks to simulate the aquifer, rather than use for that purpose some of the domain grid blocks. The sequence of the aquifer cells must be specified by the AQUNUMER keyword. The sequence of the specific lines in that table define the sequence of the grid blocks. Only the first block is connected to the domain grid blocks by either of the keywords BOUNDARY, AQUCONN or AQUCON, where BOUNDARY provides more flexibility in the specification. The AQUCONN and AQUCON keywords are just a simpler form of the BOUNDARY keyword.

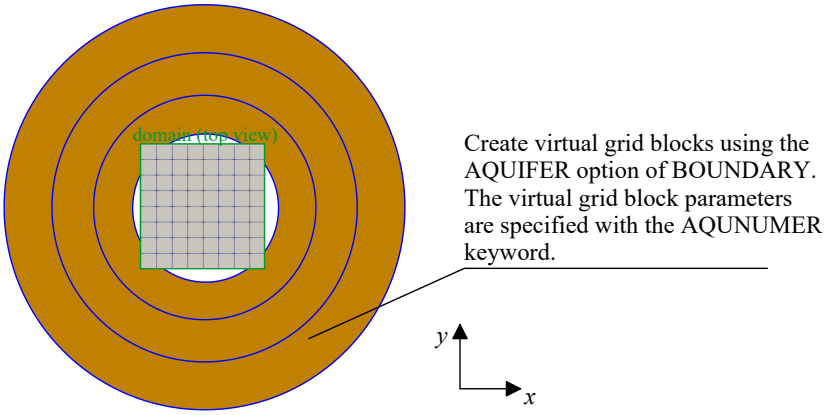


Figure 7.4: This schematic demonstrates Option 4 for modeling an aquifer. It utilizes the standard concept of a numerical aquifer represented by a 1-D sequence of virtual grid blocks around the domain.

8 New Developments

8.1. Release 2024.A

1. The compositional module COMPS can now take into account the CO_2 solubility in the water phase. The option is enabled by the CO2SOL keyword placed in section RUNSPEC. The PVT properties of water in such case must be supplied by the keyword SOLUBILI. This table can be exported from the GASSTORE module by using the GSTOSOLU keyword.

2. The numerical aquifers can now be specified by using the keywords AQUNUMER, AQUCON and AQUCONN. The simulator creates a row of virtual grid blocks to model the inflow from the aquifer. The aquifer can be saturated with any fluid and can even be used to simulate heat losses in thermal runs.

3. The BOUNDARY keyword was extended to account for sepcific boundary conditions. The new functionaly include some extended options for modeling aquifers and the flux boundary conditions.

4. For injection wells the simulator can calculate the tubing-head pressure (THP). The vertical flow performance tables that relate BHP with THP and flow rate can be specified by the VFPINJ keyword. Other keywords associated with VFP tables are VFPCHK and VFPSTBLZ. The target or limit THP pressures for each well can be specified by the WELLINJE keyword. The VFPINJ tables can be associated with wells by the WELLINJE keyword.

5. Now wells and point sources are merged into groups. Every new well should be associated with a group by the WELSPECS or WELSPECC keyword. If the group for a well is not specified, then the well is assumed belonging to the uppermost group FIELD in the group tree. The group parameters can be reported by the RPTGRUP and POSTGRUP keywords.

6. Group control for injection wells is now available. It can be specified by the GRUPINJE keyword. The wells in group can be both by individual control and under group control. The latter option is enabled by using GRUP for the well target in WELLINJE keyword. The flow rate within a group can be balanced by both the potential injection rates and fixed tubing-head pressure for a group. Note the calculation of the group and well potentials is enabled by the SRCPOTEN keyword.

7. The ARITH and ARITHCND can now be used within the brackets MAKE-ENDMAKE.

8. The path for the include files can now be explicitely specified by using the PATH keyword.

9. The set of available arithmetic operations was extended by LINEXY, POLYLINE, POLYGON, UF3D. See the associated POLYLINE, UF3DSPEC, and LOADGRD keywords.

8.2. Release 2022.B

1. The hysteresis model for the realtive permeability has been implemented. The model is enabled by the HYSTER keyword placed in section RUNSPEC. The hysteresis options can be set up with the EHYSTR keyword. The saturation tables (i.e. the tables specified within SAT brakets) are

associated with the drainage curves by the SATNUM regions, whereas they are associated with the imbibition curves by the IMBNUM regions. The end-points scaling option (see ENDSCALE) is compatible with the hysteresis options in BLACKOIL, COMPS and GASSTORE modules.

2. A much more convinient methods for operations on arrays has been introduced. This can be done through arithmetic operations on arrays and scalar variables by the ARITH and ARITHREG keywords.

3. The meaning of the AUTO and NOAUTO keywords has been changed. The old versions of the keywords are available with the AUTOPART and NOAUTOPA keywords. The new versions of the keywords make a more convinient specification of the automatic operations.

4. The CPVARAUT, OPAUTO, OPAUTREG, OPAUTNAM, EVAUTO, EVAUTREG, OPVARAUT keywords are deprecated, althoug they are still recognized for backward compatibility. The older functionality can be achived by placing AUTO ahead of CPVAR, OPERATE, OPERAREG,OPERANAM, EV, OPVAR, respectively.

5. The ACTION brakets are available now to specify conditional keywords, i.e the keywords executed only under required conditions.

6. A map of a quantity can be calculated by the software. The calculation can be performed by the MAPPER keyword.

8.3. Release 2022.A

1. An option for modelling the molecular diffusion caused by the chemical potential gradients is now available for the COMPS module. The option is enabled by the DIFFCHEM keyword. The coefficients of molecular diffusion can be specified by the DIFFAGAS and keywords.

2. Separators can now be used with the COMPS module. The seprator conditions can be specified by the FIELDSEP and SEPCOND keywords. The seprator conditions can be associated with the wells and groups by the WSEPCOND and GSEPCOND keywords.

3. The intial fluid equilibrium in the COMPS module can now be specified by the COMPVD keyword.

4. Well tracks for diferent branches of wells can be specified by the WELTRACK keyword. The corresponding completion intervals are specified by the COMPDATM keyword. The well location can be introduced by the coordinates of its head (see WELSPECC keyword).

5. A new automatic time step control based on the target variables change is now avaible. It can be enabled by the DVARCTRL keyword.

6. The ENDSCALE options is now available with the COMPS and GASSTORE EoS-modules.

7. The simulator can read the EGRID file by using keyword GDFILE.

8. The syntax of CARFIN has been changed.

8.4. Release 2021.C

1. The simulator capabilities are extended by adding the porosity waves options, i.e. the POROWAVE option. The option is useful for an improved modelling of magmatic fluid transport above, below, and through the brittle-ductile transition. The option is based on a mathematical model of fluid flow in a compacting porous medium. It can be tuned to account for the thermal softening of rocks and the plastic deformation of the solid matrix through decompaction weakening. The option is enabled by

placing the POROWAVE keyword in the RUNSPEC section. The material properties of the rock and their dependence on temperature can be specified by the PWVROCK keyword within the ROCK-ENDROCK brackets. The decompaction weakening can be simulated by using the PWVPEMLT keyword.

2. The Utilities program was extended by adding the section SATFC-ENDSATFC for generating the saturation function tables. Several models of the the relative permeability and capillary pressure are incorporated in the software to ease the data preparation to the hydrodynamic modelling.

3. Some of the new keywords are: the PV2TRMLT relating the permeability and porosity changes; the PWVMODEL keyword for choosing the porosity waves model; and the ROCKDENS keyword for entering the rock grain density.

8.5. Release 2021.B

1. The COMPS module has been extended by introducing a new section SW of the built-in library of EoS coefficients. The SW EoS is activated by specifying the SW argument for the EOSTATE keyword. The SW section of the library incorporates a set of correlations for modelling phase equilibria of aqueous solutions by using the approach of Søreide and Whitson. The extension is useful for modelling various gases co-injection into saline aquifers including the applications related to the cushion gas substitution with CO_2. A modified SW EoS is used by the default option. The simulator can be switched to using the original correlations developed by Søreide and Whitson (1992) by using SWORIG keyword.

2. The new keywords of the noted extension include RSMOL, RXS, RXSM for characterising the salinity of formation brine. One of these keywords must be specified within every EOS-ENDEOS brackets.

3. The initial capillary equilibrium can now be specified within the COMPS module by the keywords EQUIL, TEMPVD, TEMPCVD, and ZMFVD.

4. An improved calculation of the aqueous phase density can be activated by the keyword FTDOIL.

5. The fluid components can now be characterised by the CHARACT keyword.

6. The standard conditions can be changed by the STCOND keyword.

8.6. Release 2021.A

1. The USEREOS module is available now. The module allows for linking the code with a provided shared library. A new shared library can be programmed by simulator users. The library exports PVT properties of the fluid. Optionally, the library can also export the saturation functions.

2. Now the simulator allows for auxiliary scalar variables. These are considered as global quantities. The variables are defined by the AUXVAR keyword and reported by the RPTVAR and POSTVAR keywords. The arithmetic operations with the scalar variable can be done with the OPVAR and OPVARAUT keywords. The parameters of the grid blocks, wellbores, regions, etc. can be copied to a scalar variable by the CPVAR and CPVARAUT keywords. Integral quantities can also be calculated over a region of grid cells and then copied to a scalar variable by the EV, EVREG and EVNAM keywords.

3. Now the simulator is capable of automated arithmetic operations on different quantities, including the grid block parameters, the wellbore parameters, the field parameters, the auxiliary variables, etc. The automated operations are applied immediately before any new summary file is saved. The operations are applied in the sequence of their loading by the OPAUTO, OPAUTNAM, OPAUTREG, CPVARAUT and OPVARAUT. The list of automated operations can be erased by the OPAUTPAR keyword.

4. The syntax of the AUXARRAY keyword has changed.

8.7. Release 2020.B

1. History matching to the parameters at the observation points is possible by using the RPTOOBS, OBJOBS, and OBJTOBS keywords.

2. The new mnemonics are DDENGRAV and DVOLDISP. The first mnemonic is the bulk density change with respect to the reference value used in the calculation of the gravity changes. The second mnemonic is the relative volume change used in the calculation of the ground displacement.

3. Several bug fixes.

8.8. Release 2020.A

1. The preliminary version of the COMPS EOS-module has been made available. The module is intended for the thermal and compositional modelling when the fluid properties are predicted by a cubic equation of state.

2. The COMPS EOS-module is supplied with the build-in library of components and binary interaction coefficients which can be loaded from the library by the LIBCOMPS keyword. The library includes coefficients for the three-parametric Peng-Robinson (PR3) and Soave-Redlich-Kwong (SRK3) equations of state. Particular parameters or components of the thermodynamic system can be supplied by user whereas others can be loaded from the library.

3. The COMPS EOS-module can be used for both thermal and isothermal compositional simulations. The latter is invoked by enabling the ISOTHERM option.

4. The new object type Observation Point has been introduced. The observation points are intended for reporting the gravity changes, the ground displacement and other additional parameters. The associated keywords are OBSPOINT, OBSSPECM, RPTOBS, POSTOBS, and POSTOBSM. The coordinates of the observation points can be reported using the XOBS, YOBS, ZOBS, and DEPTHOBS mnemonics.

5. The gravity changes can now be computed by the simulator in an automated manner. The gravity changes can be reported for the observation points (see OBSPOINT). The keywords controlling the computation are GRAVIMET and SETDGRAV. The mnemonics for the gravity changes are GRAVMT, GRAVMTX, GRAVMTY, and GRAVMTZ.

6. The ground displacement (the surface uplift and subsidence) can now be computed by the simulator in an automated manner. The ground displacement can be reported for the observation points (see OBSPOINT). The keywords controlling the computation are GRDDISP, GRDPARAM, and SETREFPT. The mnemonics for the displacements

are UTOT, UX, UY, and UZ. The positive values of UZ correspond to the uplift.

7. Improved option for coupled simulations. The syntax of the COUPLED keyword has changed.

8. Some other new keywords are SRCSPECM, SRCHEAT, and TERMINAL.

8.9. Release 2019.B

1. The simulation software now includes the Utility program (U-program). This program can be used for history matching (see OPTIM-ENDOPTIM and Sect. 1.18) or automated preparation of SCHEDULE section (see SCHED-ENDSCHED and Sect. 1.19). The auxiliary calculations or data preparations are expected to be included in this program in future developments.

2. The same RUN-file can be used for the H-program, U-program and I-program. The associated keywords are HPROG, IPROG, UPROG and ALLPROG.

3. A preliminary software development for the inverse modelling, the history matching, the flow optimization and automated parametrization studies is available now. See Sect. 1.18 and 7.2 for details. This development is not finished yet and improvements in future are expected. The H-program keywords associated with this option (for the objective function output) are OBJBLOC, OBJTFLD, OBJPARAM and other keywords starting with the OBJ prefix.

4. Automated preparation of the SCHEDULE section of the RUN-file is now possible. This can be done within the SCHED-ENDSCHED brack-

ets of the U-program. This development is useful when variable in time production or injection rates need to be specified. See Sect. 1.19 for details.

5. The option for modelling ductile behavior of rocks at elevated temperatures has been developed. The associated keywords are DUCTILE, DUCTTAB, PRESLITH, PRESFAIL and others.

6. The option for modelling dynamic fractures has been developed. The associated keywords are DYNFRACK, DYNFRTAB, PRESFAIL, PERMXFR, PERMYFR, PERMZFR and others.

7. The mechanical dispersion option was developed. The new keywords associated with this options are MECHDISP, MDISPL, and MDISPT. The new application examples can be seen in Sect. 9.27, 9.28, 9.29, and 9.30.

8. The new options for primary variables initialisation with the BINMIXT module were introduced. See the ENTHPRIM and TEMPPRIM keywords.

9. The list of functions that can be used with the OPERATE, OPERAREG and others was extended. See also the UF2DNAME, UF2DSPE0, and UF2DSPEC keywords.

10. The restarting of hydrodynamic simulations is available now. The associated keywords are RESTART and CONTINUE.

11. The automated time step selection based on the CFL condition is possible now. See the CFLCTRL keyword.

12. The automated operations with arrays can be specified with the OPAUTO, OPAUTREG, OPAUTNAM, and OPAUTPAR keywords. This op-

tion is useful for reporting complicated functions of the arrays that are dynamically changed in simulation.

13. Some other new keywords are RPTFUNC, IF-ENDIF, GO2DATE, GO2STEP, GO2TIME, and POSTBLK2.

14. The exact simulation time can be specified with the DATES and START keywords.

15. The syntax of the RANDSEED keyword has slightly changed.

8.10. Release 2017.A

1. Double porosity and double permeability models were implemented for modelling fractured reservoirs. The new keyword associated with this option are: DUALPORO, DUALPERM, GRAVDR, SIGMA, SIGMAV, LTOSIGMA, LX, LY, LZ, DZMTRX, DZMTRXV, DPNUM.

2. Multiple interacting continua (multi-porosity) model was implemented. The new keywords associated with this option are: MINC, NMATRIX, NMATOPTS, MINCSIZE.

3. Capillary pressure can be activated with the T2EOS1 module. That is the CAPPRES can be used with the T2EOS1 keyword.

4. In the BLACKOIL module the capillary pressure (or surface tension) can depend on pressure. The new keywords associated with this option are: SURFTENS, STOG, STOW, STWG.

5. The first and second Stone"s formulas can be used for three-phase relative permeability in the BLACKOIL module. These options are activated with the STONE1 and STONE2 keywords.

9 Testing

9.1. Overview

In this chapter the input data for several comparative studies are summarised. These studies are used for testing the simulator. The name of the EOS module used for simulating a particular test is given in the brackets in the title of the corresponding section.

9.2. 1st SPE Comparative Study (BLACKOIL)

This problem is discussed in the paper: Odeh, A. 1981 Comparison of Solutions to a Three-Dimensional Black-Oil Reservoir Simulation Problem. JPT 33, 13-25. DOI:10.2118/9723-PA. See a description of the problem, input data files and visualisations under *the link*.

SPE1.RUN

```
1  ################################################################################
2
3                            1ST SPE COMPARATIVE STUDY
4
5      The problem is from SPE-9723 paper (case 2):
6      Odeh, A. 1981 Comparison of Solutions to a Three-Dimensional Black-Oil
7         Reservoir Simulation Problem. JPT 33, 13-25. DOI: 10.2118/9723-PA.
8
9      See description at: http://dx.doi.org/10.2118/9723-PA
10
11 --------------------------------------------------------------------------------
12
13     Last update on 28.04.2015
14     Program Version: 2015.C05
15
16 ################################################################################
17
18
19    Follow comments starting with $1$ to get better fit to the article.
20
21 --------------------------------------------------------------------------------
22
23    Note:
24    We type all comments in lowercase letters while all keywords and mnemonics
25       must be in uppercase letters.
26
27 --------------------------------------------------------------------------------
28
29    Note:
30 -- any line beginning with '!' or '--' is a comment line;
31    any line not beginning with an uppercase letter outside keyword
32 !  instruction is also a comment line.
33
34 --------------------------------------------------------------------------------
35
36    Note:
37    Any data line must be terminated by the slash '/' sign.
38
```

```
--------------------------------------------------------------------------------

   Note:
   We denote by the repeated symbols the following structural elements of this
      RUN-file:
      '###############' - delimits the sections of the RUN-file.
      '<<<<<<<<<<<<<<' - designates an opening bracket.
      '>>>>>>>>>>>>>>' - designates a closing bracket.
      '^^^^^^^^^^^^^^' - designates the keyword, which affects the order of the
                         data input for all the following keywords (both in the
                         current and in the following sections).
      '==============' - merge the keywords and the comments in paragraphs.

--------------------------------------------------------------------------------


RUNSPEC   ################### RUNSPEC section begins here ######################

BLACKOIL  ^^^^^^ We switch the program to the BLACKOIL EOS module ^^^^^^^^^^^^^^

                                       We enable:
WATER                                  - water phase;
OIL                                    - oil phase;
GAS                                    - gas phase;
DISGAS                                 - gas dissolution in oil phase.

FIELD     ^^^^^^^^^^^^ We use FIELD units ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
                                       Note, this keywords affects the units
                                       in many keywords below.

ROCKCOMP                               We enable rock compaction.

FAST                                   We use FAST option.

--OPTIONS                              $1$ Piecewise linear interpolation
--  53* 1 /                            of relative permeabilities. Use this
                                       option if you want a better fit to the
                                       SPE-9723 paper.

GRID      ###################### GRID section begins here ######################

          The grid is specified within brackets MAKE-ENDMAKE
MAKE      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
-- cartesian                           We select Cartesian gridding
--    grid   nx  ny  nz                option and specify the number of
      CART   10  10  3   /             grid blocks along every axis.
XYZBOUND
-- xmin-xmax   ymin-ymax   zmin-zmax    We specify the domain extent in feet.
```

```
   0   10000   0   10000   8325 8425 /  The reservoir is buried at depth
                                        8325 ft.

DZV                                     We define the thickness of 3 layers:
  20 30 50 /                            top layer thickness is 20 ft;
                                        middle layer thickness is 30 ft;
                                        bottom layer thickness is 50 ft.

GETWELLS                                We read wells specificactions from the
                                        SCHEDULE section.

DZWELL                                  We define that the wells are visible in
  100 10 /                              ParaView up to 100 feet above the top
                                        layer and up to 10 feet below the
                                        bottom layer.

ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

EQUALS
--  --------  --------                  We equal:
    PORO      0.3      /                - porosity to 0.3 in the whole domain;
    PERMX     500       4* 2*1 /        - X-perm. to 500 mD in the layer k=1;
    PERMX     50        4* 2*2 /        - X-perm. to 50 mD in the layer k=2;
    PERMX     200       4* 2*3 /        - X-perm. to 200 mD in the layer k=3.
--  --------  --------
/

COPY                                    We copy:
  PERMX PERMY /                         - X-perm. into Y-perm in the whole
/                                         domain.

<<<<<<<<<<<<<<<<<<<<<< We define Z-direction permeability <<<<<<<<<<<<<<<<<<<<<<

COPY                                    We copy:
  PERMX PERMZ /                         - X-perm. into Y-perm in the whole
/                                         domain.

EQUALS
--  --------  --------                  We equal:
    MULTZ     0.64      4* 2*1 /        - Z-perm. multiplyer to 0.64 between
--  --------  --------                    layers k=1 and k=2;
    MULTZ     0.265625  4* 2*2 /        - Z-perm. multiplyer to 0.64 between
--  --------  --------                    layers k=2 and k=3.
/

========= This keyword can be used insted of the two keywords above ============

--EQUALS                                We equal:
```

```
--    PERM  50   4* 2*1   4* 2*2 /      - perm. to 50 mD between layers k=1,2;
--    PERM  25   4* 2*2   4* 2*3 /      - perm. to 25 mD between layers k=2,3.
--/

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

PROPS     ######################## PROPS section begins here ####################

          Rock properties are specified within brackets ROCK-ENDROCK
ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

ROCKECL                                 We specify the reference pressure (psia)
-- -refpres- --compr--                  and the rock compressibility (1/psi).
      14.7     3.0D-6   /

ENDROCK   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The relative permeabilities are specified within brackets SAT-ENDSAT
SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

SWFN                                    We define the water relative
--  swat  krw                           permeability (krw) as a function of
    0.12  0       /                     water saturation (swat).
    1.0   0.00001 /
/

SGFN                                    We define the gas relative
--  sgas  krg                           permeability (krg) as a function of
    0     0     /                       gas saturation (sgas).
    0.02  0     /
    0.05  0.005 /
    0.12  0.025 /
    0.2   0.075 /
    0.25  0.125 /
    0.3   0.19  /
    0.4   0.41  /
    0.45  0.6   /
    0.5   0.72  /
    0.6   0.87  /
    0.7   0.94  /
    0.85  0.98  /
    1.0   1.0   /
/

SOF3                                    We define oil relative permeability
--  soil     krow     krog              for oil-water flows (krow) and oil
```

```
    0         0         0       /          relative permeability for oil-gas
    0.18      0         0       /          flows (krog) as a function of oil
    0.28      0.0001    0.0001  /          saturation (soil).
    0.38      0.001     0.001   /
    0.43      0.01      0.01    /
    0.48      0.021     0.021   /
    0.58      0.09      0.09    /
    0.63      0.2       0.2     /
    0.68      0.35      0.35    /
    0.76      0.7       0.7     /
    0.83      0.98      0.98    /
    0.86      0.997     0.997   /
    0.879     1         1       /
    0.88      1         1       /
/

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The fluid properties are specified within brackets EOS-ENDEOS
EOS       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

--OPTBO                                    $1$ Use these options if you want a
--  10*  10*1 /                            better fit to the SPE papaer

DENSITY                                    We specify the phases densities at
--   oil       water      gas              surface conditions (lb/cft).
    49.1      64.79     0.06054  /

PVTW                                       We specify water properties: reference
--  refpres   bwat    viswat  comprw       pressure (refpres), form. vol. factor at
    4014.7   1.029   3.13D-6   0.31  /     refpres (bwats), viscosity (viswat) and
                                           water compressibility (comprw).

PVDG
--   pgas   bgas    visgas                 We specify the properies of dry gas,
     14.7 166.666   0.008    /             i.e. the formation volume factor (bgas),
    264.7  12.093   0.0096   /             and viscosity (visgas) on pressure
    514.7   6.274   0.0112   /             (pgas).
   1014.7   3.197   0.014    /
   2014.7   1.614   0.0189   /
   2514.7   1.294   0.0208   /
   3014.7   1.080   0.0228   /
   4014.7   0.811   0.0268   /
   5014.7   0.649   0.0309   /
   9014.7   0.386   0.047    /
/

```

```
PVTO
--   rs      poil  boil  visoil         We specify the properties of live oil,
    0.001    14.7 1.062  1.04    /      i.e. the oil pressure (poil),
    0.0905  264.7 1.15   0.975   /      formation volume factor (boil),
    0.18    514.7 1.207  0.91    /      viscosity (visoil) on gas-oil ratio
    0.371  1014.7 1.295  0.83    /      of the solution (rs).
    0.636  2014.7 1.435  0.695   /
    0.775  2514.7 1.5    0.641   /
    0.93   3014.7 1.565  0.594   /
    1.270  4014.7 1.695  0.51
           5014.7 1.671  0.549
           9014.7 1.579  0.74    /
    1.618  5014.7 1.827  0.449
           9014.7 1.726  0.605   /
    2.984  9014.7 2.357  0.203   /      $1$ Comment out this line if you want
/                                       a better fit to the SPE paper.

ENDEOS    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

INIT      ######################## INIT section begins here #####################

          The initial equilibrium is specified within brackets EQL-ENDEQL
EQL       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

EQUIL
-- -------- -------- -------- -------- -------- --------
-- datum    datum     OWC     OWC      GOC      GOC
--   depth  pressure   depth   cappres   depth   cappres
-- -------- -------- -------- -------- -------- --------
     8400     4800     8500      1*       8200      1*    /
-- -------- -------- -------- -------- -------- --------

                   Using the keyword EQUIL we specify that
                   - pressure is 4800 (psi) at depth 8400 ft;
                   - oil-water contact is below the bottom layer;
                   - gas-oil contact is above the top layer.

RSVD
-- --depth- ---rs---
     8200     1.270  /                 We specify that initial gas-oil ratio
     8500     1.270  /                 (rs) is 1.270 (Mscf/stb).
-- -------- --------
/

ENDEQL    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

RPTSUM                                  We specify the properties which are
```

```
   PRESSURE SWAT SOIL SGAS RS /          saved at every report time (in the files
                                         SPE1.####.SUM (.vtu)).

RPTSUM                                   These properties are saved too at every
   I-IJKRES J-IJKRES K-IJKRES /          report step.

RPTWELL                                  We specify the wells parameters saved
   WBHP WGOR WOPR WGPR /                 at every report time (in the files
                                         SPE1.####.SUM (.vtu)).

SCHEDULE   ##################### SCHEDULE section begins here ####################

WEEKTOL                                  We enable option WEEKTOL (week
                                         convergence tolerances).
VARS
  PRES  DMAX 300   /                     Maximal pressure change in a grid block
  PRES  MAXV 10000 /                     is 300 psi. Maximal pressure is 10000
/                                        psi.

WELSPECS
-- -------- -- ---- ---- --------
-- well        i-   j-   datum          We specify two wells named PRODUCER and
--     name     ind  ind   depth        INJECTOR:
-- -------- -- ---- ---- --------
   PRODUCER 1*   10 10    8400    /     - PRODUCER well head is in  i,j=10,10;
   INJECTOR 1*    1  1    8335    /     - INJECTOR well head is in  i,j=1,1.
-- -------- -- ---- ---- --------
/

COMPDAT
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
-- well     i-   j-   k-   k-   mode satnum   explicit   well
--     name  ind  ind  min  max                 tran   diameter
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
   PRODUCER  10   10   3    3   OPEN  1*        1*        0.5    /
   INJECTOR   1    1   1    1   OPEN  1*        1*        0.5    /
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
/

                    Using the keyword COMPDAT we specify that:
                    - PRODUCER is completed in the layer k=3;
                    - INJECTOR is completed in the layer k=1;
                    - both wellbore diameter is 0.5 ft.

WCONHIST
   PRODUCER OPEN ORAT  20000 5* 1000 /
/
                    Using the keyword WCONHIST we specify that:
```

```
                    - PRODUCER target production rate is 20000 stb/day of oil;
                    - the minimal bottom-hole pressure is 1000 psi.

WCONINJE
   INJECTOR GAS OPEN GRAT 100000 /
/
                  Using the keyword WCONINJE we specify that:
                  - INJECTOR target injection rate is 100000 Mscf/day.

REPORTS                                 No reports in the LOG-file.
   NOTHING  /

TSTEP                                   We advance simulation to 10 years
  1 2 3 4 5 10 11.525 99*36.525 /       (3652.5 days)

POST      ######################## POST section begins here ######################

RPTPOST                                 We define the properties output from the
  TIMEY WBHP WGOR WOPR WGPR /           following POSTWELL keyword.
POSTWELL                                We save consolidated time series
   PRODUCER /                           report for PRODUCER.
/

RPTPOST                                 We define the properties output from the
  NOTHING TIMEY WBHP /                  following POSTWELL keyword.
POSTWELL                                We save consolidated time series
   INJECTOR /                           report for INJECTOR.
/

RPTPOST
   NOTHING TIMEY PRESSURE SGAS /        We save consolidated time series reports
POSTBLOC                                for the grid blocks (1,1,1) and
    1 1 1   /                           (10,10,3).
   10 10 3  /
/

RPTPOST
   NOTHING I-IJKRES PRESSURE SGAS TIMEY /
POSTSPEC
   1* LAYER1.CSV  86 /                  We save parameters in the grid blocks
POSTBLK                                 i=j in the layer k=1 at 8 years (report
    1  1  1 /                          step 86). The output file is LAYER1.CSV.
    2  2  1 /
    3  3  1 /
    4  4  1 /
    5  5  1 /
    6  6  1 /
    7  7  1 /
```

```
    8  8  1 /
    9  9  1 /
   10 10  1 /
/

POSTSPEC                                  We save parameters in the grid blocks
   1* LAYER2.CSV  86 /                    i=j in the layer k=1 at 8 years (report
POSTBLK                                   step 86). The output file is LAYER2.CSV.
    1  1  2 /
    2  2  2 /
    3  3  2 /
    4  4  2 /
    5  5  2 /
    6  6  2 /
    7  7  2 /
    8  8  2 /
    9  9  2 /
   10 10  2 /
/

CONVERT                                   We convert the output to ParaView
                                          compatible format.

END        ###############################################################################
```

9.3. 2nd SPE Comparative Study (BLACKOIL)

This problem is discussed in the paper: Weinstein H.G. et al. 1986 Second comparative solution project: A three-phase coning study. J. Pet. Tech. 38(3): 345-353. DOI:10.2118/10489-PA. See a description of the problem, input data files and visualisations under *the link*.

SPE2.RUN

```
################################################################################

                          2ND SPE COMPARATIVE STUDY

    The problem is from SPE-10489-PA paper:
    Weinstein H.G. et al. 1986 Second comparative solution project:
         A three-phase coning study. J. Pet. Tech. 38(3): 345-353.
         DOI:10.2118/10489-PA

    See description at: http://dx.doi.org/10.2118/10489-PA

--------------------------------------------------------------------------------

    Last update on 17.05.2015
    Program Version: 2015.C08

################################################################################

--------------------------------------------------------------------------------

   Note:
   We type all comments in lowercase letters while all keywords and mnemonics
      must be in uppercase letters.

--------------------------------------------------------------------------------

   Note:
-- any line beginning with '!' or '--' is a comment line;
   any line not beginning with an uppercase letter outside keyword
!  instruction is also a comment line.

--------------------------------------------------------------------------------

   Note:
   Any data line must be terminated by the slash '/' sign.

--------------------------------------------------------------------------------

```

```
   Note:
   We denote by the repeated symbols the following structural elements of this
      RUN-file:
      '###############' - delimits the sections of the RUN-file.
      '<<<<<<<<<<<<<<' - designates an opening bracket.
      '>>>>>>>>>>>>>>' - designates a closing bracket.
      '^^^^^^^^^^^^^^' - designates the keyword, which affects the order of the
                         data input for all the following keywords (both in the
                         current and in the following sections).
      '==============' - merge the keywords and the comments in paragraphs.

--------------------------------------------------------------------------------



RUNSPEC   ################### RUNSPEC section begins here ######################

BLACKOIL  ^^^^^^ We switch the program to the BLACKOIL EOS module ^^^^^^^^^^^^^^

                                       We enable:
WATER                                  - water phase;
OIL                                    - oil phase;
GAS                                    - gas phase;
DISGAS                                 - gas dissolution in oil phase.

FIELD     ^^^^^^^^^^^^ We use FIELD units ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
                                       Note, this keywords affects the units
                                       in many keywords below.

ROCKCOMP                               We enable rock compaction.

CAPPRES                                We enable capillary pressure.

TANKPRES                               We set the default stock tank pressure
  400 /                                associated with the well to 400 psia,
                                       which is the minimal pressure in the
                                       PVT tables below. By default this
                                       pressure is 14.7 psia, out of range.
                                       Actually this keyword can be commented
                                       out, in this case simulator will
                                       extrapolate PVT properties to 14.7 psia.

OPTIONS                                Piecewise linear interpolation
  53* 1 /                              of relative permeabilities. Use this
                                       option if you want a better fit to the
                                       SPE-9723 paper.

GRID      ###################### GRID section begins here ######################
```

```

         The grid is specified within brackets MAKE-ENDMAKE
MAKE      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
-- cartesian                           We select Radial gridding
--    grid   nr ntheta  nz             option and specify the number of
     RADIAL  10   1     15   /         grid blocks along every direction.

RTZBOUND
-- rmin-rmax           tmin-tmax        zmin-zmax
   0.25 2050.0   -0.0314159 0.0314159    9000 10000 /
                                       We specify the domain extent in feet
                                       along R and Z axes, and in radians
                                       along Theta direction. Thus, we
                                       consider the 0.01 fraction of the
                                       full circle. The reservoir is buried
                                       at depth 9000 feet.

DRV
  1.75   2.32   5.01  10.84   23.39
 50.55 109.21 235.92 509.68 1101.08 /   We define the grid blocks extensions
                                       along R axis (ft).

DZV
  20 15 26  15 16
  14  8  8  18 12
  19 18 20  50 100 /                   We define the thickness of 15 layers
                                       (from top to bottom).

WELSPECS
-- -------- -- ---- ---- --------      We specify a well named PROD1 with
-- well        i-   j-   datum         the well head at i=1, j=1. The datum
--     name        ind  ind   depth       depth for bottom-hole pressure is
-- -------- -- ---- ---- --------      9110 feet.
    PROD1   1*   1    1   9110    /
-- -------- -- ---- ---- --------
/

COMPDAT
-- -------- ---- ---- ---- ---- ---- -------- --------    We specify that the
-- well     i-   j-   k-   k-   mode satnum   explicit    well is completed in
--     name  ind  ind  min  max                  tran      layers k=7 and k=8.
-- -------- ---- ---- ---- ---- ---- -------- --------    We explicitly define
    PROD1    1*   1*    7    7  OPEN  1*      27.228   /  the completions
    PROD1    1*   1*    8    8  OPEN  1*      2.1079   /  transmissibilities.
-- -------- ---- ---- ---- ---- ---- -------- --------
/

DZWELL                                 We define that the well is visible in
```

```
  100 25 /                              ParaView up to 100 feet above the top
                                        layer and up to 25 feet below the
                                        bottom layer.

ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

EQUALS
  PERMI   35       4*  2*1  / layer 1   We define laver by layer i-index
  PORO    0.087            /           direction permeability (along R-axis)
  PERMI   47.5     4*  2*2  / layer 2   and porosity.
  PORO    0.097            /
  PERMI   148      4*  2*3  / layer 3
  PORO    0.111            /
  PERMI   202      4*  2*4  / layer 4
  PORO    0.160            /
  PERMI   90       4*  2*5  / layer 5
  PORO    0.130            /
  PERMI   418.5    4*  2*6  / layer 6
  PORO    0.170            /
  PERMI   775      4*  2*7  / layer 7
  PORO    0.170            /
  PERMI   60       4*  2*8  / layer 8
  PORO    0.080            /
  PERMI   682      4*  2*9  / layer 9
  PORO    0.140            /
  PERMI   472      4*  2*10 / layer 10
  PORO    0.130            /
  PERMI   125      4*  2*11 / layer 11
  PORO    0.120            /
  PERMI   300      4*  2*12 / layer 12
  PORO    0.105            /
  PERMI   137.5    4*  2*13 / layer 13
  PORO    0.120            /
  PERMI   191      4*  2*14 / layer 14
  PORO    0.116            /
  PERMI   350      4*  2*15 / layer 15
  PORO    0.157            /
/

COPY                                    We copy i-index direction permeability
  PERMI PERMZ /                         into Z-axis permeability.
/

MULTIPLY                                We multiply Z direction permeability by
  PERMZ 0.1 /                           the factor 0.1.
/

MULTIPLY                                Because we specified grid only for
```

```
   CELLVOL   100.0 /                    0.01 fraction of the full circle we
   AREA      100.0 /                    multiply cell volumes and interface
/                                       areas (both in R and Z directions)
                                        by the factor 100.

RPTGRID                                 We report from the GRID section
  PORO PERMI PERMZ CELLVOL  /           porosity, permeabilities and grid block
                                        volumes.

PROPS     ###################### PROPS section begins here ####################

          Rock properties are specified within brackets ROCK-ENDROCK
ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

ROCKECL                                 We specify the reference pressure (psia)
-- -refpres- --compr--                  and the rock compressibility (1/psi).
      3600     4.0D-6   /

ENDROCK   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The relative permeabilities are specified within brackets SAT-ENDSAT
SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

SWFN
--  swat  krw    pcow
    0.22  0       7   /                 We define the water relative (krw)
    0.3   0.07    4   /                 and oil-water capillary pressure
    0.4   0.15    3   /                 (pcow, psia) as tabulated functions
    0.5   0.24    2.5 /                 of water saturation (swat).
    0.6   0.33    2   /
    0.8   0.65    1   /
    0.9   0.83    0.5 /
    1     1       0   /
/

SGFN
--  sgas    krg    pcgo
    .0000  .0000  .0000 /               We define gas relative permeability
    .0400  .0000  .2000 /               (krg) and gas-oil capillary pressure
    .1000  .0220  .5000 /               (pcgo, psia) as tabulated functions of
    .2000  .1000 1.0000 /               gas saturation (sgas).
    .3000  .2400 1.5000 /
    .4000  .3400 2.0000 /
    .5000  .4200 2.5000 /
    .6000  .5000 3.0000 /
    .7000  .8125 3.5000 /
```

```
    .7800 1.0000 3.9000 /
/

SOF3
--  soil     krow      krog                 We define oil relative permeability
    0        0         0       /            for oil-water flows (krow) and oil
    0.2      0         0       /            relative permeability for oil-gas
    0.38     0.00432   0       /            flows (krog) as a function of oil
    0.4      0.0048    0.004   /            saturation (soil).
    0.48     0.05288   0.02    /
    0.5      0.0649    0.036   /
    0.58     0.11298   0.1     /
    0.6      0.125     0.146   /
    0.68     0.345     0.33    /
    0.7      0.4       0.42    /
    0.74     0.7       0.6     /
    0.78     1         1       /
/

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The fluid properties are specified within brackets EOS-ENDEOS
EOS       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

--OPTBO
--  10* 10*1

DENSITY                                     We specify the phases densities at
--   oil       water      gas               surface conditions (lb/cft).
     45        63.02     0.0702  /

PVTW                                        We specify water properties: reference
--  refpres   bwat    viswat  comprw        pressure (refpres), form. vol. factor at
    3600    1.00341   3.0D-6   0.96  /      refpres (bwats), viscosity (viswat) and
                                            water compressibility (comprw).

PVDG
--   pgas    bgas  visgas                   We specify the properies of dry gas,
      400    5.9   0.013  /                 i.e. the formation volume factor (bgas),
      800    2.95  0.0135 /                 and viscosity (visgas) on pressure
     1200    1.96  0.014  /                 (pgas).
     1600    1.47  0.0145 /
     2000    1.18  0.015  /
     2400    0.98  0.0155 /
     2800    0.84  0.016  /
     3200    0.74  0.0165 /
     3600    0.65  0.017  /
```

```
     4000    0.59  0.0175 /
     4400    0.54  0.018  /
     4800    0.49  0.0185 /
     5200    0.45  0.019  /
     5600    0.42  0.0195 /
/

PVTO
--   rs      poil   boil    visoil      We specify the properties of live oil,
   0.165      400   1.012    1.17  /    i.e. the oil pressure (poil),
   0.335      800   1.0255   1.14  /    formation volume factor (boil),
   0.500     1200   1.038    1.11  /    viscosity (visoil) on gas-oil ratio
   0.665     1600   1.051    1.08  /    of the solution (rs).
   0.828     2000   1.063    1.06  /
   0.985     2400   1.075    1.03  /
   1.130     2800   1.087    1.00  /
   1.270     3200   1.0985   0.98  /
   1.390     3600   1.11     0.95  /
   1.500     4000   1.12     0.94  /
   1.600     4400   1.13     0.92  /
   1.676     4800   1.14     0.91  /
   1.750     5200   1.148    0.9   /
   1.810     5600   1.155    0.89
             6000   1.1504   0.89
             6400   1.1458   0.89
             6800   1.1412   0.89
             7200   1.1367   0.89  /
/

ENDEOS    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

RPTSATTA                                 We report saturation functions for
  1* 'SPE2.WAT-OIL.CSV' 1000 0 1   1 /  visualization in ParaView.
RPTSATTA
  1* 'SPE2.OIL-GAS.CSV' 1000 0 1   2 /

INIT      ######################## INIT section begins here #####################

          The initial equilibrium is specified within brackets EQL-ENDEQL
EQL       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

EQUIL
-- -------- -------- -------- -------- -------- --------
-- datum    datum     OWC     OWC      GOC      GOC
--   depth  pressure   depth   cappres   depth   cappres
-- -------- -------- -------- -------- -------- --------
     9035     3600     9209      1*       9035      1*    /
```

```
-- -------- -------- -------- -------- -------- --------
/

                    Using the keyword EQUIL we specify that
                    - pressure is 3600 (psi) at depth 9035 ft;
                    - oil-water contact is at depth 9209;
                    - gas-oil contact is at depth 9035.

ENDEQL    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

RPTSUM                                  We specify the properties which are
   PRESSURE SWAT SOIL SGAS RS /         saved at every report time (in the files
                                        SPE2.####.SUM (.vtu)).

RPTSUM                                  These properties are saved too at every
   I-IJKRES J-IJKRES K-IJKRES /         report step.

RPTWELL                                 We specify the wells parameters saved
  WBHP WGOR WOPR WGPR WWPR WWCT /       at every report time (in the files
                                        SPE2.####.SUM (.vtu)).

SCHEDULE   #####################################################################

WEEKTOL                                 We enable the option WEEKTOL (week
                                        convergence tolerances).

REPORTS                                 No reports in the LOG-file.
   NOTHING  /

WELLPROD
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
--   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
--                        rate     rate           rate             dimens
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
   PROD1    OPEN ORAT   1000    1*         3000     5E4       OIL  /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
/
                                        Using the keyword WELLPROD we
                                        specify that the well is operating
                                        at constunt oil production rate
                                        1000 stb/day with minimal bottom-
                                        hole pressure at 3000 psia.

TSTEP                                   We advance simulation to 10 days.
   0.1 0.9 2 3 4 /

WELLPROD
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
```

```
--   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
--                         rate     rate            rate              dimens
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
   PROD1    OPEN ORAT   100    1*       3000        5E4        OIL /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
/
                                       Using the keyword WELLPROD we
                                       specify that the well is operating
                                       at constunt oil production rate
                                       100 stb/day with minimal bottom-
                                       hole pressure at 3000 psia.

TSTEP                                  We advance simulation to 50 days.
  0.1 9.9 3*10 /

WELLPROD
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
--   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
--                         rate     rate            rate              dimens
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
   PROD1    OPEN ORAT   1000    1*        3000     5E4       OIL  /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
/
                                       Using the keyword WELLPROD we
                                       specify that the well is operating
                                       at constunt oil production rate
                                       1000 stb/day with minimal bottom-
                                       hole pressure at 3000 psia.


TSTEP                                  We advance simulation to 720 days.
 0.1 9.9 66*10 /

WELLPROD
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
--   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
--                         rate     rate            rate              dimens
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
   PROD1    OPEN ORAT   100    1*       3000     5E4         OIL /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
/
                                       Using the keyword WELLPROD we
                                       specify that the well is operating
                                       at constunt oil production rate
                                       100 stb/day with minimal bottom-
                                       hole pressure at 3000 psia.

TSTEP                                  We advance simulation to 900 days.
```

```
 0.1 9.9 17*10 /

POST      ######################## POST section begins here ######################

CONVERT                                  We convert the output to ParaView
                                         compatible format.

RPTPOST
  TIME WBHP WGOR WOPR WGPR WWPR WWCT /  We save consolidated time series
POSTWELL                                 report for the production well PROD1.
   PROD1 /                               The properties to be reported are
/                                        TIME, WBHP, WGOR etc.

RPTPOST                                  We save consolidated time series for
  NOTHING TIME PRESSURE /                the grid block (1,1,7). The properties
POSTBLOC                                 to be reported are TIME and PRESSURE.
   1 1 7 /
/

END        ###############################################################################
```

9.4. 6th SPE Comparative Study; Water injection (BLACKOIL)

This problem is discussed in the paper: Firoozabadi, A. and Thomas, L.K. 1990 Sixth SPE Comparative solution project: Dual-Porosity simulators. J. Pet. Tech. 43(6): 710-764. DOI: 10.2118/18741-PA.. See a description of the problem, input data files and visualisations under *the link*.

SPE6-WATINJO.RUN

```
################################################################################

                          6TH SPE COMPARATIVE STUDY
                (WATER INJECTION; SIMPLIFIED PROBLEM STATEMENT)

    The problem is from SPE-18741-PA paper:
    Firoozabadi, A. and Thomas, L.K. 1990 Sixth SPE Comparative solution
       project: Dual-Porosity simulators. J. Pet. Tech. 43(6). 710-764.
       DOI: 10.2118/18741-PA.

    See description at: http://dx.doi.org/10.2118/18741-PA

    We consider the Water Injection scenario with simplified relative
    permeability data (the orientational or hysteresis effects are
    not simulated).

--------------------------------------------------------------------------------

    Last update on 16.05.2017
    Program Version: 2017.A01

################################################################################

--------------------------------------------------------------------------------

   Note:
   We type all comments in lowercase letters while all keywords and mnemonics
      must be in uppercase letters.

--------------------------------------------------------------------------------

   Note:
-- any line beginning with '!' or '--' is a comment line;
   any line not beginning with an uppercase letter outside keyword
!  instruction is also a comment line.

```

```
--------------------------------------------------------------------------------

   Note:
   Any data line must be terminated by the slash '/' sign.

--------------------------------------------------------------------------------

   Note:
   We denote by the repeated symbols the following structural elements of this
      RUN-file:
      '###############' - delimits the sections of the RUN-file.
      '<<<<<<<<<<<<<<' - designates an opening bracket.
      '>>>>>>>>>>>>>>' - designates a closing bracket.
      '^^^^^^^^^^^^^^' - designates the keyword, which affects the order of the
                         data input for all the following keywords (both in the
                         current and in the following sections).
      '==============' - merge the keywords and the comments in paragraphs.

--------------------------------------------------------------------------------

RUNSPEC   ################### RUNSPEC section begins here ######################

BLACKOIL  ^^^^^^ We switch the program to the BLACKOIL EOS module ^^^^^^^^^^^^^^

                                       We enable:
WATER                                  - water phase;
OIL                                    - oil phase;
GAS                                    - gas phase;
DISGAS                                 - gas dissolution in oil.

                                       We enable
CAPPRES                                - capillary pressure;
ROCKCOMP                               - rock compressibility.

DUALPORO                               We activate the double porosity option
GRAVDR                                 and activate the gravity drainage
                                       option.

FIELD     ^^^^^^^^^^^^^^^^^^^^^ We use FIELD units ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
                                       Note, this keywords affects the units
                                       in many keywords below.

FAST                                   We activate the FAST option in order to
                                       accelerate the simulation run.

GRID      ##################### GRID section begins here ######################

          The grid is specified within brackets MAKE-ENDMAKE
```

```
MAKE     <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
-- cartesian                                We select Cartesian gridding
--    grid   nx  ny  nz                     option and specify the number of
      CART   10   1   5  /                  grid blocks along every axis. This is a
                                            2D problem because ny=1.

XYZBOUND                                    We specify that xmin=0 ft,
-- xmin-xmax  ymin-ymax  zmin-zmax          ymin=0 ft, the top surface of the
   0.0  10.     0.0 10.  3000 3010. /       reservoir is at 3000 feet depth. The
                                            number 3010 is redefined by the
                                            following keyword DZV.

DXV                                         We specify grid blocks extension
  10*200 /                                  along X axis.
DYV                                         We specify grid blocks extension
  1000 /                                    along Y axis.
DZV                                         We specify grid blocks extension
  5*50 /                                    along Z axis.

WELSPECS
-- -------- -- ---- ---- --------
-- well        i-   j-   datum              We specify two wells named PRODUCER and
--     name     ind  ind   depth            INJECTOR:
-- -------- -- ---- ---- --------
   PRODUCER 1*   10  1    1*    /           - PRODUCER well head is in i=10, j=1;
   INJECTOR 1*    1  1    1*    /           - INJECTOR well head is in i=1, j=1.
-- -------- -- ---- ---- --------
/

COMPDAT
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
-- well     i-   j-   k-   k-   mode satnum   explicit   well
--     name  ind  ind  min  max                  tran   diameter
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
   PRODUCER   1*   1*  6   8   OPEN  1*        2.0      /
   INJECTOR   1*   1*  6   10  OPEN  1*        2.0      /
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
/                                           The producer is completed in the
                                            layers 1,2 and 3 in the fractures
                                            media.
                                            The INJECTOR is completed in the
                                            in the layers 1 to 5 in the fractures
                                            media.

DZWELL                                      We specify wells visualization
  150 50 /                                  parameters. This data does not
                                            influence simulation results.
```

```
ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

EQUALS
-- -------- -------- -- --------        We equate porosity
   PORO     0.29     4*  1  5   /       - to 0.29 in matrix;
   PORO     0.01     4*  6  10  /       - to 0.01 in fractures.
-- -------- -------- -- --------
/

EQUALS
-- -------- -------- -- --------        We equate X permeability
   PERMX    1        4*  1  5   /       - to 1 mD in matrix;
   PERMX    10       4*  6  7   /       - to 10 mD in top two fracture layers;
   PERMX    90       4*  8  8   /       - to 90 mD in the 3rd fracture layer;
   PERMX    20       4*  9  10  /       - to 20 mD in the 4th and 5th layers.
-- -------- -------- -- --------
/

COPY
-- -------- --------                    We copy
   PERMX    PERMY   /                   - X permeability to Y permeability;
   PERMX    PERMZ   /                   - X permeability to Z permeability;
-- -------- --------                    both in matrix and fractures.
/

MULTIPLY
-- -------- --------                    We multiply Z permeability by 0.1.
   PERMZ    0.1     /
-- -------- --------
/

EQUALS
-- -------- -------- -- --------        We equate
   SIGMAV   0.040    4*  1  2   /       - sigma-factor to 0.04 and matrix blocks
   DZMTRXV  25                  /         height to 25 feet in layers 1 and 2;
   SIGMAV   1.000    4*  3  3   /       - sigma-factor to 1.0 and matrix blocks
   DZMTRXV  5                   /         height to 5 feet in layer 3;
   SIGMAV   0.250    4*  4  5   /       - sigma-factor to 0.25 and matrix blocks
   DZMTRXV  10                  /         height to 10 feet in layers 4 and 5.
-- -------- -------- -- --------
/

RPTGRID
 PORO PERMX PERMY PERMZ SIGMAV
 DZMTRXV SATNUM K-IJKRES DEPTH /        We report these properties from the
                                        grid section.

PROPS     ####################### PROPS section begins here ####################
```

```

          Rock properties are specified within brackets ROCK-ENDROCK
ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

ROCKECL                                 We specify the reference pressure (psia)
-- -refpres-     --compr--              and the rock compressibility (1/psi).
    6000.00      0.35E-05 /

ENDROCK   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

EQUALS
-- -------- -------- -- --------        We introduce two regions for saturation
   SATNUM   1        4*  1 5    /       functions: SATNUM=1 in matrix media,
   SATNUM   2        4*  6 10   /       and SATNUM=2 in fractures media.
-- -------- -------- -- --------
/

          The saturation functions are specified within brackets SAT-ENDSAT
SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
  1 /                                   Here, we input saturation functions
                                        in the matrix media.

SWOF
-- swat      krw      krow      pcwo    For two-phase water-oil system, we
   0.2000  0.0000    1.000    1.0000 /  specify water relative permeability
   0.2500  0.0050    0.860    0.5000 /  (krw), oil releative permeability
   0.3000  0.0100    0.723    0.3000 /  (krow), water-oil capillary pressure
   0.3500  0.0200    0.600    0.1500 /  (pcwo) as tabulated functions of
   0.4000  0.0300    0.492    0.0000 /  water saturation (swat).
   0.4500  0.0450    0.392   -0.2000 /
   0.5000  0.0600    0.304   -1.2000 /
   0.6000  0.1100    0.154   -4.0000 /
   0.7000  0.1800    0.042  -10.0000 /
   0.7500  0.2300    0.0    -40.0000 /
/

SGOF
-- sgas     krg     krog  pcgo          For two-phase gas-oil system, we
   0.0000  0.0000   1.00  0.075 /       specify gas relative permeability (krg),
   0.1000  0.0150   0.70  0.085 /       oil relative permeability (krog), gas-
   0.2000  0.0500   0.45  0.095 /       oil capillary pressure (pcgo) as
   0.3000  0.1030   0.25  0.115 /       tabulated functions of gas saturation
   0.4000  0.1900   0.11  0.145 /       (sgas).
   0.5000  0.3100   0.028 0.255 /
   0.5500  0.4200   0.0   0.386 /
/

```

```
ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The saturation functions are specified within brackets SAT-ENDSAT
SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
  2 /                                         Here, we input saturation functions in
                                              the fractures media. The relative
                                              permeabilities are linear functions
                                              and capillary pressure is zero.

SWOF
-- swat  krw  krow  pcwo
   0.0   0.0  1.0   0.0  /
   1.0   1.0  0.0   0.0  /
/

SGOF
-- sgas  krg  krog  pcgo
   0.0   0.0  1.0   0.0  /
   1.0   1.0  0.0   0.0  /
/

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The fluid properties are specified within brackets EOS-ENDEOS
EOS       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

OPTBO                                       Use these options if you want a
  10*  10*1 /                               better fit to the SPE paper, otherwise
                                            you can comment it.

DENSITY                                       We specify the phases densities at
--   oil       water      gas                 surface conditions (lb/cft).
   51.1400  65.0000     0.05800 /

PVTW                                          We specify water properties: reference
--  refpres   bwat   comprw   viswat          pressure (refpres), form. vol. factor at
    0.0     1.07000  0.35E-05  0.35  /        refpres (bwat),  water compressibility
                                              (comprw) and viscosity (viswat).

PVDG
--   pgas      bgas      visgas               We specify the properies of dry gas,
   1674.00    1.98000    0.01620 /            i.e. the formation volume factor (bgas),
   2031.00    1.62000    0.01710 /            and viscosity (visgas) on pressure
   2530.00    1.30000    0.01840 /            (pgas).
   2991.00    1.11000    0.01970 /
   3553.00    0.95900    0.02130 /
   4110.00    0.85500    0.02300 /
```

```
   4544.00    0.79500    0.02440 /
   4935.00    0.75100    0.02550 /
   5255.00    0.72000    0.02650 /
   5545.00    0.69600    0.02740 /
   7000.00    0.60000    0.03300 /
/

PVTO
--   rs      poil   boil    visoil      We specify the properties of live oil,
  0.36700 1674.00 1.30010  0.52900 /    i.e. the oil pressure (poil),
  0.44700 2031.00 1.33590  0.48700 /    formation volume factor (boil),
  0.56400 2530.00 1.38910  0.43600 /    viscosity (visoil) on gas-oil ratio
  0.67900 2991.00 1.44250  0.39700 /    of the solution (rs).
  0.83200 3553.00 1.51410  0.35100 /
  1.00000 4110.00 1.59380  0.31000 /
  1.14300 4544.00 1.66300  0.27800 /
  1.28500 4935.00 1.73150  0.24800 /
  1.41300 5255.00 1.79530  0.22900 /
  1.53000 5545.00 1.85400  0.21000
          5600.00 1.85330  0.21090
          5700.00 1.85210  0.21260
          5800.00 1.85090  0.21430
          5900.00 1.84970  0.21610
          6000.00 1.84850  0.21780
          6100.00 1.84730  0.21950
          6200.00 1.84610  0.22120 /
  2.25900 7000.00 2.19780  0.10900
          7100.00 2.19660  0.11070
          7200.00 2.19540  0.11240
          7300.00 2.19420  0.11410 /
/

ENDEOS    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

INIT      ######################## INIT section begins here ######################

          The initial equilibrium is specified within brackets EQL-ENDEQL
EQL       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

EQUIL
-- -------- -------- -------- -------- -------- --------
-- datum    datum     OWC      OWC      GOC      GOC
--   depth  pressure   depth    cappres   depth   cappres
-- -------- -------- -------- -------- -------- --------
     3025    6000      3300   1*         2900     1*     /
-- -------- -------- -------- -------- -------- --------

```

```
                    Using the keyword EQUIL we specify that
                    - pressure is 6000 psi at depth 3025 ft;
                    - oil-water contact is at depth 3300 ft (thus, at the
                      initial moment of time there is no mobile water because
                      3300 m is below the bottom layer);
                    - gas-oil contact is at depth 2900 ft (thus, at the
                      initial moment of time there is no free gas because
                      2900 m is above the top layer).

RSVD
-- depth  rs                            The initial gas-oil ratio is
   2900  1.53 /                         1.53 Mscf/stb.
   3300  1.53 /
/

ENDEQL    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

RPTSUM                                  At every report step, we report pressure
   PRES SWAT SOIL SGAS /                and saturations in the summary files.

RPTWELL
  WBHP WWIR WWIT WLPR WOPR
  WGPR WWPR WWCT WOPT WWPT WGPT /       For every well we report these
                                        properties.

SCHEDULE   ##################### SCHEDULE section begins here ####################

VARS                                    We increase the maximum allowed pressure
  PRES  MAXV  7000 /                    in the simulation up to 7000 psi.
/

REPORTS                                 No convergence or other reports in the
   NOTHING /                            LOG file.

WELLPROD
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
--   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
--                        rate     rate            rate              dimens
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
   PRODUCER OPEN LRAT   1000      1*       100     1E6 /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
/                                       We specify that the PRODUCER produces
                                        1000 stb/day of liquid (oil+water) at
                                        minimum bottom-hole pressure 100 psi.

WELLINJE
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
--   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
```

```
--                        rate     rate            rate              dimens
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
   INJECTOR OPEN WRAT   1750      1*       6100     1E6     WATER /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
/                                    We inject 1750 stb/day of water at
                                     maximum bottom-hole pressure 6100 psi.

TUNING                               The maximum time step is 25 days.
  1* 25 /

TSTEP                                We simulate the flow up to 14610 days
 200*73.05 /                         reporting results every 73.05 days.

POST      ######################## POST section begins here ######################

CONVERT                              We convert the output to ParaView
                                     compatible format.

RPTPOST                              We save consolidated time series
  TIME WBHP WWIR WWIT /              report for the INJECTOR well.
POSTWELL                             The properties to be reported are
  INJECTOR /                         TIME, WBHP, WWIR, WWIT.
/

RPTPOST
  NOTHING TIME WBHP WWCT WLPR
  WOPR WGPR WWPR /                   We save consolidated time series
POSTWELL                             report for the PRODUCER well.
  PRODUCER /
/

END        ###########################################################################
```

9.5. 6th SPE Comparative Study; Depletion (BLACKOIL)

This problem is discussed in the paper: Firoozabadi, A. and Thomas, L.K. 1990 Sixth SPE Comparative solution project: Dual-Porosity simulators. J. Pet. Tech. 43(6): 710-764. DOI: 10.2118/18741-PA.. See a description of the problem, input data files and visualisations under *the link*.

SPE6-DEPLETION0.RUN

```
################################################################################

                          6TH SPE COMPARATIVE STUDY
                  (DEPLETION; SIMPLIFIED PROBLEM STATEMENT)

    The problem is from SPE-18741-PA paper:
    Firoozabadi, A. and Thomas, L.K. 1990 Sixth SPE Comparative solution
       project: Dual-Porosity simulators. J. Pet. Tech. 43(6). 710-764.
       DOI: 10.2118/18741-PA.

    See description at: http://dx.doi.org/10.2118/18741-PA

    We consider the Depletion scenario with simplified relative
    permeability data (the orientational or hysteresis effects are
    not simulated). We also use a complicated production schedule for
    demonstration purposes (stepwise decreasing bottom-hole pressure).

--------------------------------------------------------------------------------

    Last update on 16.05.2017
    Program Version: 2017.A01

################################################################################


RUNSPEC   ################### RUNSPEC section begins here ######################

BLACKOIL  ^^^^^^ We switch the program to the BLACKOIL EOS module ^^^^^^^^^^^^^^

                                       We enable:
WATER                                  - water phase;
OIL                                    - oil phase;
GAS                                    - gas phase;
DISGAS                                 - gas dissolution in oil.

                                       We enable
```

```
CAPPRES                                 - capillary pressure;
SURFTENS                                - surface tension dependence on
                                          pressure;
ROCKCOMP                                - rock compressibility.

DUALPORO                                We activate the double porosity option
GRAVDR                                  and activate the gravity drainage
                                        option.

FIELD     ^^^^^^^^^^^^^^^^^^^^^^ We use FIELD units ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
                                        Note, this keywords affects the units
                                        in many keywords below.

FAST                                    We activate the FAST option in order to
                                        accelerate the simulation run.

OPTIONS                                 Use these options for better fit to
  52* 1 1 /                                    the SPE paper (piecewise linear
                                        saturation functions), otherwise comment
                                        this keyword.

GRID      ###################### GRID section begins here ######################

          The grid is specified within brackets MAKE-ENDMAKE
MAKE      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
-- cartesian                            We select Cartesian gridding
--    grid   nx  ny  nz                  option and specify the number of
      CART   10   1   5  /               grid blocks along every axis. This is a
                                        2D problem because ny=1.

XYZBOUND                                We specify that xmin=0 meters,
-- xmin-xmax  ymin-ymax  zmin-zmax       ymin=0 meters, the top surface of the
   0.0  10.     0.0 10.  3000 3010. /   reservoir is at 3000 meters depth. The
                                        number 3010 is redefined by the
                                        following keyword DZV.

DXV                                     We specify grid blocks extension
  10*200 /                              along X axis.
DYV                                     We specify grid blocks extension
  1000 /                                along Y axis.
DZV                                     We specify grid blocks extension
  5*50 /                                along Z axis.

WELSPECS
-- -------- -- ---- ---- --------
-- well        i-   j-   datum          We specify well PRODUCER with
--     name     ind  ind   depth        well head in the i=10, j=1
-- -------- -- ---- ---- --------       grid block.
```

```
   PRODUCER 1*   10  1    1*    /
-- -------- -- ---- ---- --------
/

COMPDAT
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
-- well     i-   j-   k-   k-   mode satnum   explicit   well
--     name  ind  ind  min  max                 tran   diameter
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
   PRODUCER   1*   1*  10  10   OPEN  1*        2.0      /
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
/                                    The PRODUCER is completed in the
                                     layer 5 in the fractures media.

DZWELL                               We specify wells visualization
  150 50 /                           parameters. This data does not
                                     influence simulation results.

ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

EQUALS
-- -------- -------- -- --------     We equate porosity
   PORO     0.29     4*  1  5   /    - to 0.29 in matrix;
   PORO     0.01     4*  6  10  /    - to 0.01 in fractures.
-- -------- -------- -- --------
/

EQUALS
-- -------- -------- -- --------     We equate X permeability
   PERMX    1        4*  1  5   /    - to 1 mD in matrix;
   PERMX    10       4*  6  7   /    - to 10 mD in top two fracture layers;
   PERMX    90       4*  8  8   /    - to 90 mD in the 3rd fracture layer;
   PERMX    20       4*  9  10  /    - to 20 mD in the 4th and 5th layers.
-- -------- -------- -- --------
/

COPY
-- -------- --------                 We copy
   PERMX    PERMY   /                - X permeability to Y permeability;
   PERMX    PERMZ   /                - X permeability to Z permeability;
-- -------- --------                 both in matrix and fractures.
/

MULTIPLY
-- -------- --------                 We multiply Z permeability by 0.1.
   PERMZ    0.1     /
-- -------- --------
/
```

```
133
134 EQUALS
135 -- -------- -------- -- --------        We equate
136    SIGMAV   0.040    4*  1  2   /       - sigma-factor to 0.04 and matrix blocks
137    DZMTRXV  25                  /         height to 25 feet in layers 1 and 2;
138    SIGMAV   1.000    4*  3  3   /       - sigma-factor to 1.0 and matrix blocks
139    DZMTRXV  5                   /         height to 5 feet in layer 3;
140    SIGMAV   0.250    4*  4  5   /       - sigma-factor to 0.25 and matrix blocks
141    DZMTRXV  10                  /         height to 10 feet in layers 4 and 5.
142 -- -------- -------- -- --------
143 /
144
145 RPTGRID
146  PORO PERMX PERMY PERMZ SIGMAV
147  DZMTRXV SATNUM K-IJKRES DEPTH /        We report these properties from the
148                                         grid section.
149
150 PROPS     ######################## PROPS section begins here ####################
151
152           Rock properties are specified within brackets ROCK-ENDROCK
153 ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
154    /
155
156 ROCKECL                                 We specify the reference pressure (psia)
157 -- -refpres-     --compr--              and the rock compressibility (1/psi).
158     6000.00      0.35E-05 /
159
160 ENDROCK   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
161
162 EQUALS
163 -- -------- -------- -- --------        We introduce two regions for saturation
164    SATNUM   1        4*  1 5    /       functions: SATNUM=1 in matrix media,
165    SATNUM   2        4*  6 10   /       and SATNUM=2 in fractures media.
166 -- -------- -------- -- --------
167 /
168
169           The saturation functions are specified within brackets SAT-ENDSAT
170 SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
171   1 /                                   Here, we input saturation functions
172                                         in the matrix media.
173
174 SWOF
175 -- swat      krw      krow      pcwo    For two-phase water-oil system, we
176    0.2000  0.0000    1.000    1.0000 /  specify water relative permeability
177    0.2500  0.0050    0.860    0.5000 /  (krw), oil releative permeability
178    0.3000  0.0100    0.723    0.3000 /  (krow), water-oil capillary pressure
179    0.3500  0.0200    0.600    0.1500 /  (pcwo) as tabulated functions of
180    0.4000  0.0300    0.492    0.0000 /  water saturation (swat).
```

```
   0.4500  0.0450    0.392   -0.2000 /
   0.5000  0.0600    0.304   -1.2000 /
   0.6000  0.1100    0.154   -4.0000 /
   0.7000  0.1800    0.042  -10.0000 /
   0.7500  0.2300    0.0    -40.0000 /
/

SGOF
-- sgas     krg     krog  pcgo          For two-phase gas-oil system, we
   0.0000  0.0000   1.00  0.075 /       specify gas relative permeability (krg),
   0.1000  0.0150   0.70  0.085 /       oil relative permeability (krog), gas-
   0.2000  0.0500   0.45  0.095 /       oil capillary pressure (pcgo) as
   0.3000  0.1030   0.25  0.115 /       tabulated functions of gas saturation
   0.4000  0.1900   0.11  0.145 /       (sgas).
   0.5000  0.3100   0.028 0.255 /
   0.5500  0.4200   0.0   0.386 /
/

SATOPT
   20* 1 /                              The 21st flag of the SATOPT keyword
                                        is responsible for the interpolation
                                        of the surface tension (see the STOG
                                        keyword below). If this option is
                                        activated then the STOG data is a
                                        piecewise linear function.

STOG
-- refpres                              We specify the gas-oil capillary
    5545 /                              pressure dependence on pressure.
-- pres  surftens                       The data in the fourth column in the
   1674   6.0   /                       SGOF table are at reference pressure
   2031   4.7   /                       5545 psi. This capillary pressure is
   2530   3.3   /                       scaled in accordance to the surface
   2991   2.2   /                       tension dependence on pressure as it
   3553   1.28  /                       is specified with the STOG keyword.
   4110   0.72  /                       The first column is pressure (pres)
   4544   0.444 /                       and the second column is surface tension
   4935   0.255 /                       (surftens).
   5255   0.155 /
   5545   0.090 /
   7000   0.050 /
/

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The saturation functions are specified within brackets SAT-ENDSAT
SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
  2 /                                   Here, we input saturation functions in
```

```
                                        the fractures media. The relative
                                        permeabilities are linear functions
                                        and capillary pressure is zero.

SWOF
-- swat  krw  krow  pcwo
   0.0   0.0  1.0   0.0  /
   1.0   1.0  0.0   0.0  /
/

SGOF
-- sgas  krg  krog  pcgo
   0.0   0.0  1.0   0.0  /
   1.0   1.0  0.0   0.0  /
/

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The fluid properties are specified within brackets EOS-ENDEOS
EOS       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

OPTBO                                 Use these options if you want a
  10*  10*1 /                         better fit to the SPE paper, otherwise
                                      you can comment it.

DENSITY                                 We specify the phases densities at
--   oil       water      gas           surface conditions (lb/cft).
   51.1400  65.0000     0.05800 /

PVTW                                    We specify water properties: reference
--  refpres   bwat   comprw   viswat    pressure (refpres), form. vol. factor at
    0.0     1.07000  0.35E-05  0.35  /  refpres (bwat),  water compressibility
                                        (comprw) and viscosity (viswat).

PVDG
--   pgas      bgas      visgas         We specify the properies of dry gas,
   1674.00    1.98000    0.01620 /      i.e. the formation volume factor (bgas),
   2031.00    1.62000    0.01710 /      and viscosity (visgas) on pressure
   2530.00    1.30000    0.01840 /      (pgas).
   2991.00    1.11000    0.01970 /
   3553.00    0.95900    0.02130 /
   4110.00    0.85500    0.02300 /
   4544.00    0.79500    0.02440 /
   4935.00    0.75100    0.02550 /
   5255.00    0.72000    0.02650 /
   5545.00    0.69600    0.02740 /
   7000.00    0.60000    0.03300 /
```

```
/

PVTO
--   rs      poil   boil    visoil      We specify the properties of live oil,
  0.36700 1674.00 1.30010  0.52900 /    i.e. the oil pressure (poil),
  0.44700 2031.00 1.33590  0.48700 /    formation volume factor (boil),
  0.56400 2530.00 1.38910  0.43600 /    viscosity (visoil) on gas-oil ratio
  0.67900 2991.00 1.44250  0.39700 /    of the solution (rs).
  0.83200 3553.00 1.51410  0.35100 /
  1.00000 4110.00 1.59380  0.31000 /
  1.14300 4544.00 1.66300  0.27800 /
  1.28500 4935.00 1.73150  0.24800 /
  1.41300 5255.00 1.79530  0.22900 /
  1.53000 5545.00 1.85400  0.21000
          5600.00 1.85330  0.21090
          5700.00 1.85210  0.21260
          5800.00 1.85090  0.21430
          5900.00 1.84970  0.21610
          6000.00 1.84850  0.21780
          6100.00 1.84730  0.21950
          6200.00 1.84610  0.22120 /
  2.25900 7000.00 2.19780  0.10900
          7100.00 2.19660  0.11070
          7200.00 2.19540  0.11240
          7300.00 2.19420  0.11410 /
/

ENDEOS    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

INIT      ######################## INIT section begins here #####################

          The initial equilibrium is specified within brackets EQL-ENDEQL
EQL       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

EQUIL
-- -------- -------- -------- -------- -------- --------
-- datum    datum     OWC     OWC      GOC      GOC
--   depth  pressure   depth   cappres   depth   cappres
-- -------- -------- -------- -------- -------- --------
     3025    6000      3300   1*         2900     1*     /
-- -------- -------- -------- -------- -------- --------

                    Using the keyword EQUIL we specify that
                    - pressure is 6000 psi at depth 3025 ft;
                    - oil-water contact is at depth 3300 ft (thus, at the
                      initial moment of time there is no mobile water because
                      3300 m is below the bottom layer);
```

```
                    - gas-oil contact is at depth 2900 ft (thus, at the
                      initial moment of time there is no free gas because
                      2900 m is above the top layer).

RSVD
-- depth  rs                            The initial gas-oil ratio is
   2900  1.53 /                         1.53 Mscf/stb.
   3300  1.53 /
/

ENDEQL    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

RPTSUM                                  At every report step, we report pressure
   PRES SWAT SOIL SGAS /                and saturations in the summary files.

RPTWELL
  WBHP WWIR WWIT WLPR WOPR
  WGPR WWPR WWCT WOPT WWPT
  WGPT WGOR /                           For every well we report these
                                        properties.

SCHEDULE   ##################### SCHEDULE section begins here ####################

VARS                                    We increase the maximum allowed pressure
  PRES  MAXV  7000 /                    in the simulation up to 7000 psi.
/

REPORTS                                 No convergence or other reports in the
   NOTHING /                            LOG file.

TUNING                                  Maximum timestep is 10 days.
  1* 10 /

WELLPROD
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
--   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
--                        rate     rate            rate              dimens
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
   PRODUCER OPEN ORAT   500       1*      5500     1E6 /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
/                                       We produce 500 stb/day oil at
                                        5500 psi minimum bottom-hole
                                        pressure.

TSTEP
 1 2 33.525 9*36.525 /                  We simulate 1 year of production.

WELLPROD
```

```
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
--   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
--                         rate     rate            rate              dimens
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
   PRODUCER OPEN ORAT   500       1*      5000     1E6 /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
/                                     We produce 500 stb/day oil at
                                      5000 psi minimum bottom-hole
                                      pressure.

TSTEP
 1 2 33.525 19*36.525 /               We simulate 2 years of production.

WELLPROD
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
--   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
--                         rate     rate            rate              dimens
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
   PRODUCER OPEN ORAT   500       1*      4500     1E6 /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
/                                     We produce 500 stb/day oil at
                                      4500 psi minimum bottom-hole
                                      pressure.

TSTEP
 1 2 33.525 19*36.525 /               We simulate 2 years of production.

WELLPROD
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
--   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
--                         rate     rate            rate              dimens
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
   PRODUCER OPEN ORAT   500       1*      4000     1E6 /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
/                                     We produce 500 stb/day oil at
                                      4000 psi minimum bottom-hole
                                      pressure.

TSTEP
 1 2 33.525 19*36.525 /               We simulate 2 years of production.

WELLPROD
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
--   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
--                         rate     rate            rate              dimens
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
   PRODUCER OPEN ORAT   500       1*      3500     1E6 /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
```

```
/                                       We produce 500 stb/day oil at
                                        3500 psi minimum bottom-hole
                                        pressure.

TSTEP
 1 2 33.525 19*36.525 /                 We simulate 2 years of production.

WELLPROD
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
--   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
--                        rate     rate           rate              dimens
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
   PRODUCER OPEN ORAT   500       1*      3000     1E6 /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
/                                       We produce 500 stb/day oil at
                                        3000 psi minimum bottom-hole
                                        pressure.

TSTEP
 1 2 33.525 19*36.525 /                 We simulate 2 years of production.

WELLPROD
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
--   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
--                        rate     rate           rate              dimens
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
   PRODUCER OPEN ORAT   500       1*      2500     1E6 /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
/                                       We produce 500 stb/day oil at
                                        2500 psi minimum bottom-hole
                                        pressure.

TSTEP
 1 2 33.525 19*36.525 /                 We simulate 2 years of production.

WELLPROD
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
--   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
--                        rate     rate           rate              dimens
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
   PRODUCER OPEN ORAT   500       1*      2000     1E6 /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
/                                       We produce 500 stb/day oil at
                                        2000 psi minimum bottom-hole
                                        pressure.

TSTEP
 1 2 33.525 29*36.525 /                 We simulate 3 years of production.
```

```
469
470 POST      ######################## POST section begins here ######################
471
472 CONVERT                                  We convert the output to ParaView
473                                          compatible format.
474
475 RPTPOST
476   NOTHING TIME WBHP WWCT WLPR
477   WOPR WGPR WWPR WOPT WGPT WGOR /         We save consolidated time series
478 POSTWELL                                 report for the PRODUCER well.
479   PRODUCER /
480 /
481
482 END        ############################################################################
```

9.6. 7th SPE Comparative Study; case 3a (BLACKOIL)

This problem is discussed in the paper: Nghiem, L. et al. 1991 Seventh SPE Comparative Solution Project: Modelling of Horizontal Wells in Reservoir Simulation. SPE Symposium on Reservoir Simulation, 17-20 February, Anaheim, California. DOI: 10.2118/21221-MS.. See a description of the problem, input data files and visualisations under *the link*.

SPE7-CASE3A.RUN

```
################################################################################

                    7TH SPE COMPARATIVE STUDY (CASE 3A)

    The problem is from SPE-21221 paper (case 3a):
    Nghiem, L. et al. 1991. Seventh SPE Comparative Solution Project: Modelling
        of Horizontal Wells in Reservoir Simulation. SPE Symposium on Reservoir
        Simulation, 17-20 February, Anaheim, California. DOI: 10.2118/21221-MS.

    See description at: http://dx.doi.org/10.2118/21221-MS

--------------------------------------------------------------------------------

    Last update on 10.01.2017
    Program Version: 2016.C01

################################################################################


   Note:
   We type all comments in lowercase letters while all keywords and mnemonics
      must be in uppercase letters.

--------------------------------------------------------------------------------

   Note:
-- any line beginning with '!' or '--' is a comment line;
   any line not beginning with an uppercase letter outside keyword
!  instruction is also a comment line.

--------------------------------------------------------------------------------

   Note:
   Any data line must be terminated by the slash '/' sign.

```

```
--------------------------------------------------------------------------------

   Note:
   We denote by the repeated symbols the following structural elements of this
      RUN-file:
      '###############' - delimits the sections of the RUN-file.
      '<<<<<<<<<<<<<<' - designates an opening bracket.
      '>>>>>>>>>>>>>>' - designates a closing bracket.
      '^^^^^^^^^^^^^^' - designates the keyword, which affects the order of the
                         data input for all the following keywords (both in the
                         current and in the following sections).
      '==============' - merge the keywords and the comments in paragraphs.

--------------------------------------------------------------------------------

RUNSPEC   ################### RUNSPEC section begins here ######################

BLACKOIL  ^^^^^^ We switch the program to the BLACKOIL EOS module ^^^^^^^^^^^^^^

                                       We enable:
WATER                                  - water phase;
OIL                                    - oil phase;
GAS                                    - gas phase;
DISGAS                                 - gas dissolution in oil phase.

FIELD     ^^^^^^^^^^^^ We use FIELD units ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
                                       Note, this keywords affects the units
                                       in many keywords below.

                                       We enable:
ROCKCOMP                               - rock compressibility;
CAPPRES                                - capillary pressure;
MUSEGWEL                               - multisegmented wells.

FAST                                   We use FAST option.

GRID      ###################### GRID section begins here ######################

          The grid is specified within brackets MAKE-ENDMAKE
MAKE      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
-- cartesian                           We select Cartesian gridding
--    grid   nx  ny  nz                option and specify the number of
      CART    9   9   6  /             grid blocks along every axis.
XYZBOUND
-- xmin-xmax   ymin-ymax   zmin-zmax    We specify the domain extent in feet.
   0    1      0    1      3590 3591 /  The top of reservoir is at depth 3590
                                       feet.

```

```
DXV                                     The grid blocks length along X axis
  9*300 /                               is 300 feet.

DYV                                     Here, we specify the grid blocks
  620 400 200 100 60 100 200 400 620 /  length along Y axis.

DZV                                     Here, we specify the thickness of the
  4*20 30 50 /                          layers.

WELSPECS
-- -------- -- ---- ---- -------- -- --- -- ------
-- well        i-   j-   datum       mod    flux
--     name     ind  ind   depth      el       num
-- -------- -- ---- ---- -------- -- --- -- ------    Here, we specify
   INJECTOR 1*    1  5    1*      6* 1*  4*  222  /   injection well used as a
-- -------- -- ---- ---- -------- -- --- -- ------    line source of water.
/                                       The AVG well model (default option)
                                        is used for the INJECTOR. The pipe
                                        segment associated with the INJECTOR
                                        can be farther referred to by using
                                        FLUXNUM=222.
COMPDAT
-- -------- ---- ---- ---- ---- ---- ------ -------- -------- -- --- -- -----
-- well     i-   j-   k-   k-   mode satnum well      well       dir    rough
--     name  ind  ind  min  max                index   diameter           ness
-- -------- ---- ---- ---- ---- ---- ------ -------- -------- -- --- -- -----
   INJECTOR   1    5   6    6   OPEN  1*    24.3457 /  24.3457 cP*bbl/day/psi
   INJECTOR   2    5   6    6   OPEN  1*    24.3457 /  =2.16e4 ft*mD
   INJECTOR   3    5   6    6   OPEN  1*    24.3457 /
   INJECTOR   4    5   6    6   OPEN  1*    24.3457 /
   INJECTOR   5    5   6    6   OPEN  1*    24.3457 /
   INJECTOR   6    5   6    6   OPEN  1*    24.3457 /
   INJECTOR   7    5   6    6   OPEN  1*    24.3457 /
   INJECTOR   8    5   6    6   OPEN  1*    24.3457 /
   INJECTOR   9    5   6    6   OPEN  1*    24.3457 /
-- -------- ---- ---- ---- ---- ---- ------ -------- -------- -- --- -- -----
/                                       Here, we specify the INJECTOR
                                        completions in the bottom layer. The
                                        well index is specified explicitly.

CARFIN    <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
-- name  imin imax jmin jmax kmin kmax nx ny nz    Here, we specify local grid
   LGR1    5    9    5    5    1    1   9  3  3 /  refinement around the
                                                   production well.
          The refinement is applied to the box of gridblocks consisting of the
          row of blocks containg the production well plus an extra block
          on either side.

```

```
NXFIN                                  Two blocks at each end of the refinement
   2*2 1 2*2 /                         box are redefined into 4 blocks.
HYFIN                                  The extention of the grid blocks along
   24 12 24 /                          Y-axis.
HZFIN                                  The thickness of the layers in the
   8  4  8  /                         refinement.

WELSPECS
-- -------- -- ---- ---- -------- -- --- -- ------
-- well        i-   j-   datum       mod    flux
--     name     ind  ind   depth      el       num
-- -------- -- ---- ---- -------- -- --- -- ------    Here, we specify the
   PRODUCER 1*  7    2    1*      6* SEG 4*  111  /   production well
-- -------- -- ---- ---- -------- -- --- -- ------    and use segmented model.
/                                      All segements asscociated with the well
                                       PRODUCER have FLUXNUM=111.

COMPDAT
-- -------- ---- ---- ---- ---- ---- ------ -------- -------- -- --- -- -----
-- well     i-   j-   k-   k-   mode satnum well      well       dir    rough
--     name  ind  ind  min  max                index   diameter           ness
-- -------- ---- ---- ---- ---- ---- ------ -------- -------- -- --- -- -----
   PRODUCER   7    2   2    2   OPEN  1*    1*       0.375    3* 'X' 1* 0.001 /
   PRODUCER   6    2   2    2   OPEN  1*    1*       0.375    3* 'X' 1* 0.001 /
   PRODUCER   5    2   2    2   OPEN  1*    1*       0.375    3* 'X' 1* 0.001 /
   PRODUCER   4    2   2    2   OPEN  1*    1*       0.375    3* 'X' 1* 0.001 /
   PRODUCER   3    2   2    2   OPEN  1*    1*       0.375    3* 'X' 1* 0.001 /
-- -------- ---- ---- ---- ---- ---- ------ -------- -------- -- --- -- -----
/                                      Here, we specify completions for the
                                       production well.
ENDFIN    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

DZWELL                                 This is for visualization.
  300 30 /

ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

EQUALS
--  --------  --------                 We equal
    PORO      0.2       /              - porosity to 0.2;
    PERMX     300       /              - X-axis permeability to 300;
    PERMY     300       /              - Y-axis permeability to 300;
    PERMZ     30        /              - Z-axis permeability to 30.
--  --------  --------
/

RPTGRID                                This properties are reported from the
  DEPTH PORO PERMX PERMY PERMZ /       section GRID.
```

```
180
181 PROPS     ###################### PROPS section begins here ####################
182
183           Rock properties are specified within brackets ROCK-ENDROCK
184 ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
185    /
186
187 ROCKECL                                  We specify the reference pressure (psia)
188 -- -refpres- --compr--                   and the rock compressibility (1/psi).
189       3600     4.0D-6   /
190
191 ENDROCK   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
192
193           The relative permeabilities are specified within brackets SAT-ENDSAT
194 SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
195    /
196
197 STONE1                                   We use the 1st Stone formula for the
198                                          oil relative permeability in the
199                                          three-phase-flow regions.
200
201 --STONE2                                    Uncomment this keyword if you want to
202                                          use the 2nd Stone formula.
203
204 SWOF
205 -- swat   krw      krow      pcow
206    0.22    0        1         6.30  /  Here, we specify the water relative
207    0.3     0.07     0.4       3.60  /  permeability (krw), oil relative
208    0.4     0.15     0.125     2.70  /  permeability for the oil-water flows
209    0.5     0.24     0.0649    2.25  /  (krow) and oil-water capillary
210    0.6     0.33     0.0048    1.80  /  pressure as the functions of water
211    0.8     0.65     0         0.90  /  saturation (swat).
212    0.9     0.83     0         0.45  /
213    1       1        0         0     /
214 /
215
216 SGOF
217 -- sgas    krg       krog   pcog
218    0       0         1      0   /      Here, we specify the gas relative
219    0.04    0         0.6    0.2 /      permeability (krg), oil relative
220    0.1     0.022     0.33   0.5 /      permeability for the oil-gas flows
221    0.2     0.1       0.1    1.0 /      (krg) and gas-oil capillary pressure
222    0.3     0.24      0.02   1.5 /      as the functions of gas saturation
223    0.4     0.34      0      2.0 /      (sgas).
224    0.5     0.42      0      2.5 /
225    0.6     0.5       0      3.0 /
226    0.7     0.8125    0      3.5 /
227    0.78    1         0      3.9 /
```

```
/

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The fluid properties are specified within brackets EOS-ENDEOS
EOS       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

DENSITY                                We specify the phases densities at
--   oil       water      gas          surface conditions (lb/cft).
     45        62.14     0.0702  /

PVTW                                   We specify water properties: reference
--  refpres   bwat    comprw   viswat  pressure (refpres), form. vol. factor at
    14.7     1.0142   3.0D-6   0.96  / refpres (bwats), viscosity (viswat) and
                                       water compressibility (comprw).

PVDG
--   pgas    bgas  visgas              We specify the properies of dry gas,
      400    5.9   0.013  /            i.e. the formation volume factor (bgas),
      800    2.95  0.0135 /            and viscosity (visgas) on pressure
     1200    1.96  0.014  /            (pgas).
     1600    1.47  0.0145 /
     2000    1.18  0.015  /
     2400    0.98  0.0155 /
     2800    0.84  0.016  /
     3200    0.74  0.0165 /
     3600    0.65  0.017  /
     4000    0.59  0.0175 /
     4400    0.54  0.018  /
     4800    0.49  0.0185 /
     5200    0.45  0.019  /
     5600    0.42  0.0195 /
/

PVTO
--   rs      poil   boil    visoil     We specify the properties of live oil,
   0.165      400   1.012    1.17  /   i.e. the oil pressure (poil),
   0.335      800   1.0255   1.14  /   formation volume factor (boil),
   0.500     1200   1.038    1.11  /   viscosity (visoil) on gas-oil ratio
   0.665     1600   1.051    1.08  /   of the solution (rs).
   0.828     2000   1.063    1.06  /
   0.985     2400   1.075    1.03  /
   1.130     2800   1.087    1.00  /
   1.270     3200   1.0985   0.98  /
   1.390     3600   1.11     0.95  /
   1.500     4000   1.12     0.94  /
   1.600     4400   1.13     0.92  /
```

```
   1.676     4800   1.14     0.91  /
   1.750     5200   1.148    0.9   /
   1.810     5600   1.155    0.89
             6000   1.1504   0.89
             6400   1.1458   0.89
             6800   1.1412   0.89
             7200   1.1367   0.89  /
/

ENDEOS    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

INIT      ######################## INIT section begins here ######################

                                    We specify pressure and water saturation
EQUALS                              for every layer:
  PRES  3600  4*  2*1 /             - first (top) layer;
  SWAT  0.289 /                     - first (top) layer;
  PRES  3608  4*  2*2 /             - second layer;
  SWAT  0.348 /                     - second layer;
  PRES  3616  4*  2*3 /             etc.
  SWAT  0.473 /
  PRES  3623  4*  2*4 /
  SWAT  0.649 /
  PRES  3633  4*  2*5 /
  SWAT  0.869 /
  PRES  3650  4*  2*6 /
  SWAT  1.0 /
/

COPY                                We copy pressure array to the buble
  PRES PBUB /                       pressure array.
/

REGALL                              This keyword affects the following
                                    keyword EQUALREG.

                                    We specify initial parameters of the
EQUALREG                            fluid in the boreholes.
  PRES 3600 FLUXNUM 111 /           -- in the PRODUCER.
  SWAT 0.289 /
  PBUB 3600 /
  PRES 3650 FLUXNUM 222 /           -- in the INJECTOR.
  PBUB 3650 /
  SWAT 1.0 /
/

RPTSUM                              This parameters are saved at every
 PRES SWAT SOIL SGAS PBUB RS  /     report time.
```

```
RPTWELL
  WBHP WWIR WOPR WWPR WGPR WLPR
  WOPT WWPT WGPT WLPT WWOR WGOR/

SCHEDULE   ################### SCHEDULE section begins here ####################

REPORTS                                 No reports in the LOG-file.
  NOTHING /

WEEKTOL                                 We use the WEEKTOL option.

WELLPROD
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
--   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
--                        rate     rate            rate              dimens
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
   PRODUCER OPEN LRAT  9000      1*       1500     1E6 /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
/                                       We produce 9000 bbl/day of liquid
                                        (oil and water) at the minimum
                                        bottom-hole pressure 1500 psi.

WELLINJE
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
--   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
--                        rate     rate            rate              dimens
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
   INJECTOR OPEN WRAT  100000.         1*     3700      1E6      WATER /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
/                                       We inject 10000 bbl/day of water
                                        at the maximum bottom-hole pressure
                                        3700 psi.

VARS                                    Maximimum pressure alteration at
  PRES DMAX 500 /                       a time step is 500 psi.
/

TUNING                                  The next timestep is limited to
  0.1 25 /                              0.1 days. The maximum timestep is
                                        25 days.

TSTEP                                   We simulate the production over
  15*100 /                              1500 days reporting every 100 days.

POST      ###################### POST section begins here ####################

RPTPOST                                 We save time series report for
```

```
372   TIME WBHP WWIR /                        the INJECTOR.
373 POSTWELL
374   INJECTOR /
375 /
376
377 RPTPOST
378   NOTHING TIME WBHP WOPR WOPT WWPT
379           WWOR WLPR WGPT WWPR WGOR
380           WGPR /                          We save time series report for the
381 POSTWELL                                  PRODUCER.
382   PRODUCER /
383 /
384
385 CONVERT                                   We convert the output to ParaView
386                                           compatible format.
387
388 END        ###########################################################################
```

9.7. 7th SPE Comparative Study; case 3b (BLACKOIL)

This problem is discussed in the paper: Nghiem, L. et al. 1991 Seventh SPE Comparative Solution Project: Modelling of Horizontal Wells in Reservoir Simulation. SPE Symposium on Reservoir Simulation, 17-20 February, Anaheim, California. DOI: 10.2118/21221-MS.. See a description of the problem, input data files and visualisations under *the link*.

SPE7-CASE3B.RUN

```
################################################################################

                    7TH SPE COMPARATIVE STUDY (CASE 3B)

    The problem is from SPE-21221 paper (case 3b):
    Nghiem, L. et al. 1991. Seventh SPE Comparative Solution Project: Modelling
        of Horizontal Wells in Reservoir Simulation. SPE Symposium on Reservoir
        Simulation, 17-20 February, Anaheim, California. DOI: 10.2118/21221-MS.

    See description at: http://dx.doi.org/10.2118/21221-MS

--------------------------------------------------------------------------------

    Last update on 10.01.2017
    Program Version: 2016.C01

################################################################################


   Note:
   We type all comments in lowercase letters while all keywords and mnemonics
      must be in uppercase letters.

--------------------------------------------------------------------------------

   Note:
-- any line beginning with '!' or '--' is a comment line;
   any line not beginning with an uppercase letter outside keyword
!  instruction is also a comment line.

--------------------------------------------------------------------------------

   Note:
   Any data line must be terminated by the slash '/' sign.

```

```
--------------------------------------------------------------------------------

   Note:
   We denote by the repeated symbols the following structural elements of this
      RUN-file:
      '###############' - delimits the sections of the RUN-file.
      '<<<<<<<<<<<<<<' - designates an opening bracket.
      '>>>>>>>>>>>>>>' - designates a closing bracket.
      '^^^^^^^^^^^^^^' - designates the keyword, which affects the order of the
                         data input for all the following keywords (both in the
                         current and in the following sections).
      '==============' - merge the keywords and the comments in paragraphs.

--------------------------------------------------------------------------------

RUNSPEC   ################### RUNSPEC section begins here ######################

BLACKOIL  ^^^^^^ We switch the program to the BLACKOIL EOS module ^^^^^^^^^^^^^^

                                       We enable:
WATER                                  - water phase;
OIL                                    - oil phase;
GAS                                    - gas phase;
DISGAS                                 - gas dissolution in oil phase.

FIELD     ^^^^^^^^^^^^ We use FIELD units ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
                                       Note, this keywords affects the units
                                       in many keywords below.

                                       We enable:
ROCKCOMP                               - rock compressibility;
CAPPRES                                - capillary pressure;
MUSEGWEL                               - multisegmented wells.

FAST                                   We use FAST option.

GRID      ###################### GRID section begins here ######################

          The grid is specified within brackets MAKE-ENDMAKE
MAKE      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
-- cartesian                           We select Cartesian gridding
--    grid   nx  ny  nz                option and specify the number of
      CART    9   9   6  /             grid blocks along every axis.
XYZBOUND
-- xmin-xmax   ymin-ymax   zmin-zmax    We specify the domain extent in feet.
   0    1      0    1      3590 3591 /  The top of reservoir is at depth 3590
                                       feet.

```

```
DXV                                     The grid blocks length along X axis
  9*300 /                               is 300 feet.

DYV                                     Here, we specify the grid blocks
  620 400 200 100 60 100 200 400 620 /  length along Y axis.

DZV                                     Here, we specify the thickness of the
  4*20 30 50 /                          layers.

WELSPECS
-- -------- -- ---- ---- -------- -- --- -- ------
-- well        i-   j-   datum       mod    flux
--     name     ind  ind   depth      el       num
-- -------- -- ---- ---- -------- -- --- -- ------    Here, we specify
   INJECTOR 1*    1  5    1*      6* 1*  4*  222  /   injection well used as a
-- -------- -- ---- ---- -------- -- --- -- ------    line source of water.
/                                       The AVG well model (default option)
                                        is used for the INJECTOR. The pipe
                                        segment associated with the INJECTOR
                                        can be farther referred to by using
                                        FLUXNUM=222.
COMPDAT
-- -------- ---- ---- ---- ---- ---- ------ -------- -------- -- --- -- -----
-- well     i-   j-   k-   k-   mode satnum well      well       dir    rough
--     name  ind  ind  min  max                index   diameter           ness
-- -------- ---- ---- ---- ---- ---- ------ -------- -------- -- --- -- -----
   INJECTOR   1    5   6    6   OPEN  1*    24.3457 /  24.3457 cP*bbl/day/psi
   INJECTOR   2    5   6    6   OPEN  1*    24.3457 /  =2.16e4 ft*mD
   INJECTOR   3    5   6    6   OPEN  1*    24.3457 /
   INJECTOR   4    5   6    6   OPEN  1*    24.3457 /
   INJECTOR   5    5   6    6   OPEN  1*    24.3457 /
   INJECTOR   6    5   6    6   OPEN  1*    24.3457 /
   INJECTOR   7    5   6    6   OPEN  1*    24.3457 /
   INJECTOR   8    5   6    6   OPEN  1*    24.3457 /
   INJECTOR   9    5   6    6   OPEN  1*    24.3457 /
-- -------- ---- ---- ---- ---- ---- ------ -------- -------- -- --- -- -----
/                                       Here, we specify the INJECTOR
                                        completions in the bottom layer. The
                                        well index is specified explicitly.

CARFIN    <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
-- name  imin imax jmin jmax kmin kmax nx ny nz    Here, we specify local grid
   LGR1    1    9    5    5    1    1  13  3  3 /  refinement around the
                                                   production well.
          The refinement is applied to the box of gridblocks consisting of the
          row of blocks containg the production well plus an extra block
          on either side.

```

```
NXFIN                                  Two blocks at each end of the refinement
   2*2 5*1 2*2 /                       box are redefined into 4 blocks.
HYFIN                                  The extention of the grid blocks along
   24 12 24 /                          Y-axis.
HZFIN                                  The thickness of the layers in the
   8  4  8  /                         refinement.

WELSPECS
-- -------- -- ---- ---- -------- -- --- -- ------
-- well        i-   j-   datum       mod    flux
--     name     ind  ind   depth      el       num
-- -------- -- ---- ---- -------- -- --- -- ------    Here, we specify the
   PRODUCER 1*  11   2    1*      6* SEG 4*  111  /   production well
-- -------- -- ---- ---- -------- -- --- -- ------    and use segmented model.
/                                      All segements asscociated with the well
                                       PRODUCER have FLUXNUM=111.

COMPDAT
-- -------- ---- ---- ---- ---- ---- ------ -------- -------- -- --- -- -----
-- well     i-   j-   k-   k-   mode satnum well      well       dir    rough
--     name  ind  ind  min  max                index   diameter           ness
-- -------- ---- ---- ---- ---- ---- ------ -------- -------- -- --- -- -----
   PRODUCER  11    2   2    2   OPEN  1*    1*       0.375    3* 'X' 1* 0.001 /
   PRODUCER  10    2   2    2   OPEN  1*    1*       0.375    3* 'X' 1* 0.001 /
   PRODUCER   9    2   2    2   OPEN  1*    1*       0.375    3* 'X' 1* 0.001 /
   PRODUCER   8    2   2    2   OPEN  1*    1*       0.375    3* 'X' 1* 0.001 /
   PRODUCER   7    2   2    2   OPEN  1*    1*       0.375    3* 'X' 1* 0.001 /
   PRODUCER   6    2   2    2   OPEN  1*    1*       0.375    3* 'X' 1* 0.001 /
   PRODUCER   5    2   2    2   OPEN  1*    1*       0.375    3* 'X' 1* 0.001 /
   PRODUCER   4    2   2    2   OPEN  1*    1*       0.375    3* 'X' 1* 0.001 /
   PRODUCER   3    2   2    2   OPEN  1*    1*       0.375    3* 'X' 1* 0.001 /
-- -------- ---- ---- ---- ---- ---- ------ -------- -------- -- --- -- -----
/                                      Here, we specify completions for the
                                       production well.
ENDFIN    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

DZWELL                                 This is for visualization.
  300 30 /

ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

EQUALS
--  --------  --------                 We equal
    PORO      0.2       /              - porosity to 0.2;
    PERMX     300       /              - X-axis permeability to 300;
    PERMY     300       /              - Y-axis permeability to 300;
    PERMZ     30        /              - Z-axis permeability to 30.
--  --------  --------
```

```
/

RPTGRID                                  This properties are reported from the
  DEPTH PORO PERMX PERMY PERMZ /         section GRID.

PROPS     ######################## PROPS section begins here ####################

          Rock properties are specified within brackets ROCK-ENDROCK
ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

ROCKECL                                  We specify the reference pressure (psia)
-- -refpres- --compr--                   and the rock compressibility (1/psi).
      3600     4.0D-6   /

ENDROCK   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
          The relative permeabilities are specified within brackets SAT-ENDSAT
SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

STONE1                                   We use the 1st Stone formula for the
                                         oil relative permeability in the
                                         three-phase-flow regions.

--STONE2                                     Uncomment this keyword if you want to
                                         use the 2nd Stone formula.

SWOF
-- swat   krw      krow      pcow
   0.22    0        1         6.30  /  Here, we specify the water relative
   0.3     0.07     0.4       3.60  /  permeability (krw), oil relative
   0.4     0.15     0.125     2.70  /  permeability for the oil-water flows
   0.5     0.24     0.0649    2.25  /  (krow) and oil-water capillary
   0.6     0.33     0.0048    1.80  /  pressure as the functions of water
   0.8     0.65     0         0.90  /  saturation (swat).
   0.9     0.83     0         0.45  /
   1       1        0         0     /
/

SGOF
-- sgas    krg       krog   pcog
   0       0         1      0   /      Here, we specify the gas relative
   0.04    0         0.6    0.2 /      permeability (krg), oil relative
   0.1     0.022     0.33   0.5 /      permeability for the oil-gas flows
   0.2     0.1       0.1    1.0 /      (krg) and gas-oil capillary pressure
   0.3     0.24      0.02   1.5 /      as the functions of gas saturation
   0.4     0.34      0      2.0 /      (sgas).
```

```
   0.5     0.42      0      2.5 /
   0.6     0.5       0      3.0 /
   0.7     0.8125    0      3.5 /
   0.78    1         0      3.9 /
/

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The fluid properties are specified within brackets EOS-ENDEOS
EOS       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

DENSITY                                 We specify the phases densities at
--   oil       water      gas           surface conditions (lb/cft).
     45        62.14     0.0702  /

PVTW                                    We specify water properties: reference
--  refpres   bwat    comprw   viswat   pressure (refpres), form. vol. factor at
    14.7     1.0142   3.0D-6   0.96  /  refpres (bwats), viscosity (viswat) and
                                        water compressibility (comprw).

PVDG
--   pgas    bgas  visgas               We specify the properies of dry gas,
      400    5.9   0.013  /             i.e. the formation volume factor (bgas),
      800    2.95  0.0135 /             and viscosity (visgas) on pressure
     1200    1.96  0.014  /             (pgas).
     1600    1.47  0.0145 /
     2000    1.18  0.015  /
     2400    0.98  0.0155 /
     2800    0.84  0.016  /
     3200    0.74  0.0165 /
     3600    0.65  0.017  /
     4000    0.59  0.0175 /
     4400    0.54  0.018  /
     4800    0.49  0.0185 /
     5200    0.45  0.019  /
     5600    0.42  0.0195 /
/

PVTO
--   rs      poil   boil    visoil      We specify the properties of live oil,
   0.165      400   1.012    1.17  /    i.e. the oil pressure (poil),
   0.335      800   1.0255   1.14  /    formation volume factor (boil),
   0.500     1200   1.038    1.11  /    viscosity (visoil) on gas-oil ratio
   0.665     1600   1.051    1.08  /    of the solution (rs).
   0.828     2000   1.063    1.06  /
   0.985     2400   1.075    1.03  /
   1.130     2800   1.087    1.00  /
```

```
   1.270     3200   1.0985   0.98  /
   1.390     3600   1.11     0.95  /
   1.500     4000   1.12     0.94  /
   1.600     4400   1.13     0.92  /
   1.676     4800   1.14     0.91  /
   1.750     5200   1.148    0.9   /
   1.810     5600   1.155    0.89
             6000   1.1504   0.89
             6400   1.1458   0.89
             6800   1.1412   0.89
             7200   1.1367   0.89  /
/

ENDEOS    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

INIT      ####################### INIT section begins here #####################

                                        We specify pressure and water saturation
EQUALS                                  for every layer:
  PRES  3600  4*  2*1 /                 - first (top) layer;
  SWAT  0.289 /                         - first (top) layer;
  PRES  3608  4*  2*2 /                 - second layer;
  SWAT  0.348 /                         - second layer;
  PRES  3616  4*  2*3 /                 etc.
  SWAT  0.473 /
  PRES  3623  4*  2*4 /
  SWAT  0.649 /
  PRES  3633  4*  2*5 /
  SWAT  0.869 /
  PRES  3650  4*  2*6 /
  SWAT  1.0 /
/

COPY                                    We copy pressure array to the buble
  PRES PBUB /                           pressure array.
/

REGALL                                  This keyword affects the following
                                        keyword EQUALREG.

                                        We specify initial parameters of the
EQUALREG                                fluid in the boreholes.
  PRES 3600 FLUXNUM 111 /               -- in the PRODUCER.
  SWAT 0.289 /
  PBUB 3600 /
  PRES 3650 FLUXNUM 222 /               -- in the INJECTOR.
  PBUB 3650 /
  SWAT 1.0 /
```

```
/

RPTSUM                                  This parameters are saved at every
 PRES SWAT SOIL SGAS PBUB RS  /         report time.

RPTWELL
  WBHP WWIR WOPR WWPR WGPR WLPR
  WOPT WWPT WGPT WLPT WWOR WGOR/

SCHEDULE   ################### SCHEDULE section begins here ####################

REPORTS                                 No reports in the LOG-file.
  NOTHING /

WEEKTOL                                 We use the WEEKTOL option.

WELLPROD
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
--   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
--                        rate     rate            rate              dimens
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
   PRODUCER OPEN LRAT  9000      1*       1500     1E6 /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
/                                       We produce 9000 bbl/day of liquid
                                        (oil and water) at the minimum
                                        bottom-hole pressure 1500 psi.

WELLINJE
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
--   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
--                        rate     rate            rate              dimens
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
   INJECTOR OPEN WRAT  100000.         1*     3700      1E6      WATER /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
/                                       We inject 10000 bbl/day of water
                                        at the maximum bottom-hole pressure
                                        3700 psi.

VARS                                    Maximimum pressure alteration at
  PRES DMAX 500 /                       a time step is 500 psi.
/

TUNING                                  The next timestep is limited to
  0.1 25 /                              0.1 days. The maximum timestep is
                                        25 days.

TSTEP                                   We simulate the production over
  15*100 /                              1500 days reporting every 100 days.
```

```
372
373 POST      ######################## POST section begins here ######################
374
375 RPTPOST                                  We save time series report for
376   TIME WBHP WWIR /                       the INJECTOR.
377 POSTWELL
378   INJECTOR /
379 /
380
381 RPTPOST
382   NOTHING TIME WBHP WOPR WOPT WWPT
383           WWOR WLPR WGPT WWPR WGOR
384           WGPR /                         We save time series report for the
385 POSTWELL                                 PRODUCER.
386   PRODUCER /
387 /
388
389 CONVERT                                  We convert the output to ParaView
390                                          compatible format.
391
392 END        ############################################################################
```

9.8. 7th SPE Comparative Study; case 4a (BLACKOIL)

This problem is discussed in the paper: Nghiem, L. et al. 1991 Seventh SPE Comparative Solution Project: Modelling of Horizontal Wells in Reservoir Simulation. SPE Symposium on Reservoir Simulation, 17-20 February, Anaheim, California. DOI: 10.2118/21221-MS.. See a description of the problem, input data files and visualisations under *the link*.

SPE7-CASE4A.RUN

```
################################################################################

                    7TH SPE COMPARATIVE STUDY (CASE 4A)

    The problem is from SPE-21221 paper (case 4a):
    Nghiem, L. et al. 1991. Seventh SPE Comparative Solution Project: Modelling
        of Horizontal Wells in Reservoir Simulation. SPE Symposium on Reservoir
        Simulation, 17-20 February, Anaheim, California. DOI: 10.2118/21221-MS.

    See description at: http://dx.doi.org/10.2118/21221-MS

--------------------------------------------------------------------------------

    Last update on 10.01.2017
    Program Version: 2016.C01

################################################################################


   Note:
   We type all comments in lowercase letters while all keywords and mnemonics
      must be in uppercase letters.

--------------------------------------------------------------------------------

   Note:
-- any line beginning with '!' or '--' is a comment line;
   any line not beginning with an uppercase letter outside keyword
!  instruction is also a comment line.

--------------------------------------------------------------------------------

   Note:
   Any data line must be terminated by the slash '/' sign.

```

```
--------------------------------------------------------------------------------

   Note:
   We denote by the repeated symbols the following structural elements of this
      RUN-file:
      '###############' - delimits the sections of the RUN-file.
      '<<<<<<<<<<<<<<' - designates an opening bracket.
      '>>>>>>>>>>>>>>' - designates a closing bracket.
      '^^^^^^^^^^^^^^' - designates the keyword, which affects the order of the
                         data input for all the following keywords (both in the
                         current and in the following sections).
      '==============' - merge the keywords and the comments in paragraphs.

--------------------------------------------------------------------------------

RUNSPEC   ################### RUNSPEC section begins here ######################

BLACKOIL  ^^^^^^ We switch the program to the BLACKOIL EOS module ^^^^^^^^^^^^^^

                                       We enable:
WATER                                  - water phase;
OIL                                    - oil phase;
GAS                                    - gas phase;
DISGAS                                 - gas dissolution in oil phase.

FIELD     ^^^^^^^^^^^^ We use FIELD units ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
                                       Note, this keywords affects the units
                                       in many keywords below.

                                       We enable:
ROCKCOMP                               - rock compressibility;
CAPPRES                                - capillary pressure;
MUSEGWEL                               - multisegmented wells.

OPTIONS                                Piecewise linear interpolation
  53* 1 /                              of relative permeabilities. Use this
                                       option if you want a better fit to the
                                       SPE-21221 paper.

FAST                                   We use FAST option.

GRID      ###################### GRID section begins here ######################

          The grid is specified within brackets MAKE-ENDMAKE
MAKE      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
-- cartesian                           We select Cartesian gridding
--    grid   nx  ny  nz                option and specify the number of
      CART    9   9   6  /             grid blocks along every axis.
```

```
XYZBOUND
-- xmin-xmax   ymin-ymax   zmin-zmax    We specify the domain extent in feet.
   0    1      0    1      3590 3591 /  The top of reservoir is at depth 3590
                                        feet.

DXV                                     The grid blocks length along X axis
  9*300 /                               is 300 feet.

DYV                                     Here, we specify the grid blocks
  620 400 200 100 60 100 200 400 620 /  length along Y axis.

DZV                                     Here, we specify the thickness of the
  4*20 30 50 /                          layers.

WELSPECS
-- -------- -- ---- ---- -------- -- --- -- ------
-- well        i-   j-   datum       mod    flux
--     name     ind  ind   depth      el       num
-- -------- -- ---- ---- -------- -- --- -- ------    Here, we specify
   INJECTOR 1*    1  5    1*      6* 1*  4*  222  /   both wells
   PRODUCER 1*    8  5    1*      6* SEG 4*  111  /
-- -------- -- ---- ---- -------- -- --- -- ------
/                                       The AVG well model (default option)
                                        is used for the INJECTOR. and the
                                        SEG model is used for the PRODUCER.
                                        The pipe segment associated with the
                                        INJECTOR can be farther referred to
                                        by using FLUXNUM=222, and the pipe
                                        segments asscoiated with the PRODUCER
                                        can be reffered to by FLUXNUM=111.
COMPDAT
-- -------- ---- ---- ---- ---- ---- ------ -------- -------- -- --- -- -----
-- well     i-   j-   k-   k-   mode satnum well      well       dir    rough
--     name  ind  ind  min  max                index   diameter           ness
-- -------- ---- ---- ---- ---- ---- ------ -------- -------- -- --- -- -----
   INJECTOR   1    5   6    6   OPEN  1*    243.457 /  243.457 cP*bbl/day/psi
   INJECTOR   2    5   6    6   OPEN  1*    243.457 /  =2.16e5 ft*mD
   INJECTOR   3    5   6    6   OPEN  1*    243.457 /
   INJECTOR   4    5   6    6   OPEN  1*    243.457 /
   INJECTOR   5    5   6    6   OPEN  1*    243.457 /
   INJECTOR   6    5   6    6   OPEN  1*    243.457 /
   INJECTOR   7    5   6    6   OPEN  1*    243.457 /
   INJECTOR   8    5   6    6   OPEN  1*    243.457 /
   INJECTOR   9    5   6    6   OPEN  1*    243.457 /
-- -------- ---- ---- ---- ---- ---- ------ -------- -------- -- --- -- -----
   PRODUCER   8    5   1    1   OPEN  1*    1*       0.375    3* 'X' 1* 0.001 /
   PRODUCER   7    5   1    1   OPEN  1*    1*       0.375    3* 'X' 1* 0.001 /
   PRODUCER   6    5   1    1   OPEN  1*    1*       0.375    3* 'X' 1* 0.001 /
```

```
-- -------- ---- ---- ---- ---- ---- ------ -------- -------- -- --- -- -----
/                                       Here, we specify the INJECTOR
                                        completions in the bottom layer. The
                                        well index is specified explicitly.
                                        And we specify the PRODUCER completions
                                        in the top layer.

DZWELL                                  This is for visualization.
  300 30 /

ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

EQUALS
--  --------  --------                  We equal
    PORO      0.2      /                - porosity to 0.2;
    PERMX     3000     /               - X-axis permeability to 300;
    PERMY     3000     /               - Y-axis permeability to 300;
    PERMZ     300      /               - Z-axis permeability to 30.
--  --------  --------
/

RPTGRID                                 This properties are reported from the
  DEPTH PORO PERMX PERMY PERMZ ASCII NDTNAME/        section GRID.

PROPS     ######################## PROPS section begins here ####################

          Rock properties are specified within brackets ROCK-ENDROCK
ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

ROCKECL                                 We specify the reference pressure (psia)
-- -refpres- --compr--                  and the rock compressibility (1/psi).
      3600     4.0D-6   /

ENDROCK   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The relative permeabilities are specified within brackets SAT-ENDSAT
SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

STONE2                                  We use the 2nd Stone formula for the
                                        oil relative permeability in the
                                        three-phase-flow regions.

SWOF
-- swat   krw      krow      pcow
   0.22    0        1         6.30  /  Here, we specify the water relative
   0.3     0.07     0.4       3.60  /  permeability (krw), oil relative
```

```
   0.4     0.15     0.125     2.70  /  permeability for the oil-water flows
   0.5     0.24     0.0649    2.25  /  (krow) and oil-water capillary
   0.6     0.33     0.0048    1.80  /  pressure as the functions of water
   0.8     0.65     0         0.90  /  saturation (swat).
   0.9     0.83     0         0.45  /
   1       1        0         0     /
/

SGOF
-- sgas    krg       krog   pcog
   0       0         1      0   /      Here, we specify the gas relative
   0.04    0         0.6    0.2 /      permeability (krg), oil relative
   0.1     0.022     0.33   0.5 /      permeability for the oil-gas flows
   0.2     0.1       0.1    1.0 /      (krg) and gas-oil capillary pressure
   0.3     0.24      0.02   1.5 /      as the functions of gas saturation
   0.4     0.34      0      2.0 /      (sgas).
   0.5     0.42      0      2.5 /
   0.6     0.5       0      3.0 /
   0.7     0.8125    0      3.5 /
   0.78    1         0      3.9 /
/

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The fluid properties are specified within brackets EOS-ENDEOS
EOS       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

DENSITY                                  We specify the phases densities at
--   oil       water      gas           surface conditions (lb/cft).
     45        62.14     0.0702  /

PVTW                                     We specify water properties: reference
--  refpres   bwat    comprw   viswat    pressure (refpres), form. vol. factor at
    14.7     1.0142   3.0D-6   0.96  /  refpres (bwats), viscosity (viswat) and
                                         water compressibility (comprw).

PVDG
--   pgas    bgas  visgas               We specify the properies of dry gas,
      400    5.9   0.013  /             i.e. the formation volume factor (bgas),
      800    2.95  0.0135 /             and viscosity (visgas) on pressure
     1200    1.96  0.014  /             (pgas).
     1600    1.47  0.0145 /
     2000    1.18  0.015  /
     2400    0.98  0.0155 /
     2800    0.84  0.016  /
     3200    0.74  0.0165 /
     3600    0.65  0.017  /
```

```
     4000    0.59  0.0175 /
     4400    0.54  0.018  /
     4800    0.49  0.0185 /
     5200    0.45  0.019  /
     5600    0.42  0.0195 /
/

PVTO
--   rs      poil   boil    visoil      We specify the properties of live oil,
   0.165      400   1.012    1.17  /    i.e. the oil pressure (poil),
   0.335      800   1.0255   1.14  /    formation volume factor (boil),
   0.500     1200   1.038    1.11  /    viscosity (visoil) on gas-oil ratio
   0.665     1600   1.051    1.08  /    of the solution (rs).
   0.828     2000   1.063    1.06  /
   0.985     2400   1.075    1.03  /
   1.130     2800   1.087    1.00  /
   1.270     3200   1.0985   0.98  /
   1.390     3600   1.11     0.95  /
   1.500     4000   1.12     0.94  /
   1.600     4400   1.13     0.92  /
   1.676     4800   1.14     0.91  /
   1.750     5200   1.148    0.9   /
   1.810     5600   1.155    0.89
             6000   1.1504   0.89
             6400   1.1458   0.89
             6800   1.1412   0.89
             7200   1.1367   0.89  /
/

ENDEOS    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

INIT      ######################## INIT section begins here ######################

                                    We specify pressure and water saturation
EQUALS                              for every layer:
  PRES  3600  4*  2*1 /             - first (top) layer;
  SWAT  0.289 /                     - first (top) layer;
  PRES  3608  4*  2*2 /             - second layer;
  SWAT  0.348 /                     - second layer;
  PRES  3616  4*  2*3 /             etc.
  SWAT  0.473 /
  PRES  3623  4*  2*4 /
  SWAT  0.649 /
  PRES  3633  4*  2*5 /
  SWAT  0.869 /
  PRES  3650  4*  2*6 /
  SWAT  1.0 /
/
```

```
276
277 COPY                                    We copy pressure array to the buble
278   PRES PBUB /                           pressure array.
279 /
280
281 REGALL                                  This keyword affects the following
282                                         keyword EQUALREG.
283
284                                         We specify initial parameters of the
285 EQUALREG                                fluid in the boreholes.
286   PRES 3600 FLUXNUM 111 /               -- in the PRODUCER.
287   SWAT 0.289 /
288   PBUB 3600 /
289   PRES 3650 FLUXNUM 222 /               -- in the INJECTOR.
290   PBUB 3650 /
291   SWAT 1.0 /
292 /
293
294 RPTSUM                                  This parameters are saved at every
295  PRES SWAT SOIL SGAS PBUB RS  /         report time.
296
297 RPTWELL
298   WBHP WWIR WOPR WWPR WGPR WLPR
299   WOPT WWPT WGPT WLPT WWOR WGOR/
300
301 SCHEDULE   ################### SCHEDULE section begins here ####################
302
303 REPORTS                                 No reports in the LOG-file.
304   NOTHING /
305
306 WEEKTOL                                 We use the WEEKTOL option.
307
308 WELLPROD
309 -- -------- ---- ---- -------- -------- -------- -------- -------- --------
310 --   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
311 --                        rate     rate           rate              dimens
312 -- -------- ---- ---- -------- -------- -------- -------- -------- --------
313    PRODUCER OPEN LRAT  9000      1*       1500     1E5 /
314 -- -------- ---- ---- -------- -------- -------- -------- -------- --------
315 /                                       We produce 9000 bbl/day of liquid
316                                         (oil and water) at the minimum
317                                         bottom-hole pressure 1500 psi.
318
319 WELLINJE
320 -- -------- ---- ---- -------- -------- -------- -------- -------- --------
321 --   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
322 --                        rate     rate           rate              dimens
323 -- -------- ---- ---- -------- -------- -------- -------- -------- --------
```

```
   INJECTOR OPEN WRAT  6000.      1*     4000      1E5      WATER /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
/                                      We inject 6000 bbl/day of water
                                       at the maximum bottom-hole pressure
                                       4000 psi.

VARS                                   Maximimum pressure alteration at
  PRES DMAX 500 /                      a time step is 500 psi.
/

TUNING                                 The next timestep is limited to
  0.1 25 /                             0.1 days. The maximum timestep is
                                       25 days.

TSTEP                                  We simulate the production over
  15*100 /                             1500 days reporting every 100 days.

POST      ######################## POST section begins here ######################

RPTPOST                                We save time series report for
  TIME WBHP WWIR /                     the INJECTOR.
POSTWELL
  INJECTOR /
/

RPTPOST
  NOTHING TIME WBHP WOPR WOPT WWPT
          WWOR WLPR WGPT WWPR WGOR
          WGPR /                       We save time series report for the
POSTWELL                               PRODUCER.
  PRODUCER /
/

RPTPOST                                We report pressure in the nodes
   NOTHING TIME PRES /                 489 and 491. These nodes are the first
POSTCELL                               and the last nodes of the PRODUCER. This
   489 /                               can be seen through formatted output of
   491 /                               any summary. The difference between
/                                      these pressures is the pressure drop
                                       along in the PRODUCER.

CONVERT                                We convert the output to ParaView
                                       compatible format.

END       ############################################################################
```

9.9. 7th SPE Comparative Study; case 4b (BLACKOIL)

This problem is discussed in the paper: Nghiem, L. et al. 1991 Seventh SPE Comparative Solution Project: Modelling of Horizontal Wells in Reservoir Simulation. SPE Symposium on Reservoir Simulation, 17-20 February, Anaheim, California. DOI: 10.2118/21221-MS.. See a description of the problem, input data files and visualisations under *the link*.

SPE7-CASE4B.RUN

```
################################################################################

                    7TH SPE COMPARATIVE STUDY (CASE 4B)

    The problem is from SPE-21221 paper (case 4b):
    Nghiem, L. et al. 1991. Seventh SPE Comparative Solution Project: Modelling
        of Horizontal Wells in Reservoir Simulation. SPE Symposium on Reservoir
        Simulation, 17-20 February, Anaheim, California. DOI: 10.2118/21221-MS.

    See description at: http://dx.doi.org/10.2118/21221-MS

--------------------------------------------------------------------------------

    Last update on 10.01.2017
    Program Version: 2016.C01

################################################################################


   Note:
   We type all comments in lowercase letters while all keywords and mnemonics
      must be in uppercase letters.

--------------------------------------------------------------------------------

   Note:
-- any line beginning with '!' or '--' is a comment line;
   any line not beginning with an uppercase letter outside keyword
!  instruction is also a comment line.

--------------------------------------------------------------------------------

   Note:
   Any data line must be terminated by the slash '/' sign.

```

```
--------------------------------------------------------------------------------

   Note:
   We denote by the repeated symbols the following structural elements of this
      RUN-file:
      '###############' - delimits the sections of the RUN-file.
      '<<<<<<<<<<<<<<' - designates an opening bracket.
      '>>>>>>>>>>>>>>' - designates a closing bracket.
      '^^^^^^^^^^^^^^' - designates the keyword, which affects the order of the
                         data input for all the following keywords (both in the
                         current and in the following sections).
      '==============' - merge the keywords and the comments in paragraphs.

--------------------------------------------------------------------------------

RUNSPEC   ################### RUNSPEC section begins here ######################

BLACKOIL  ^^^^^^ We switch the program to the BLACKOIL EOS module ^^^^^^^^^^^^^^

                                       We enable:
WATER                                  - water phase;
OIL                                    - oil phase;
GAS                                    - gas phase;
DISGAS                                 - gas dissolution in oil phase.

FIELD     ^^^^^^^^^^^^ We use FIELD units ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
                                       Note, this keywords affects the units
                                       in many keywords below.

                                       We enable:
ROCKCOMP                               - rock compressibility;
CAPPRES                                - capillary pressure;
MUSEGWEL                               - multisegmented wells.

OPTIONS                                Piecewise linear interpolation
  53* 1 /                              of relative permeabilities. Use this
                                       option if you want a better fit to the
                                       SPE-21221 paper.

FAST                                   We use FAST option.

GRID      ###################### GRID section begins here ######################

          The grid is specified within brackets MAKE-ENDMAKE
MAKE      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
-- cartesian                           We select Cartesian gridding
--    grid   nx  ny  nz                option and specify the number of
      CART    9   9   6  /             grid blocks along every axis.
```

```
XYZBOUND
-- xmin-xmax   ymin-ymax   zmin-zmax    We specify the domain extent in feet.
   0    1      0    1      3590 3591 /  The top of reservoir is at depth 3590
                                        feet.

DXV                                     The grid blocks length along X axis
  9*300 /                               is 300 feet.

DYV                                     Here, we specify the grid blocks
  620 400 200 100 60 100 200 400 620 /  length along Y axis.

DZV                                     Here, we specify the thickness of the
  4*20 30 50 /                          layers.

WELSPECS
-- -------- -- ---- ---- -------- -- --- -- ------
-- well        i-   j-   datum       mod    flux
--     name     ind  ind   depth      el       num
-- -------- -- ---- ---- -------- -- --- -- ------    Here, we specify
   INJECTOR 1*    1  5    1*      6* 1*  4*  222  /   both wells
   PRODUCER 1*    8  5    1*      6* SEG 4*  111  /
-- -------- -- ---- ---- -------- -- --- -- ------
/                                       The AVG well model (default option)
                                        is used for the INJECTOR. and the
                                        SEG model is used for the PRODUCER.
                                        The pipe segment associated with the
                                        INJECTOR can be farther referred to
                                        by using FLUXNUM=222, and the pipe
                                        segments asscoiated with the PRODUCER
                                        can be reffered to by FLUXNUM=111.
COMPDAT
-- -------- ---- ---- ---- ---- ---- ------ -------- -------- -- --- -- -----
-- well     i-   j-   k-   k-   mode satnum well      well       dir    rough
--     name  ind  ind  min  max                index   diameter           ness
-- -------- ---- ---- ---- ---- ---- ------ -------- -------- -- --- -- -----
   INJECTOR   1    5   6    6   OPEN  1*    243.457 /  243.457 cP*bbl/day/psi
   INJECTOR   2    5   6    6   OPEN  1*    243.457 /  =2.16e5 ft*mD
   INJECTOR   3    5   6    6   OPEN  1*    243.457 /
   INJECTOR   4    5   6    6   OPEN  1*    243.457 /
   INJECTOR   5    5   6    6   OPEN  1*    243.457 /
   INJECTOR   6    5   6    6   OPEN  1*    243.457 /
   INJECTOR   7    5   6    6   OPEN  1*    243.457 /
   INJECTOR   8    5   6    6   OPEN  1*    243.457 /
   INJECTOR   9    5   6    6   OPEN  1*    243.457 /
-- -------- ---- ---- ---- ---- ---- ------ -------- -------- -- --- -- -----
   PRODUCER   8    5   1    1   OPEN  1*    1*       0.375    3* 'X' 1* 0.001 /
   PRODUCER   7    5   1    1   OPEN  1*    1*       0.375    3* 'X' 1* 0.001 /
   PRODUCER   6    5   1    1   OPEN  1*    1*       0.375    3* 'X' 1* 0.001 /
```

```
   PRODUCER   5    5   1    1   OPEN  1*    1*       0.375    3* 'X' 1* 0.001 /
   PRODUCER   4    5   1    1   OPEN  1*    1*       0.375    3* 'X' 1* 0.001 /
   PRODUCER   3    5   1    1   OPEN  1*    1*       0.375    3* 'X' 1* 0.001 /
   PRODUCER   2    5   1    1   OPEN  1*    1*       0.375    3* 'X' 1* 0.001 /
-- -------- ---- ---- ---- ---- ---- ------ -------- -------- -- --- -- -----
/                                      Here, we specify the INJECTOR
                                       completions in the bottom layer. The
                                       well index is specified explicitly.
                                       And we specify the PRODUCER completions
                                       in the top layer.

DZWELL                                 This is for visualization.
  300 30 /

ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

EQUALS
--  --------  --------                 We equal
    PORO      0.2      /               - porosity to 0.2;
    PERMX     3000     /              - X-axis permeability to 300;
    PERMY     3000     /              - Y-axis permeability to 300;
    PERMZ     300      /              - Z-axis permeability to 30.
--  --------  --------
/

RPTGRID                                This properties are reported from the
  DEPTH PORO PERMX PERMY PERMZ /       section GRID.

PROPS     ######################## PROPS section begins here ####################

          Rock properties are specified within brackets ROCK-ENDROCK
ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

ROCKECL                                We specify the reference pressure (psia)
-- -refpres- --compr--                 and the rock compressibility (1/psi).
      3600     4.0D-6   /

ENDROCK   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The relative permeabilities are specified within brackets SAT-ENDSAT
SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

STONE2                                 We use the 2nd Stone formula for the
                                       oil relative permeability in the
                                       three-phase-flow regions.

```

```
SWOF
-- swat   krw      krow      pcow
   0.22    0        1         6.30  /  Here, we specify the water relative
   0.3     0.07     0.4       3.60  /  permeability (krw), oil relative
   0.4     0.15     0.125     2.70  /  permeability for the oil-water flows
   0.5     0.24     0.0649    2.25  /  (krow) and oil-water capillary
   0.6     0.33     0.0048    1.80  /  pressure as the functions of water
   0.8     0.65     0         0.90  /  saturation (swat).
   0.9     0.83     0         0.45  /
   1       1        0         0     /
/

SGOF
-- sgas    krg       krog   pcog
   0       0         1      0   /      Here, we specify the gas relative
   0.04    0         0.6    0.2 /      permeability (krg), oil relative
   0.1     0.022     0.33   0.5 /      permeability for the oil-gas flows
   0.2     0.1       0.1    1.0 /      (krg) and gas-oil capillary pressure
   0.3     0.24      0.02   1.5 /      as the functions of gas saturation
   0.4     0.34      0      2.0 /      (sgas).
   0.5     0.42      0      2.5 /
   0.6     0.5       0      3.0 /
   0.7     0.8125    0      3.5 /
   0.78    1         0      3.9 /
/

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The fluid properties are specified within brackets EOS-ENDEOS
EOS       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

DENSITY                                  We specify the phases densities at
--   oil       water      gas           surface conditions (lb/cft).
     45        62.14     0.0702  /

PVTW                                     We specify water properties: reference
--  refpres   bwat    comprw   viswat    pressure (refpres), form. vol. factor at
    14.7     1.0142   3.0D-6   0.96  /   refpres (bwats), viscosity (viswat) and
                                         water compressibility (comprw).

PVDG
--   pgas    bgas  visgas                We specify the properies of dry gas,
      400    5.9   0.013  /              i.e. the formation volume factor (bgas),
      800    2.95  0.0135 /              and viscosity (visgas) on pressure
     1200    1.96  0.014  /              (pgas).
     1600    1.47  0.0145 /
     2000    1.18  0.015  /
```

```
     2400    0.98  0.0155 /
     2800    0.84  0.016  /
     3200    0.74  0.0165 /
     3600    0.65  0.017  /
     4000    0.59  0.0175 /
     4400    0.54  0.018  /
     4800    0.49  0.0185 /
     5200    0.45  0.019  /
     5600    0.42  0.0195 /
/

PVTO
--   rs      poil   boil    visoil      We specify the properties of live oil,
   0.165      400   1.012    1.17  /    i.e. the oil pressure (poil),
   0.335      800   1.0255   1.14  /    formation volume factor (boil),
   0.500     1200   1.038    1.11  /    viscosity (visoil) on gas-oil ratio
   0.665     1600   1.051    1.08  /    of the solution (rs).
   0.828     2000   1.063    1.06  /
   0.985     2400   1.075    1.03  /
   1.130     2800   1.087    1.00  /
   1.270     3200   1.0985   0.98  /
   1.390     3600   1.11     0.95  /
   1.500     4000   1.12     0.94  /
   1.600     4400   1.13     0.92  /
   1.676     4800   1.14     0.91  /
   1.750     5200   1.148    0.9   /
   1.810     5600   1.155    0.89
             6000   1.1504   0.89
             6400   1.1458   0.89
             6800   1.1412   0.89
             7200   1.1367   0.89  /
/

ENDEOS    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

INIT      ######################## INIT section begins here ######################

                                     We specify pressure and water saturation
EQUALS                               for every layer:
  PRES  3600  4*  2*1 /              - first (top) layer;
  SWAT  0.289 /                      - first (top) layer;
  PRES  3608  4*  2*2 /              - second layer;
  SWAT  0.348 /                      - second layer;
  PRES  3616  4*  2*3 /              etc.
  SWAT  0.473 /
  PRES  3623  4*  2*4 /
  SWAT  0.649 /
  PRES  3633  4*  2*5 /
```

```
  SWAT  0.869 /
  PRES  3650  4*  2*6 /
  SWAT  1.0 /
/

COPY                                    We copy pressure array to the buble
  PRES PBUB /                           pressure array.
/

REGALL                                  This keyword affects the following
                                        keyword EQUALREG.

                                        We specify initial parameters of the
EQUALREG                                fluid in the boreholes.
  PRES 3600 FLUXNUM 111 /               -- in the PRODUCER.
  SWAT 0.289 /
  PBUB 3600 /
  PRES 3650 FLUXNUM 222 /               -- in the INJECTOR.
  PBUB 3650 /
  SWAT 1.0 /
/

RPTSUM                                  This parameters are saved at every
 PRES SWAT SOIL SGAS PBUB RS  /         report time.

RPTWELL
  WBHP WWIR WOPR WWPR WGPR WLPR
  WOPT WWPT WGPT WLPT WWOR WGOR/

SCHEDULE   ################### SCHEDULE section begins here ####################

REPORTS                                 No reports in the LOG-file.
  NOTHING /

WEEKTOL                                 We use the WEEKTOL option.

WELLPROD
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
--   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
--                        rate     rate            rate              dimens
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
   PRODUCER OPEN LRAT  9000      1*       1500     1E5 /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
/                                       We produce 9000 bbl/day of liquid
                                        (oil and water) at the minimum
                                        bottom-hole pressure 1500 psi.

WELLINJE
```

```
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
--   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
--                        rate     rate            rate              dimens
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
   INJECTOR OPEN WRAT  6000.      1*     4000      1E5      WATER /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
/                                      We inject 6000 bbl/day of water
                                       at the maximum bottom-hole pressure
                                       4000 psi.

VARS                                   Maximimum pressure alteration at
  PRES DMAX 500 /                      a time step is 500 psi.
/

TUNING                                 The next timestep is limited to
  0.1 25 /                             0.1 days. The maximum timestep is
                                       25 days.

TSTEP                                  We simulate the production over
  15*100 /                             1500 days reporting every 100 days.

POST      ######################## POST section begins here ####################

RPTPOST                                We save time series report for
  TIME WBHP WWIR /                     the INJECTOR.
POSTWELL
  INJECTOR /
/

RPTPOST
  NOTHING TIME WBHP WOPR WOPT WWPT
          WWOR WLPR WGPT WWPR WGOR
          WGPR /                       We save time series report for the
POSTWELL                               PRODUCER.
  PRODUCER /
/

RPTPOST                                We report pressure in the nodes
   NOTHING TIME PRES /                 489 and 495. These nodes are the first
POSTCELL                               and the last nodes of the PRODUCER. This
   489 /                               can be seen through formatted output of
   495 /                               any summary. The difference between
/                                      these pressures is the pressure drop
                                       along in the PRODUCER.

CONVERT                                We convert the output to ParaView
                                       compatible format.

```

```
372 END         ##########################################################################
```

9.10. 9th SPE Comparative Study (BLACKOIL)

This problem is discussed in the paper: Killough, J.E. 1995 Ninth SPE Comparative Solution Project: A Reexamination of Black-Oil Simulation. 13th SPE Symposium on Reservoir Simulation, San Antonio, Feb 12-15, 1995. DOI: 10.2118/29110-MS. See a description of the problem, input data files and visualisations under *the link*.

SPE9.RUN

```
################################################################################

                          9TH SPE COMPARATIVE STUDY

    The problem is from SPE-29110 paper:
    Killough, J.E. 1995 Ninth SPE Comparative Solution Project:
    A Reexamination of Black-Oil Simulation. 13th SPE Symposium on Reservoir
    Simulation, San Antonio, Feb 12-15, 1995. DOI: 10.2118/29110-MS.

    See description at: http://dx.doi.org/10.2118/29110-MS

--------------------------------------------------------------------------------

    Last update on 15.05.2015
    Program Version: 2015.C08

################################################################################

--------------------------------------------------------------------------------

   Note:
   We type all comments in lowercase letters while all keywords and mnemonics
      must be in uppercase letters.

--------------------------------------------------------------------------------

   Note:
-- any line beginning with '!' or '--' is a comment line;
   any line not beginning with an uppercase letter outside keyword
!  instruction is also a comment line.

--------------------------------------------------------------------------------

   Note:
   Any data line must be terminated by the slash '/' sign.

--------------------------------------------------------------------------------
```

```
38
39    Note:
40    We denote by the repeated symbols the following structural elements of this
41       RUN-file:
42       '###############' - delimits the sections of the RUN-file.
43       '<<<<<<<<<<<<<<<' - designates an opening bracket.
44       '>>>>>>>>>>>>>>>' - designates a closing bracket.
45       '^^^^^^^^^^^^^^^' - designates the keyword, which affects the order of the
46                           data input for all the following keywords (both in the
47                           current and in the following sections).
48       '===============' - merge the keywords and the comments in paragraphs.
49
50 ------------------------------------------------------------------------------
51
52
53
54 RUNSPEC   ################### RUNSPEC section begins here ######################
55
56 BLACKOIL  ^^^^^^ We switch the program to the BLACKOIL EOS module ^^^^^^^^^^^^^^
57
58                                      We enable:
59 WATER                                - water phase;
60 OIL                                  - oil phase;
61 GAS                                  - gas phase;
62 DISGAS                               - gas dissolution in oil phase.
63
64 FIELD     ^^^^^^^^^^^^ We use FIELD units ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
65                                      Note, this keywords affects the units
66                                      in many keywords below.
67
68 ROCKCOMP                            We enable rock compaction.
69 CAPPRES                             We enable capillary pressure.
70
71 FAST                                We use FAST option.
72
73 OPTIONS                             We enable option 54: piecewise linear
74    53* 1 /                          interpolation of relative
75                                      permeabilities. Use this option if you
76                                      want a better fit to the SPE-29110
77                                      paper. Otherwise gas  production can be
78                                      underestimated.
79
80 GRID      ###################### GRID section begins here ######################
81
82           The grid is specified within brackets MAKE-ENDMAKE
83 MAKE      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
84 -- cartesian                        We select Cartesian gridding
85 --    grid   nx  ny  nz              option and specify the number of
```

```
      CART   24  25  15   /             grid blocks along every axis.

XYZBOUND
-- xmin-xmax   ymin-ymax   zmin-zmax    We specify the domain extent in feet.
   0   10000   0   10000   9000 9010 /  The reservoir is buried at depth
                                        9000 ft. Note, xmax,ymax and zmax are
                                        redefined by the followig three
                                        keywords.

DXV                                     Every grid block extension along X-axis
  24*300 /                              is 300 ft.
DYV                                     Every grid block extension along Y-axis
  25*300 /                              is 300 ft.
DZV                                              We define the thickness
  20 15 26 15 16 14 8 8 18 12 19 18 20 50 100 /  (feet) of every layer .

ADDZCORN
 -26.44904710      1 24   4*     /      This keyword redefines corner-point
  52.89809421      2 24   4*  I- /      depths to impose the dip angle.
  52.89809421      3 24   4*  I- /
  52.89809421      4 24   4*  I- /
  52.89809421      5 24   4*  I- /
  52.89809421      6 24   4*  I- /
  52.89809421      7 24   4*  I- /
  52.89809421      8 24   4*  I- /
  52.89809421      9 24   4*  I- /
  52.89809421     10 24   4*  I- /
  52.89809421     11 24   4*  I- /
  52.89809421     12 24   4*  I- /
  52.89809421     13 24   4*  I- /
  52.89809421     14 24   4*  I- /
  52.89809421     15 24   4*  I- /
  52.89809421     16 24   4*  I- /
  52.89809421     17 24   4*  I- /
  52.89809421     18 24   4*  I- /
  52.89809421     19 24   4*  I- /
  52.89809421     20 24   4*  I- /
  52.89809421     21 24   4*  I- /
  52.89809421     22 24   4*  I- /
  52.89809421     23 24   4*  I- /
  52.89809421     24 24   4*  I- /
-------------------------------------
  52.89809421     23 24   4*     /
 -52.89809421     23 23   4*  I+ /
/

WELSPECS
-- -------- -- ---- ---- --------
```

```
-- well        i-   j-   datum
--     name         ind  ind   depth
-- -------- -- ---- ---- --------       Here, we specify 26 wells. The datum
    INJE1   1*  24   25    9110   /     depth for every well is 9110 ft.
    PROD2   1*   5    1    9110   /
    PROD3   1*   8    2    9110   /
    PROD4   1*  11    3    9110   /
    PROD5   1*  10    4    9110   /
    PROD6   1*  12    5    9110   /
    PROD7   1*   4    6    9110   /
    PROD8   1*   8    7    9110   /
    PROD9   1*  14    8    9110   /
    PROD10  1*  11    9    9110   /
    PROD11  1*  12   10    9110   /
    PROD12  1*  10   11    9110   /
    PROD13  1*   5   12    9110   /
    PROD14  1*   8   13    9110   /
    PROD15  1*  11   14    9110   /
    PROD16  1*  13   15    9110   /
    PROD17  1*  15   16    9110   /
    PROD18  1*  11   17    9110   /
    PROD19  1*  12   18    9110   /
    PROD20  1*   5   19    9110   /
    PROD21  1*   8   20    9110   /
    PROD22  1*  11   21    9110   /
    PROD23  1*  15   22    9110   /
    PROD24  1*  12   23    9110   /
    PROD25  1*  10   24    9110   /
    PROD26  1*  17   25    9110   /
-- -------- -- ---- ---- --------
/

COMPDAT
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
-- well     i-   j-   k-   k-   mode satnum   explicit   well
--     name  ind  ind  min  max                 tran   diameter
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
    INJE1    1*   1*   11  15   OPEN    1*       1*      1.0    /  The injector
    PROD2    1*   1*    2   4   OPEN    1*       1*      1.0    /  (INJE1) is
    PROD3    1*   1*    2   4   OPEN    1*       1*      1.0    /  completed in
    PROD4    1*   1*    2   4   OPEN    1*       1*      1.0    /  layers
    PROD5    1*   1*    2   4   OPEN    1*       1*      1.0    /  k=11-15.
    PROD6    1*   1*    2   4   OPEN    1*       1*      1.0    /
    PROD7    1*   1*    2   4   OPEN    1*       1*      1.0    /  All producers
    PROD8    1*   1*    2   4   OPEN    1*       1*      1.0    /  are completed
    PROD9    1*   1*    2   4   OPEN    1*       1*      1.0    /  in layers
    PROD10   1*   1*    2   4   OPEN    1*       1*      1.0    /  k=2-4.
    PROD11   1*   1*    2   4   OPEN    1*       1*      1.0    /
```

```
    PROD12   1*   1*    2   4   OPEN    1*       1*      1.0    /  Wellbores
    PROD13   1*   1*    2   4   OPEN    1*       1*      1.0    /  diameter is
    PROD14   1*   1*    2   4   OPEN    1*       1*      1.0    /  1 ft.
    PROD15   1*   1*    2   4   OPEN    1*       1*      1.0    /
    PROD16   1*   1*    2   4   OPEN    1*       1*      1.0    /
    PROD17   1*   1*    2   4   OPEN    1*       1*      1.0    /
    PROD18   1*   1*    2   4   OPEN    1*       1*      1.0    /
    PROD19   1*   1*    2   4   OPEN    1*       1*      1.0    /
    PROD20   1*   1*    2   4   OPEN    1*       1*      1.0    /
    PROD21   1*   1*    2   4   OPEN    1*       1*      1.0    /
    PROD22   1*   1*    2   4   OPEN    1*       1*      1.0    /
    PROD23   1*   1*    2   4   OPEN    1*       1*      1.0    /
    PROD24   1*   1*    2   4   OPEN    1*       1*      1.0    /
    PROD25   1*   1*    2   4   OPEN    1*       1*      1.0    /
    PROD26   1*   1*    2   4   OPEN    1*       1*      1.0    /
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
/

DZWELL                                  We define that the well are visible in
  750 75 /                              ParaView up to 750 feet above the top
                                        layer and up to 75 feet below the
                                        bottom layer.

ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

EQUALS
-- -------- --------
   PORO     0.087     4* 2*1  /         We specify porosity layer by layer.
   PORO     0.090     4* 2*2  /
   PORO     0.111     4* 2*3  /
   PORO     0.160     4* 2*4  /
   PORO     0.130     4* 2*5  /
   PORO     0.170     4* 2*6  /
   PORO     0.170     4* 2*7  /         <-- defines porosity in layer 7
   PORO     0.080     4* 2*8  /
   PORO     0.140     4* 2*9  /
   PORO     0.130     4* 2*10 /
   PORO     0.120     4* 2*11 /
   PORO     0.105     4* 2*12 /
   PORO     0.120     4* 2*13 /
   PORO     0.116     4* 2*14 /
   PORO     0.157     4* 2*15 /
-- -------- --------
/

INCLUDE                                 We load X-permeability from include
  'SPE9.PERMX' /                        file

```

```
COPY                                    We copy:
 'PERMX' 'PERMY' /                      - X perm. into Y perm;
 'PERMX' 'PERMZ' /                      - X perm. into Z perm.
/

MULTIPLY                                We multiply Z permeability by 0.01.
 'PERMZ' 0.01   /
/

RPTGRID                                 We define the output from the GRID
  PORO PERMX PERMY PERMZ /              section. These properties are saved
                                        in the file SPE9.GRID.SUM (.vtu).

PROPS     ######################## PROPS section begins here ####################

          Rock properties are specified within brackets ROCK-ENDROCK
ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

ROCKECL                                 We specify the reference pressure (psia)
-- -refpres- --compr--                  and the rock compressibility (1/psi).
     3600.     1.0D-6   /

ENDROCK   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The saturation functions are specified within brackets SAT-ENDSAT
SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

SWOF
-- swat       krw     krow      pcwo
 0.151090     0.0     1.0      400.0   /   We define the water relative
 0.151230     0.0     0.99997  359.190 /   permeability (krw), oil relative
 0.151740     0.0     0.99993  257.920 /   permeability for oil-water flows
 0.152460     0.0     0.99991  186.310 /   (krow) and water-oil capillary
 0.156470     0.0     0.999510  79.060 /   pressure as functions of water
 0.165850     0.0     0.996290  40.010 /   saturation (swat).
 0.178350     0.0     0.991590  27.930 /
 0.203350 0.000010    0.978830  20.400 /
 0.253350 0.000030    0.943730  15.550 /
 0.350000 0.000280    0.830230  11.655 /
 0.352000 0.002292    0.804277   8.720 /
 0.354000 0.004304    0.778326   5.947 /
 0.356000 0.006316    0.752374   3.317 /
 0.358000 0.008328    0.726422   1.165 /
 0.360000 0.010340    0.700470   0.463 /
 0.364395 0.015548    0.642258  -0.499 /
 0.368790 0.020756    0.584046  -1.139 /
```

```
 0.370000 0.022190    0.568020  -1.194 /
 0.380000 0.035890    0.434980  -1.547 /
 0.400000 0.069530    0.171430  -1.604 /
 0.433450 0.087900    0.125310  -1.710 /
 0.461390 0.104910    0.094980  -1.780 /
 0.489320 0.123290    0.070530  -1.860 /
 0.517250 0.143030    0.051130  -1.930 /
 0.573120 0.186590    0.024640  -2.070 /
 0.601060 0.210380    0.016190  -2.130 /
 0.656930 0.261900    0.005940  -2.260 /
 0.712800 0.318650    0.001590  -2.380 /
 0.811110 0.430920    0.000020  -2.600 /
 0.881490 0.490000    0.000000  -2.750 /
/

SGOF
-- sgas   krg  krog  pcgo
  .00     .0   1.    0   /              We define the gas relative
  .04     .0    .6    .2 /              permeability (krg), oil relative
  .1      .022  .33   .5 /              permeability (krog) and gas-oil
  .2      .1    .1   1   /              capillary pressure as functions
  .3      .24   .02  1.5 /              of gas saturation (sgas).
  .4      .34   .0   2   /
  .5      .42   .0   2.5 /
  .6      .5    .0   3   /
  .7      .8125 .0   3.5 /
  .84891 1.     .0   3.9 /
/

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The fluid properties are specified within brackets EOS-ENDEOS
EOS       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

PVTO
--     rs        poil      boil     visoil
      .0           14.69  1.0000     1.20  /   We spevify PVT properties for
      .165        400.    1.0120     1.17  /   live oil.
      .335        800.    1.0255     1.14  /
      .500       1200.    1.0380     1.11  /
      .665       1600.    1.0510     1.08  /
      .828       2000.    1.0630     1.06  /
      .985       2400.    1.0750     1.03  /
     1.130       2800.    1.0870     1.00  /
     1.270       3200.    1.0985      .98  /
     1.390       3600.    1.1100      .95  /
     1.500       4000.    1.1200      .94
```

```
                5000.    1.1189      .94  /
 /

PVDG
--    pgas       bgas   visgas
        14.69 178.08    .0125 /         We specify PVT properties for
       400.     5.4777  .0130 /         dry gas.
       800.     2.7392  .0135 /
      1200.     1.8198  .0140 /
      1600.     1.3648  .0145 /
      2000.     1.0957  .0150 /
      2400.     0.9099  .0155 /
      2800.     0.7799  .0160 /
      3200.     0.6871  .0165 /
      3600.     0.6035  .0170 /
      4000.     0.5432  .0175 /
/

PVTW                                    We specify water properties: reference
--  refpres   bwat    comprw  viswat    pressure (refpres), form. vol. factor at
     3600.   1.0034   1.0E-6   0.96 /   refpres (bwat), water compressibility
                                        (comprw)  and viscosity (viswat).

DENSITY                                 We specify the phases densities at
--   oil       water      gas           surface conditions (lb/cft).
    44.98      63.01     0.0702  /

ENDEOS    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

RPTSATTA                                We report saturation functions.
  1* 'SPE9.SAT1.CSV' 3*  1 /
RPTSATTA
  1* 'SPE9.SAT2.CSV' 3*  2 /

INIT      ######################## INIT section begins here #####################

REGALL                                  Equilibration will be applied both to
                                        grid blocks and well pipe segments.

          The initial equilibrium is specified within brackets EQL-ENDEQL
EQL       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

EQUIL
-- -------- -------- -------- -------- -------- --------
-- datum    datum     OWC      OWC      GOC      GOC
--   depth  pressure   depth    cappres   depth   cappres
-- -------- -------- -------- -------- -------- --------
```

```
    9035.    3600.    9950.     1*      8800.    1*     /
-- -------- -------- -------- -------- -------- --------

                   Using the keyword EQUIL we specify that
                   - pressure is 3600 psi at depth 9035 ft;
                   - oil-water contact is at depth 9950 ft;
                   - gas-oil contact is above the top layer (initially
                     reservoir is undersaturated).

PBVD
-- -------- --------                    We specify that initial bubble pressure
--   depth    pbub                      is 3600 psia.
-- -------- --------
     5000.   3600.   /
     9000.   3600.   /
-- -------- --------
/

ENDEQL    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

RPTSUM                                  We specify the properties which are
   PRESSURE SWAT SOIL SGAS RS /         saved at every report time (in the files
                                        SPE1.####.SUM (.vtu)).

RPTSUM                                  These properties are saved at every
   I-IJKRES J-IJKRES K-IJKRES /         report step as well.

RPTWELL
  WBHP WWIR WOPR WGPR WGOR WWIT
  WOPT WGPT WWPR WWPT WWCT  /           We specify well parameters reported
                                        at every report time (in the files
                                        SPE1.####.SUM (.vtu)).

RPTFIELD
  FGPR FOPR FWPR FGOR FWCT FWIR
  FGPT FOPT FWPT FWIT FOIP FGIP
  FWIP FOE /                            We specify field parameters reported
                                        at every report time (in the files
                                        SPE1.####.SUM (.vtu)).

SCHEDULE   ##################### SCHEDULE section begins here ####################

WEEKTOL                                 We enable option WEEKTOL (week
                                        convergence tolerances).

VARS
  PRES DMAX 500  /                      Maximal pressure change in a grid block
  PRES MAXV 10000 /                     is 500 psia, maximal pressure is 10000
```

```
/                                        psia.

REPORTS                                  No reports in the LOG-file.
   NOTHING  /

WELLINJE
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
--   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
--                        rate     rate           rate              dimens
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
    INJE1   OPEN WRAT  5000      1*     4000       1E5     WATER /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
/
                                         Using the keyword WELLINJE we
                                         specify that the well INJE1
                                         injection rate is 5000 stb/day.
                                         The maximal bottom-hole pressure is
                                         4000 psia.


WELLPROD
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
--   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
--                        rate     rate           rate              dimens
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
     PROD*  OPEN ORAT  1500      1*       1000     1E6 /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
/
                                         Using the keyword WELLPROD we
                                         specify that wells PROD*
                                         production rate is 1500 stb/day.
                                         The minimal bottom-hole pressure is
                                         1000 psia.

TSTEP
  1 2 4 8 19*15 /                           We advance simulation to 300 days.

WELLPROD
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
--   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
--                        rate     rate           rate              dimens
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
     PROD*  OPEN ORAT   100      1*       1000     1E6 /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
/
                                         Using the keyword WELLPROD we
                                         reduce production rate to 100
                                         stb/day.
```

```
470
471 TSTEP                                  We advance simulation to 360 days.
472  1 2 4 8 3*15 /
473
474 WELLPROD
475 -- -------- ---- ---- -------- -------- -------- -------- -------- --------
476 --   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
477 --                        rate     rate            rate              dimens
478 -- -------- ---- ---- -------- -------- -------- -------- -------- --------
479      PROD*  OPEN ORAT   1500      1*       1000     1E6 /
480 -- -------- ---- ---- -------- -------- -------- -------- -------- --------
481 /
482                                       Using the keyword WELLPROD we
483                                       increase production rate to 1500
484                                       stb/day.
485
486 TSTEP                                  We advance simulation to 900 days.
487  1 2 4 8 35*15 /
488
489 POST      ######################## POST section begins here ######################
490
491 CONVERT                                We convert the output to ParaView
492                                       compatible format.
493
494 RPTPOST
495   TIME FGPR FOPR FWPR FGOR FWCT FGPT
496   FOPT FWPT FWIR FWIT FOIP FGIP FWIP
497   FOE /                               We save consolidated time series report
498 POSTFLD                               for the field properties.
499 /
500
501 RPTPOST
502   NOTHING TIME WBHP WWIR WWIT /       We save consolidated time series report
503 POSTWELL                              for the injection well INJE1.
504   'INJE*'/
505 /
506
507 RPTPOST
508   NOTHING   TIME WBHP WOPR WGPR
509   WGOR WOPT WGPT WWPR WWPT WWCT /     We save consolidated time series report
510 POSTWELL                              for every production well.
511   'PROD*'/
512 /
513
514 RPTPOST
515   NOTHING TIME PRESSURE /             We save consolidated time series report
516 POSTBLOC                              for the grid block (1,1,1).
517   1  1  1 /
```

```
/

RPTPOST
  NOTHING TIME SGAS /                  We save consolidated time series report
POSTBLOC                               for the grid block (1,13,1).
  1  13  1 /
/

RPTPOST
  NOTHING TIME SWAT PRESSURE /         We save consolidated time series reports
POSTBLOC                               for the grid blocks (10,25,15),
  10  25  15 /                         (11,25 15), etc.
  11  25  15 /
  12  25  15 /
  10  25  1  /
  15  25  10 /
/

END        ###########################################################################
```

9.11. 10th SPE Comparative Study; case 1 (BLACKOIL)

This problem is discussed in the paper: Cristie M.A. et al. 2001 Tenth SPE Comparative Solution Project: A Comparison of Upscaling Techniques. SPE Res. Eval. Eng. 4(4):308-317. DOI: 10.2118/72469-PA. See a description of the problem, input data files and visualisations under *the link*.

SPE10A.RUN

```
1  ################################################################################
2
3                      10TH SPE COMPARATIVE STUDY, CASE 1
4
5      The problem is from SPE-72469-PA paper (model 1):
6      Cristie M.A. et al. 2001 Tenth SPE Comparative Solution Project: A
7         Comparison of Upscaling Techniques. SPE Res. Eval. Eng. 4(4):308-317.
8         DOI: 10.2118/72469-PA.
9
10     See description at: http://dx.doi.org/10.2118/72469-PA
11                         http://www.spe.org/web/csp/
12
13 --------------------------------------------------------------------------------
14
15     Last update on 18.05.2015
16     Program Version: 2015.C08
17
18 ################################################################################
19
20 RUNSPEC   ################### RUNSPEC section begins here ######################
21
22 BLACKOIL  ^^^^^^ We switch the program to the BLACKOIL EOS module ^^^^^^^^^^^^^^
23
24                                       We enable:
25 OIL                                   - oil phase;
26 GAS                                   - gas phase.
27
28 FIELD     ^^^^^^^^^^^^ We use FIELD units ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
29                                       Note, this keywords affects the units
30                                       in many keywords below.
31
32 ROCKCOMP                              We enable rock compaction.
33
34 FAST                                  We use FAST option.
35
36 GRID      ###################### GRID section begins here ######################
```

```

          The grid is specified within brackets MAKE-ENDMAKE
MAKE      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
-- cartesian                            We select Cartesian gridding
--    grid   nx  ny  nz                 option and specify the number of
      CART   100 1   20  /              grid blocks along every axis.
XYZBOUND
-- xmin-xmax   ymin-ymax   zmin-zmax    We specify the domain extent in feet.
   0    1      0    1      0    1 /     Actually, we specify that xmin=0 ft,
                                        ymin=0 ft, zmin=0 ft. xmax,ymax,zmax
                                        are redefined by the following keywords
                                        DXV, DYV, DZV.

DXV                                     The gid blocks extensions along X axis
  100*25 /                              are 25 ft.
DYV                                     The grid blocks extensions along Y axis
   25 /                                 are 25 ft.
DZV                                     The grid blocks extensions along Z axis
  20*2.5 /                              are 2.5 ft.

WELSPECS
-- -------- -- ---- ---- --------
-- well        i-   j-   datum         We specify two wells PRODUCER (wellhead
--     name   ind  ind   depth          in the column i=100, j=1) and
-- -------- -- ---- ---- --------       INJECTOR (wellhead in the column i=1,
   PRODUCER 1*  100  1    0.0    /      j=1). Reference depth for bottom-hole
   INJECTOR 1*    1  1    0.0    /      pressure is 0 ft.
-- -------- -- ---- ---- --------
/

COMPDAT
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
-- well     i-   j-   k-   k-   mode satnum   explicit   well
--     name  ind  ind  min  max                tran   diameter
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
   PRODUCER  1*   1*   1    20  OPEN  1*        1*        1.0   /
   INJECTOR  1*   1*   1    20  OPEN  1*        1*        1.0   /
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
/                                       Both wells are completed in every layer.
                                        Wellbore diameter is 1 ft.

DZWELL                                  We define that the wells are visible in
  50 10 /                               ParaView up to 50 feet above the top
                                        layer and up to 10 feet below the
                                        bottom layer.

ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

```

```
EQUALS
--  --------  --------
    PORO      0.2      /               We equal porosity to 0.2 in the whole
--  --------  --------                  domain.
/

PERMX                                   We load X axis permeability from
  ASCII 'SPE10A.PERMX' /                include file.

COPY                                    We copy X axis permeability into Y axis
  PERMX PERMY /                         and Z axis permeability.
  PERMX PERMZ /
/

RPTGRID                                 We report porosity and permeability
   PORO PERMX PERMY PERMZ /             from the section GRID.

PROPS     ######################## PROPS section begins here ####################

          Rock properties are specified within brackets ROCK-ENDROCK
ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

ROCKECL                                 We specify the reference pressure (psia)
-- -refpres- --compr--                  and the rock compressibility (1/psi).
      14.7     4.138e-6   /

ENDROCK   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The relative permeabilities are specified within brackets SAT-ENDSAT
SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

SGOF
-- sgas        krg        kro     not used
 0.000000   0.000000   1.000000   0.000000 /   We specify gas relative
 0.025000   0.000000   0.912530   0.000000 /   permeability (krg), oil relative
 0.050000   0.000000   0.830041   0.000000 /   permeability (kro) as tabulated
 0.075000   0.000000   0.752410   0.000000 /   functions of gas saturation
 0.100000   0.000001   0.679518   0.000000 /   (sgas).
 0.125000   0.000007   0.611238   0.000000 /
 0.150000   0.000031   0.547448   0.000000 /
 0.175000   0.000093   0.488020   0.000000 /
 0.200000   0.000232   0.432827   0.000000 /
 0.225000   0.000501   0.381737   0.000000 /
 0.250000   0.000977   0.334621   0.000000 /
 0.275000   0.001760   0.291345   0.000000 /
 0.300000   0.002980   0.251773   0.000000 /
```

```
 0.325000   0.004800   0.215767   0.000000 /
 0.350000   0.007416   0.183188   0.000000 /
 0.375000   0.011065   0.153893   0.000000 /
 0.400000   0.016028   0.127737   0.000000 /
 0.425000   0.022631   0.104573   0.000000 /
 0.450000   0.031250   0.084248   0.000000 /
 0.475000   0.042315   0.066609   0.000000 /
 0.500000   0.056314   0.051496   0.000000 /
 0.525000   0.073794   0.038746   0.000000 /
 0.550000   0.095367   0.028191   0.000000 /
 0.575000   0.121716   0.019658   0.000000 /
 0.600000   0.153590   0.012965   0.000000 /
 0.625000   0.191818   0.007925   0.000000 /
 0.650000   0.237305   0.004338   0.000000 /
 0.675000   0.291038   0.001995   0.000000 /
 0.700000   0.354093   0.000668   0.000000 /
 0.725000   0.427631   0.000103   0.000000 /
 0.750000   0.512909   0.000000   0.000000 /
 0.775000   0.611280   0.000000   0.000000 /
 0.800000   0.724196   0.000000   0.000000 /
 0.825000   0.853215   0.000000   0.000000 /
 0.850000   1.000000   0.000000   0.000000 /
/

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The fluid properties are specified within brackets EOS-ENDEOS
EOS       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

DENSITY                                 We specify the phases densities at
--   oil       water      gas           surface conditions (lb/cft).
    43.68        1*      0.0624  /

PVCDO                                   We specify dead oil properties:
--  refpres   boil    compro   visoil   reference pressure (refpres), form. vol.
    14.7      1.0      1E-7    1.0   /  factor at refpres (boil), viscosity
                                        (visoil) and water compressibility
                                        (compro).

PVDG
--   pgas    bgas    visgas             We specify the properies of dry gas,
     14.7  178.1076   0.01 /            i.e. the formation volume factor (bgas),
    400.0  173.1076   0.01 /            and viscosity (visgas) on pressure
/                                       (pgas).

ENDEOS    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

```

```
RPTSATTA                                  We report saturation functions for
1* 'SPE10A.OIL-GAS.CSV' /                 visualization in ParaView.

INIT      ######################## INIT section begins here #####################

          The initial equilibrium is specified within brackets EQL-ENDEQL
EQL       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

EQUIL
-- -------- -------- -------- -------- -------- --------
-- datum    datum     OWC      OWC      GOC      GOC
--   depth  pressure   depth    cappres   depth   cappres
-- -------- -------- -------- -------- -------- --------
     0.0     100      1000      1*       -100      1*     /
-- -------- -------- -------- -------- -------- --------
                                          Using the keyword EQUIL, we specify
                                          pressure is 100 psia at depth 0 ft.
                                          Gas-oil contact is above the top layer,
                                          thus reservoir is initially saturated
                                          with oil.

ENDEQL    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

RPTSUM                                    We save these properties distribution
   PHST PRES SOIL SGAS /                  in the domain at every report step.

RPTWELL
  WBHP WGOR WOPR WGPR
  WGIR WOPT WGIT WGPT /                   We save these parameters of each well
                                          at every report step.

SCHEDULE   ###################### SCHEDULE section begins here ####################

REPORTS
   NOTHING/                               No reports in LOG file

ILUTFILL                                  ILUT fill factor is 3.0
  3 /

ILUTDROP                                  ILUT drop factor is 0.001
  1E-3 /

<<<<<<<<<<<<<<<<<<<<<< injection/production controls <<<<<<<<<<<<<<<<<<<<<<<<<<<<

WELLPROD
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
--   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
```

```
--                        rate     rate            rate              dimens
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
   PRODUCER OPEN ORAT 1000    1*        95        1E4         /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
/                                       We specify that PRODUCER is operating
                                        under constant oil production rate
                                        1000 stb/day with minimal bottom hole
                                        with minimal bottom-hole pressure
                                        95 psia.


WELLINJE
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
--   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
--                        rate     rate            rate              dimens
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
   INJECTOR OPEN GRAT 0.24614   1*       1*       1E4      GAS       GAS     /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
/                                       We specify that INJECTOR is operating
                                        under constant gas injection rate
                                        0.24614 Mscf/day.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

TUNING                                  The inital and minimal time step is
  0.01 30 2*0.01/                       0.01 days. The maximal time step is
                                        30 days.

TSTEP
  1 2 3 4 10 20 20 30 2*45
  4*45 2*90 47*180 /                    We advance simulation to 9000 days

POST      ######################## POST section begins here ####################

CONVERT                                 We convert the output to ParaView
                                        compatible format.

RPTPOST
  TIME WBHP WGOR WOPR WGPR WOPT WGPT /  We save consolidated report for the
POSTWELL                                production well.
  PRODUCER /
/

RPTPOST
  NOTHING TIME WBHP WGIR WGIT /         We save consolidated report for the
POSTWELL                                injector.
```

```
277   INJECTOR /
278 /
279
280 END         ##############################################################################
```

9.12. 10th SPE Comparative Study; case 2 (BLACKOIL)

This problem is discussed in the paper: Cristie M.A. et al. 2001 Tenth SPE Comparative Solution Project: A Comparison of Upscaling Techniques. SPE Res. Eval. Eng. 4(4):308-317. DOI: 10.2118/72469-PA. See a description of the problem, input data files and visualisations under *the link*.

SPE10B.RUN

```
1  ################################################################################
2
3                      10TH SPE COMPARATIVE STUDY, CASE 2
4
5     The problem is from SPE-72469-PA paper (model 2):
6     Cristie M.A. et al. 2001 Tenth SPE Comparative Solution Project: A
7        Comparison of Upscaling Techniques. SPE Res. Eval. Eng. 4(4):308-317.
8        DOI: 10.2118/72469-PA.
9
10    See description at: http://dx.doi.org/10.2118/72469-PA
11                        http://www.spe.org/web/csp/
12
13 --------------------------------------------------------------------------------
14
15    Last update on 05.02.2022
16    Program Version: 2022.A01
17
18 ################################################################################
19
20 RUNSPEC   ###########################################################################
21
22 BLACKOIL  ^^^^^^ We switch the program to the BLACKOIL EOS module ^^^^^^^^^^^^^^
23
24 OIL                                    We enable oil phase.
25 WATER                                  We enable water phase.
26
27 FIELD     ^^^^^^^^^^^^ We use FIELD units ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
28                                        Note, this keywords affects the units
29                                        in many keywords below.
30
31 ROCKCOMP                              We enable rock compaction.
32
33
34 DVARCTRL                              We enable automatic time step control.
35   ON  /
36
```

```
GRID      ###################### GRID section begins here ######################

          The grid is specified within brackets MAKE-ENDMAKE
MAKE      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
-- cartesian                          We select Cartesian gridding
--    grid   nx  ny  nz                option and specify the number of
      CART   60  220 85 /              grid blocks along each axis.

XYZBOUND
-- xmin-xmax           ymin-ymax           zmin-zmax
   0    1               0    1             12000 12001 /

DXV                                     We specify grid blocks dimension along
  60*20 /                               axis Ox.
DYV                                     We specify grid blocks dimension along
   220*10 /                             axis Oy.
DZV                                     We specify grid blocks dimension along
  85*2 /                                axis Oz.

PORO                                    We load porosity distribution from
  ASCII 'SPE10.PORO' /                  SPE10.PORO. Low porosity cells will be
                                        automatically set inactive.

GETWELLS                                Wells are specified in SCHEDULE.

DZWELL                                  Visualization setup.
  100  10 /

IDTOMPI
   1 1 100000 /

ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

PERMX                                   We load permeability distribution from
  ASCII 'SPE10.PERMX' /                 external files.
PERMY
  ASCII 'SPE10.PERMY' /
PERMZ
  ASCII 'SPE10.PERMZ' /

PROPS     ###################### PROPS section begins here ####################

          Rock properties are specified within brackets ROCK-ENDROCK
ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

-- -refpres- --compr--
ROCKECL                                 We specify the reference pressure (psia)
```

```
      6000     1e-6   /                 and the pore space compressibility
                                        (1/psi).

ENDROCK   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The relative permeability is specified within brackets SAT-ENDSAT
SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

SWOF
--  swat     krw      kro
    0.200   0.0000   1.0000 /
    0.250   0.0069   0.8403 /
    0.300   0.0278   0.6944 /
    0.350   0.0625   0.5625 /
    0.400   0.1111   0.4444 /
    0.450   0.1736   0.3403 /
    0.500   0.2500   0.2500 /
    0.550   0.3403   0.1736 /
    0.600   0.4444   0.1111 /
    0.650   0.5625   0.0625 /
    0.700   0.6944   0.0278 /
    0.750   0.8403   0.0069 /
    0.800   1.0000   0.0000 /
/

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The fluid properties are specified within brackets EOS-ENDEOS
EOS       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

DENSITY
--   oil       water      gas
     53        64         1*  /

PVDO
   300    1.05   2.85  /
   800    1.02   2.99  /
   8000   1.01   3.0   /
/

PVTW
--Depth  Bw      Comp   Vw    Cv
  6000. 1.01  3.0E-6  0.3  0.0 /

ENDEOS    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

```

```
INIT      ######################## INIT section begins here #####################
          In this section, we must define the initial and boundary conditions.
             We must define the required number of parameters for every cell
             associated with the simulation.

          The initial equilibrium is specified within brackets EQL-ENDEQL
EQL       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

EQUIL
-- -------- -------- -------- -------- -------- --------
-- datum    datum     OWC      OWC      GOC      GOC
--   depth  pressure   depth    cappres   depth   cappres
-- -------- -------- -------- -------- -------- --------
    12000     6000     13000      0       1*      0     /
-- -------- -------- -------- -------- -------- --------

ENDEQL    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

RPTSUM
 PRESSURE SOIL SWAT/
RPTWELL
   WBHP WWCT WOPR WWPR WOPT WWPT WWIR WWIT /
RPTFIELD
   FWCT FOPR FWPR FOPT FWPT /

SCHEDULE   ##################### SCHEDULE section begins here ####################

WELSPECS
-- -------- -- ---- ---- -------- -- --------
-- well        i-   j-   datum       fluxnum
--     name        ind  ind   depth
-- -------- -- ---- ---- -------- -- --------
   PROD1    1*    1  1     1*     6*       /
   PROD2    1*   60  1     1*     6*       /
   PROD3    1*   60  220   1*     6*       /
   PROD4    1*    1  220   1*     6*       /
   INJE1    1*   30  110   1*     6*       /
-- -------- -- ---- ---- -------- -- --------
/

COMPDAT
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
-- well     i-   j-   k-   k-   mode satnum   explicit   well
--     name  ind  ind  min  max                 tran   diameter
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
   PROD1     1*   1*   1    85  OPEN  1*        1*       0.833  /
   PROD2     1*   1*   1    85  OPEN  1*        1*       0.833  /
```

```
   PROD3     1*   1*   1    85  OPEN  1*        1*       0.833  /
   PROD4     1*   1*   1    85  OPEN  1*        1*       0.833  /
   INJE1     30   110  1    85  OPEN  1*        1*       0.833  /
   INJE1     31   110  1    85  OPEN  1*        1*       0.833  /
   INJE1     31   111  1    85  OPEN  1*        1*       0.833  /
   INJE1     30   111  1    85  OPEN  1*        1*       0.833  /
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
/

WPIMULT
   PROD*  0.25 /                        We multiply the productivity index
   INJE1  0.25 /                        by 0.25 for every well.
/

VARS
  PRES  MAXV 12000/
  PRES  DMAX 2500 /
/

<<<<<<<<<<<<<<<<<<<< injection/production controls <<<<<<<<<<<<<<<<<<<<<<<<<<<<

WELLPROD
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
--   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
--                        rate     rate           rate              dimens
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
   PROD*    OPEN ORAT 100000    1*        4000      1E5         /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
  /

WELLINJE
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
--   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
--                        rate     rate           rate              dimens
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
   INJE1    OPEN WRAT 5000     1*       10000       1E5      WATER        /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
  /

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

TOLENG                                  Tolerances for an engineering study.

TUNING
   0.01 90  0.01 0.001 /                The maximum time step is 90 days.
                                        The next time step is 0.01 days.

TSTEP
```

```
  1 2 2 3 4 4 3*8
  4*10
  4*10
  2*20
  2*20
  45*40 /                                 We simulate 2000 days of oil recovery.

POST      ######################## POST section begins here ######################

CONVERT                                   We convert the output to ParaView
                                          compatible format.

RPTPOST
  TIME WBHP WWCT WOPR WWPR WOPT WWPT  / We save consolidated report for the
POSTWELL                                  production wells.
   'PROD*' /
/
RPTPOST
   NOTHING TIME WBHP WWIR WWIT /          We save consolidated report for the
POSTWELL                                  injector.
   'INJE*' /
/

RPTPOST
  NOTHING  TIME FWCT FOPR FWPR FOPT FWPT / We save report for the field.
POSTFLD
/

END       ###############################################################################
```

9.13. The Egg model (BLACKOIL)

This problem is discussed in the paper: Jansen J.D. et al. 2014 The egg model: a geological ensemble for reservoir simulation. Geosci. Data J. 1: 192-195. DOI:10.1002/gdj3.21. See a description of the problem, input data files and visualisations under *the link*.

EGG-MODEL.RUN

```
################################################################################

                                THE EGG MODEL

    The problem is from:
    Jansen J.D. et al. 2014 The egg model - a geological ensemble for reservoir
    simulation. Geosci. Data J. 1: 192-195. DOI: 10.1002/gdj3.21.

    See description at: http://dx.doi.org/10.1002/gdj3.21

    The include files ACTIVE.INC and mDARCY.INC can be downloaded at:
    http://dx.doi.org/10.4121/uuid:916c86cd-3558-4672-829a-105c62985ab2

--------------------------------------------------------------------------------

    Last update on 16.05.2015
    Program Version: 2015.C08

################################################################################

RUNSPEC   ################### RUNSPEC section begins here ######################

BLACKOIL  ^^^^^^ We switch the program to the BLACKOIL EOS module ^^^^^^^^^^^^^^

                                       We enable:
WATER                                  - water phase;
OIL                                    - oil phase.

METRIC    ^^^^^^^^^^^^^^^^^^^^ We use METRIC units ^^^^^^^^^^^^^^^^^^^^^^^^^^^^
                                       Note, this keywords affects the units
                                       in many keywords below.

START                                  The start date of the simulation is
  15 JUN 2011 /                        15 Jun, 2011.

FAST                                   We use FAST option.

GRID      ###################### GRID section begins here ######################
```

```
          The grid is specified within brackets MAKE-ENDMAKE
MAKE      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
-- cartesian                            We select Cartesian gridding
--    grid   nx  ny  nz                 option and specify the number of
      CART   60  60   7   /             grid blocks along every axis.

XYZBOUND                                We specify that that xmin=0 meters,
-- xmin-xmax  ymin-ymax  zmin-zmax      ymin=0 meters, the top surface of the
   0.0 1*      0.0  1*   4000 4001 /    reservoir is at 4000 meters depth. The
                                        number 4001 is redefined by the
                                        following keyword DZV.

DXV                                     Every grid block extension along X-axis
  60*8.0 /                              is 8 meters.
DYV                                     Every grid block extension along Y-axis
  60*8.0 /                              is 8 meters.
DZV                                     Every grid block extension along Z-axis
  7*4.0 /                               is 4 meters.

INCLUDE                                 We load ACTNUM property (active/inactive
  'ACTIVE.INC'/                         grid blocks from include file.

WELSPECS
-- -------- -- ---- ---- -------- -- ---- -- --------
-- well        i-   j-   datum       cros     fluxnum
--     name        ind  ind   depth     flow
-- -------- -- ---- ---- -------- -- ---- -- --------
   INJECT1  1*  5    57   1*       4*  1*  6*   1*    /  Here, we specify 12
   INJECT2  1*  30   53   1*       4*  1*  6*   1*    /  wells. The first 8
   INJECT3  1*  2    35   1*       4*  1*  6*   1*    /  wells will be injectors.
   INJECT4  1*  27   29   1*       4*  1*  6*   1*    /  The last 4 well will be
   INJECT5  1*  50   35   1*       4*  1*  6*   1*    /  producers. By default
   INJECT6  1*  8    9    1*       4*  1*  6*   1*    /  the datum depth for
   INJECT7  1*  32   2    1*       4*  1*  6*   1*    /  bottom-hole pressure
   INJECT8  1*  57   6    1*       4*  1*  6*   1*    /  is 4004 meters (the
   PROD1    1*  16   43   1*       4*  1*  6*   1*    /  depth of the top layer).
   PROD2    1*  35   40   1*       4*  1*  6*   1*    /
   PROD3    1*  23   16   1*       4*  1*  6*   1*    /
   PROD4    1*  43   18   1*       4*  1*  6*   1*    /
-- -------- -- ---- ---- -------- -- ---- -- --------
/

COMPDAT
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
-- well     i-   j-   k-   k-   mode satnum   explicit   well
--     name  ind  ind  min  max                 tran   diameter
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
```

```
   INJECT1     2*      1     7  OPEN  2*                 0.2         /
   INJECT2     2*      1     7  OPEN  2*                 0.2         /
   INJECT3     2*      1     7  OPEN  2*                 0.2         /
   INJECT4     2*      1     7  OPEN  2*                 0.2         /
   INJECT5     2*      1     7  OPEN  2*                 0.2         /
   INJECT6     2*      1     7  OPEN  2*                 0.2         /
   INJECT7     2*      1     7  OPEN  2*                 0.2         /
   INJECT8     2*      1     7  OPEN  2*                 0.2         /
   PROD1       2*      1     7  OPEN  2*                 0.2         /
   PROD2       2*      1     7  OPEN  2*                 0.2         /
   PROD3       2*      1     7  OPEN  2*                 0.2         /
   PROD4       2*      1     7  OPEN  2*                 0.2         /
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
/                                      Here, we specify that all 12 wells are
                                       completed in every layer (kmin=1,
                                       kmax=7) of the reservoir. The wellbore
                                       diameter is 0.2 meters.

DZWELL                                 We define that the well are visible in
  100 10 /                             ParaView up to 100 meters above the top
                                       layer and up to 10 meters below the
                                       bottom layer.

ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

INCLUDE                                We load heterogeneous permeability
  'mDARCY.INC'/                        distribution from include file. See also
                                       the keywords in the end of the file.

EQUALS
--  --------  --------                 We equal porosity to 0.2 in the whole
    PORO      0.2      /               domain.
--  --------  --------
/

RPTGRID                                We report porosity and permeabilities
   PORO PERMX PERMY PERMZ /            from the grid section.

PROPS     ######################## PROPS section begins here ####################

          Rock properties are specified within brackets ROCK-ENDROCK
ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

ROCKECL                                We specify the reference pressure (psia)
-- -refpres- --compr--                 and the rock compressibility (1/psi).
     400        0     /

```

```
ENDROCK   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The saturation functions are specified within brackets SAT-ENDSAT
SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

SWOF
-- swat      krw            krow
  0.1000,  0.0000e+00,  8.0000e-01  /    We define the water relative
  0.2000,  0,           8.0000e-01  /    permeability (krw), oil relative
  0.2500,  2.7310e-04,  5.8082e-01  /    permeability (krow) as tabulated
  0.3000,  2.1848e-03,  4.1010e-01  /    functions of water saturation (swat).
  0.3500,  7.3737e-03,  2.8010e-01  /
  0.4000,  1.7478e-02,  1.8378e-01  /
  0.4500,  3.4138e-02,  1.1473e-01  /
  0.5000,  5.8990e-02,  6.7253e-02  /
  0.5500,  9.3673e-02,  3.6301e-02  /
  0.6000,  1.3983e-01,  1.7506e-02  /
  0.6500,  1.9909e-01,  7.1706e-03  /
  0.7000,  2.7310e-01,  2.2688e-03  /
  0.7500,  3.6350e-01,  4.4820e-04  /
  0.8000,  4.7192e-01,  2.8000e-05  /
  0.8500,  6.0000e-01,  0.0000e+00  /
  0.9000,  7.4939e-01,  0.0000e+00  /
/

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

RPTSATTA                                We report saturation functions for
/                                       visualisation in ParaView.

          The fluid properties are specified within brackets EOS-ENDEOS
EOS       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

DENSITY                                 We specify the phases densities at
--   oil       water                    surface conditions (kg/m3).
     900       1000   /

PVCDO                                   We specify oil properties: reference
--  refpres   boil    compro  visoil    pressure (refpres), form. vol. factor at
     400      1.0      1E-5    5.0  /   refpres (boil), oil compressibility
                                        (compro)  and viscosity (visoil).


PVTW                                    We specify water properties: reference
--  refpres   bwat    comprw  viswat    pressure (refpres), form. vol. factor at
     400      1.0      1E-5    1.0  /   refpres (bwat), water compressibility
```

```
                                        (comprw)  and viscosity (viswat).

ENDEOS    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

INIT      ####################### INIT section begins here #####################

          The initial equilibrium is specified within brackets EQL-ENDEQL
EQL       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

EQUIL
-- -------- -------- -------- -------- -------- --------
-- datum    datum     OWC     OWC      GOC      GOC
--   depth  pressure   depth   cappres   depth   cappres
-- -------- -------- -------- -------- -------- --------
     4000     400      5000  /
-- -------- -------- -------- -------- -------- --------

                    Using the keyword EQUIL we specify that
                    - pressure is 400 bar at depth 4000 m;
                    - oil-water contact is at depth 5000 m (thus, at the
                      initial moment of time there is no mobile water because
                      5000 m is below the bottom layer).

ENDEQL    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

RPTSUM                                  At every report step, we report pressure
   PRESSURE SWAT SOIL /                 and saturations in the summary files.

RPTWELL                                 We also report these properties for
   WBHP WWCT WOPR WWPR WWIR /           every well.

RPTFIELD                                We also report these properties for the
   FWCT FOPR FWPR FWIR /                field.

SCHEDULE   #################### SCHEDULE section begins here ####################

WEEKTOL                                 We enable option WEEKTOL (week
                                        convergence tolerances).

REPORTS                                 No reports in the LOG-file.
   NOTHING  /

TUNING                                  We limit the first time step by 0.1 days.
  0.1 /

VARS                                    Maximal pressure change in a cell is
  PRES  DMAX 30  /                      30 bars. Maximal pressure is 700 bars.
```

```
  PRES  MAXV 700 /
/

WELLINJE
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
--   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
--                        rate     rate            rate              dimens
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
   INJECT*  OPEN WRAT  79.5      1*     420       1000     WATER /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
/                                       Using the keyword WELLINJE we
                                        specify that the injection rate is
                                        79.2 sm3/day of water for all 8
                                        injectors. The maximal bottom-hole
                                        pressure is 420 bar.

WELLPROD
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
--   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
--                        rate     rate            rate              dimens
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
     PROD*  OPEN BHP  20000      1*      395     1000 /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
/                                       Using the keyword WELLPROD we
                                        specify that 4 production wells
                                        are opertating at the constant
                                        bottom-hole pressure 395 bar.

TSTEP                                   We advance simulation to 3600 days
  120*30 /                              reporting every 30 days.

POST      ######################## POST section begins here ######################

RPTPOST
  TIME WBHP WOPR WWPR WWCT/             We save consolidated time series report
POSTWELL                                for every production well.
   'PROD*' /
/

RPTPOST
  NOTHING TIME FWCT FOPR FWPR FWIR /    We save consolidated time series report
POSTFLD                                 for the field.
/

CONVERT                                 We convert the output to ParaView
                                        compatible format.
```

```
279
280 END          ##########################################################################
```

9.14. CO_2 injection in the 10th SPE reservoir (BINMIXT)

This problem is discussed in the paper: Afanasyev A. Multiphase Compositional Modelling of CO_2 Injection Under Subcritical Conditions: The Impact of Dissolution and Phase Transitions Between Liquid and Gaseous CO_2 on Reservoir Temperature. //2013. Int. J. Greenhouse Gas Control. 9, P.731-742. DOI: 10.1016/j.ijggc.2013.01.042.. See a description of the problem, input data files and visualisations under *the link*.

EXAMPLE-H2.RUN

```
1  ##############################################################################
2
3               MODELLING CO2 INJECTION IN THE 10TH SPE RESERVOIR.
4
5      The problem is from paper:
6      Afanasyev A. Multiphase Compositional Modelling of CO2 Injection Under
7         Subcritical Conditions: The Impact of Dissolution and Phase Transitions
8         Between Liquid and Gaseous CO2 on Reservoir Temperature. //2013.
9         Int. J. Greenhouse Gas Control. 9, P.731-742.
10        DOI: 10.1016/j.ijggc.2013.01.042.
11
12     See description at: http://dx.doi.org/10.1016/j.ijggc.2013.01.042.
13
14 ------------------------------------------------------------------------------
15
16     Last update on 25.05.2015
17     Program Version: 2015.C05
18
19 ##############################################################################
20
21 ------------------------------------------------------------------------------
22
23    Note:
24    We type all comments in small letters while all keywords and mnemonics are
25       always in capital letters.
26
27 ------------------------------------------------------------------------------
28
29    Note:
30 -- any line beginning with '!' or '--' is a comment line;
31    any line not beginning with capital letter outside keyword
32 !  instruction is also a comment line.
33
```

```
--------------------------------------------------------------------------------

   Note:
   Any data line must be terminated by the slash '/' sign.

--------------------------------------------------------------------------------

   Note:
   We denote by the repeated symbols the following structural elements of this
      RUN-file:
      '###############' - delimits the sections of the RUN-file.
      '<<<<<<<<<<<<<<' - designates an opening bracket.
      '>>>>>>>>>>>>>>' - designates a closing bracket.
      '^^^^^^^^^^^^^^' - designates the keyword, which affects the order of the
                         data input for all the following keywords (both in the
                         current and in the following sections).
      '==============' - merge the keywords and the comments in paragraphs.

--------------------------------------------------------------------------------


RUNSPEC   ################### RUNSPEC section begins here ######################
          In this section, the general options and parameters of the simulation
             are specified.

                                        We enable:
SATCRT                                  - saturation functions scaling under
                                          near critical conditions;
HCROCK                                  - heat conduction in rock material.

FAST                                    We use FAST option.

GRID      ###################### GRID section begins here ######################
          In this section, we must define the geometrical and logical structures
             of the reservoir (computational grid, boundaries, faults, wells,
             point sources, etc.).

          Every new reservoir is specified within the brackets MAKE/ENDMAKE.
MAKE      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   CART 60 220 85 /
          The opening bracket MAKE has associated data, which define the type
             of the grid we want to use.
          CART data item designates that we want to use cartesian grid.
          60 220 85 data items designate that the grid has 60 cell along
             X-axis, 220 cell along Y-axis and 85 cells along Z-axis.

          ================================================================

```

```
          The following 4 keywords are used for specification the geometrical
              properties of the reservoir and the grid blocks.

XYZBOUND
-- xmin-xmax   ymin-ymax     zmin-zmax
      2*          2*         1000.0 /
          The top boundary of the reservoir is at depth=1000 m.

DXV
   60*7.62 /
          Using the keyword DXV, we specify that the length of all 60 grid
             blocks along x-axis is 7.62 meters.

DYV
   220*7.62 /
          Using the keyword DYV, we specify that the length of all 220 grid
             blocks along y-axis is 7.62 meters.

DZV
   85*0.762 /
           Using the keyword DZV, we specify that the length of all 85 grid
             blocks along z-axis is 0.762 meters.

          Note that father (when the following keyword ACTNUM is read) MUFITS
             automatically converts the cartesian grid to the corner point grid.

          ========================================================================

          The following keyword are used for setting active only the 50th
             layer of the reservoir. The cell activity is defined by the
             property ACTNUM. If ACTNUM=0 then the cell is inactive. If
             ACTNUM=1 then the cell is active. If ACTNUM=2 then the cell is
             active but its parameters are fixed, the parameters are not
             modified during the simulation. By default, all cells have the
             property ACTNUM=1.
          Firstly, we will make all cells inactive. We will set the property
             ACTNUM=0 for all cells. Secondly, we will make the 50th layer
             active. We will set the property ACTNUM=1 for all cells belonging
             to the 50th layer.

ACTNUM
   10000000*0 /
          The keyword ACTNUM can be used for specification the property ACTNUM
             for every cell. This keyword must be followed by the value of the
             property ACTNUM for every cell in the current input box. The input
             data order is defined by the ijk-indexes of the cells. The i-index
             is running in the first place, the j-index is running in the
             second place and the k-index running in the third place.
```

```
          Therefore, the keyword ACTNUM must be followed by the value 0 for
             every cell. The number of cells in the current input box is less
             than 10000000. Thus, simply typing '10000000*0' as the input data
             item we set the property ACTNUM to 0 in all cells.

BOX       ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
-- imin-imax jmin-jmax kmin-kmax
      2*        2*       50 50   /
          The input box defines a region of the grid based on the ijk-indexes
             of the grid blocks. The region is i=[imin,imax], j=[jmin,jmax],
             k=[kmin,kmax]. The input box affects the keywords used for the
             properties input, like the keyword ACTNUM. By default,
             the input box is positioned to the whole reservoir i=[1,60],
             j=[1,220], k=[1,85]. The input box is modified by the keyword BOX.
             Using this keyword we set the input box in a position when only
             the 50th layer belong the box: i=[1,60], j=[1,220], k=[50,50].

ACTNUM
   10000000*1 /
          The keyword ACTNUM can be used for the property ACTNUM specification
             only in the current input box. As the box is positioned to the 50th
             layer we can simply type '10000000*1' as the input data item to
             make the 50th layer active.

ENDBOX    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
          The keyword ENDBOX set the input box in the default position (the
             reservoir). This keyword doesn't have associated data. After this
             keyword is executed, the input box is in the position i=[1,60],
             j=[1,220], k=[1,85].

SKIP      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

CARFIN    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
-- name   imin-imax   jmin-jmax   kmin-kmax   ni   nj   nk  parent_grid
   LGR1    14   28     66   90       2*50     30   50   1  /
          The LGRs of cartesian grid type are specified by the keyword CARFIN.
             The LGR is specified by the ijk-indexes. The data associated with
             this keyword are as follows:
             item 1 (name) - the name of LGR. Father we can reference to the
                LGR by its name 'LGR1';
             items 2,3 (imin-imax) - the minimal and the maximal i-index of the
                parent grid, to which the LGR is applied;
             items 4,5 (jmin-jmax) - the minimal and the maximal j-index of the
                parent grid, to which the LGR is applied;
             items 6,7 (kmin-kmax) - the minimal and the maximal k-index of the
                parent grid, to which the LGR is applied;
             item 8 (ni) - the number of cells in the LGR along the
                i-indexation axis (ni>imax-imin);
```

```
            item 9 (nj) - the number of cells in the LGR along the
               j-indexation axis (nj>jmax-jmin);
            item 10 (nk) - the number of cells in the LGR along the
               k-indexation axis (nk>kmax-kmin).

ENDFIN    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
          After the keyword CARFIN is executed, MUFITS automatically creates
             the LGR and makes this grid active. The keyword ENDFIN makes the
             parent grid of the reservoir active. This keyword doesn't have
             associated data.
          The grid activity affect a lot of input data, e.g. the following
             keyword BOUNDARY, which is applied only to the cells nested in the
             active grid.

ENDSKIP   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          ========================================================================

          The following two keywords control the local grid refinement (LGR).
             The LGR can be disabled by commenting out these two keywords.


          ========================================================================

BOUNDARY
-- FLUXNUM   6 default    dire      5 default   option
--   ID       values      ction      values
     111      6*           J-         5*        INFTHIN /  -- left  boundary
     222      6*           J+         5*        INFTHIN /  -- right boundary
/
          We specify the left (y=0 m) and the right (y=1676.4 m) boundary of
             the reservoir by the keyword BOUNDARY. This keyword has associated
             data of the tabular form. The first data row describes the left
             boundary, the second data row describes the right boundary.
          Every row contains the following data items:
             item 1 - the boundary ID. For every boundary MUFITS automatically
                creates auxiliary cells, which are marked out by the property
                FLUXNUM=ID. By default, the property FLUXNUM=0;
             items 2-7 - imin,imax,jmin,jmax,kmin,kmax the region of the grid,
                for which the boundary is created;
             items 8-12 - the direction tags (I-,I+,J-,J+,K-,K+). This data
                items define the sides of the reservoir region, for which the
                auxiliary cells are created;
             item 13 - option. The record INFTHIN targets MUFITS to create
                infinitely thin boundary cells.
          Father the left boundary is marked out by the property FLUXNUM=111.
             The right boundary is marked out by the property FLUXNUM=222.

```

```
          ================================================================

ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

REGALL    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
          The switch REGALL does not have associated data. This keyword targets
             MUFITS to perform operations on array for all cells in the
             simulation but not only for the cells in the current input box.
             This keyword affects the data processing by the following keyword
             EQUALREG, which is related to the auxiliary cells of the boundary
             condition. It also affects the boundary conditions specification
             in the section INIT.

EQUALS                                  We equal rock heat conductivity
  HCONDCFX  2. /                        to 2.0 W/m/K in the whole domain.
  HCONDCFY  2. /
/


EQUALREG
--          new
-- property value    marker    marker value
-- -------- -------- --------- ---------
   ACTNUM   2        FLUXNUM   111       /
   ACTNUM   2        FLUXNUM   222       /
-- -------- -------- --------- ---------
/
          We must have the auxiliary cells created by MUFITS for the boundary
             conditions under the fixed state conditions (the Dirichlet boundary
             conditions). The parameters of these cells must not be modified
             during the simulation. This behavior is engaged by setting the
             property ACTNUM=2 for all auxiliary cells. The specification is
             done by the keyword EQUALREG which has tabular associated data.
             Every new row of the table engages a new data transformation. In
             every row
                item 1 - the mnemonic of the property, which is modified;
                item 2 - the new value of the property;
                item 3 - the marker property;
                item 4 - the value of the marker property.
             The EQUALREG operation can be described as follows: 'property' =
                'new value' for all cells, which have the property 'marker'
                 equal to the 'marker value'.

          ================================================================

          In this paragraph, we load the petrophysical properties from files.

POROMIN
```

```
   0.01 /
          The keyword POROMIN targets MUFITS to increase all the properties
             PORO to the value 0.01 if they are below this value. By the default
             option, the minimal porosity is 0.001. This option eliminates the
             cells with small pore volume from the simulation and consequently
             makes the simulation more stable.

PORO
   ASCII 'EXAMPLE-H2.PORO'  /
          The keyword PORO can be tuned up to load the porosity from an include
             file. If the 1st data item of this keyword is the mnemonic ASCII
             or BINARY then the loading is invoked. In this case, the 2nd data
             item is file name, from which the porosity is loaded. The mnemonic
             ASCII signifies that the porosity file is a formatted file. The
             mnemonic BINARY signifies that the the porosity file is a binary
             file.
          According to the earlier specification of the arrays loading mode
             (the keyword LOADING), the file EXAMPLE-H2.PORO must contain
             the porosity value for every cell of the reservoir. The values are
             ordered in that i-index is running in the 1st place, j-index
             is running in the 2nd place, k-index is running in the 3rd
             place.

PERMX
   ASCII 'EXAMPLE-H2.PERMX' /
          The syntax of the keyword PERMI is identical to the syntax of the
             keyword PORO except that the keyword is for loading the x-direction
             permeabilities. The permeability dimension is mD (millidarcy).

PERMY
   ASCII 'EXAMPLE-H2.PERMY' /
          The syntax of the keyword PERMJ is identical to the syntax of the
             keyword PORO except that the keyword is for loading the y-direction
             permeabilities. The permeability dimension is mD (millidarcy).

PERMZ
   ASCII 'EXAMPLE-H2.PERMZ' /
          The syntax of the keyword PERMK is identical to the syntax of the
             keyword PORO except that the keyword is for loading the z-direction
             permeabilities. The permeability dimension is mD (millidarcy).

          ============================================================================

RPTRES
   MPINUM PORO PERMX PERMY PERMZ  /
          We specify the properties we want to save in the file
             'EXAMPLE-H2.GRID.SUM'.

```

```
PROPS     ###################### PROPS section begins here ####################
          In this section, we must define the thermophysical properties of
             binary mixture, host rock as well as the relative permeability
             curves.

LOADEOS
   '../../INCLUDE/CO2H2O_V3.0.EOS' /
          The keyword LOADEOS is used for loading the mixture properties from
             the EOS-file. This keyword has as associated data the system path
             to the file.

          The rock properties for every region ROCKNUM are specified within
             the brackets ROCK/ENDROCK.
ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /
          The opening bracket ROCK has one associated data item, which is the
             ROCKNUM region ID. By default, all cells belong to the region
             ROCKNUM=1.

ROCKDH
-- grain density        heat capacity
--  [kg/m^3]              [kJ/kg/K]
      2500                   1.0    /
          The easiest way for the rock properties specification is by using the
             keyword ROCKDH.

ENDROCK   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The relative permeabilities for every region SATNUM are specified
             within the brackets SAT/ENDSAT.
SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /
          The opening bracket SAT has one associated data item, which is the
             SATNUM region ID. By default, all cells belong to the region
             SATNUM=1.

SATTAB
--  sH2O     krH2O      krCO2
    0.2     0.000000   1.000000 /
    0.3     0.000316   0.737758 /   The  Brooks and Corey curves
    0.4     0.005056   0.499535 /
    0.5     0.025600   0.302400 /      Sl_min=0.2
    0.6     0.080908   0.155832 /      Sg_min=0.05
    0.7     0.197530   0.061728 /
    0.8     0.409600   0.014400 /
    0.9     0.758834   0.000572 /
    0.95    1.000000   0.000000 /
/
```

```
          The keyword SATTAB is followed by the tabular data for the relative
             permeabilites.
             The 1st column is the liquid phase saturation, which is water
             (sH2O). The 2nd column is the liquid phase relative
             permeability (krH2O). The 3rd column is the gaseous phase
             relative permeability, which is the relative permeability
             of supercritical CO2 phase.

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

PHASES
-- name     pres    enth    composition
--          [MPa] [kJ/mol]  CO2     H2O
   LH2O     1.0      5.0    0.0     1.0  /        - H2O rich phase
   SCO2     10.0    10.0    1.0     0.0  /        - CO2 rich phase
/
          The keyword PHASES defines new MNEMONICS for data output. This keyword
             is followed by tabular data and every line of these data defines an
             individual phase. The 1st column is the phase name (not more
             than 4 characters). The following data items define typical
             thermodynamic parameters of the phase. The 2nd column is
             pressure, the 3rd is the enthalpy, the 4th and 5th are the molar
             composition of binary mixture.
          We define two phases. The 1st is the liquid water (LH2O). The 2nd is
             the supercritical CO2 phase.

RPTSATTA
/

INIT      ####################### INIT section begins here ####################
          In this section, we must define the initial and boundary conditions.
             We must define the required number of parameters for every cell
             associated with the simulation.

OPERAREG
-- modified argument          marker    opera    paramet  paramet  paramet
-- property property marker   value       tion      er1      er2      er3
-- -------- -------- --------- --------- -------- -------- -------- ---------
   PRES     YCOORD   FLUXNUM  1         MULTA    7.5     -1.789549034E-3 /
-- -------- -------- --------- --------- -------- -------- -------- ---------
/
          It is easy to calculate that if pressure is 7.5 MPa at the boundary
             y=0 m, and pressure 4.5 MPa at the boundary y=1676.4 m then the
             pressure distribution must be the following function:

                 PRESSURE=7.5-1.789E-3*YCOORD    [MPa]

          To specify this distribution of pressure we use the keyword OPERAREG.
```

```
            This keyword has tabular associated data. The keyword data items
            are as follows:
            item 1 - the modified property mnemonic;
            item 2 - the argument property mnemonic;
            item 3 - the marker mnemonic;
            item 4 - the marker value;
            item 5 - the data transformation mnemonic. This argument specifies,
               which operation is applied to the data;
            items 6-8 - the parameters of the operation.
         If the operation is MULTA then MUFITS do the following:

             'modified property'='parameter1'+'parameter2'*'argument property'
             for all cells marked out as 'marker' is equal to 'marker value'.

EQUALREG
-- -------- -------- --------- ---------
   COMP1T   0.0      FLUXNUM   1         /
   TEMP     295.15                       /
-- -------- -------- --------- ---------
   PRES     4.5      FLUXNUM   222       /
   COMP1T   0.0                          /
   TEMP     295.15                       /
-- -------- -------- --------- ---------
   PRES      7.5      FLUXNUM   111       /
   COMP1T   1.0                          /
   TEMP     320.15                       /
-- -------- -------- --------- ---------
/
         The other initial condition except the conditions for pressure as well
            as the boundary conditions are specified by the keyword EQUALREG.
            We specify that:
            data row 1 - the reservoir is initially saturated by pure H2O;
            data row 2 - the initial reservoir temperature is 295.15 K;
            data row 3 - the pressure at the right boundary (y=1676.4 m)
               is 4.5 MPa;
            data row 4 - there is pure H2O at the right boundary;
            data row 5 - the temperature at the right boundary is
               295.15 K;
            data row 6 - the pressure at the left boundary (y=0 m) is 7.5 MPa;
            data row 7 - pure CO2 is injected through the left boundary;
            data row 8 - the temperature of the injected CO2 is 320.15 K.
         Note that the numbers 111 and 222 are the boundary markers, which
            are specified by the keyword BOUNDARY in the section GRID.

         ================== output specification ============================

RPTSOL
   PHST PRES TEMP SAT#LH2O SAT#SCO2 COMP1T ENTHT COMP2T /
```

```
466             The properties, which we want to save in summary files during the
467                simulation, can be specified using the keyword RPTSOL.
468             We want to save the properties DEPTH, PHST, PRES etc.
469             Note that SAT#LH2O is the saturation of the phase LH2O which was
470                defined using keyword PHASES in the section PROPS.
471             Note that the keyword RPTSOL has an arbitrary number (up to 10000)
472                of associated data items. The data items are the property
473                mnemonics. The mnemonics with the special meaning are as follows:
474                   CLEAN - targets MUFITS to delete all data items in the list
475                      RPTSOL;
476                   INSERT - the subsequent mnemonics are added to the list RPTSOL
477                     (default option);
478                   DELETE - the subsequent mnemonics are deleted from the list
479                      RPTSOL.
480             The input of the same type is for the keywords RPTGRID, PHYSICS and
481                REPORTS in this RUN-file.
482
483 SCHEDULE   ##################### SCHEDULE section begins here ####################
484
485 VARS                                    Maximal pressure alteration is 3.0 MPa.
486   PRES  DMAX 3.0 /
487   DEN   DMAX 1000 /
488 /
489
490 REPORTS                                 No reports in LOG file.
491    NOTHING /
492
493 ILUTFILL                                We change default linear solver
494   5.0 /                                 settings. Ilut fill factor is equal to
495 ILUTDROP                                5. Ilut drop is equal to 0.001.
496   0.001 /
497
498 TUNING
499 -- timestep setup
500 -- limit   max    next   min
501     0.001      31     10   0.00001   /
502             The timestep control can be tuned up using the keyword TUNING. The
503                data items related to the timestep control are:
504                   item 1 - the timestep limit [days]. The next timestep can't
505                      be greater than this value;
506                   item 2 - the maximal timestep [days];
507                   item 3 - the explicitly specified next timestep [days];
508                   item 4 - the minimal timestep [days].
509
510             ================================================================
511
512 TSTEP
513    50*365.25 /
```

```
514             By the keyword TSTEP we define the time of output. The year duration
515                in days is estimated as 365.25 days. We want to simulate the flow
516                over 50 years and save the simulation results after every
517                year in the simulation schedule. This behavior is specified by
518                the argument '50*365.25'. Thus, the output will be saved in the
519                summary files at times: 365.25, 730.50, 1095.75, etc. days.
520             When the simulator comes to this record TSTEP in the RUN-file
521                it performs simulation and produces output for the described
522                times.
523
524 POST      ######################## POST section begins here ######################
525
526 CONVERT                                  We convert the output to ParaView
527                                          compatible format.
528
529 END        ###########################################################################
```

9.15. Benchmark study for CO_2 storage (BINMIXT)

This problem is discussed in the paper: Class H. et al. 2009 A benchmark study on problems related to CO_2 storage in geological formations. Comput. Geosci. 13(4):409-434. DOI:10.1007/s10596-009-9146-x. See a description of the problem, input data files and visualisations under *the link*.

EXAMPLE-H3.RUN

```
################################################################################

                          BENCHMARK STUDY FOR CO2 STORAGE

    This is problem 3.1 from
    Class H. et al. 2009 A benchmark study on problems related to CO2 storage
       in geological formations. Comput. Geosci. 13(4):409-434.
        DOI:10.1007/s10596-009-9146-x

    See description at: http://dx.doi.org/10.1007/s10596-009-9146-x

--------------------------------------------------------------------------------

    Last update on 16.11.2015
    Program Version: 2015.G01

################################################################################


RUNSPEC   ################### RUNSPEC section begins here ######################

FAST                                  We use FAST option.

STARTTIM                              The initial time is -365.25 days=
   -365.25 /                          = -1 year.

GRID      ###################### GRID section begins here ######################

          The grid is specified within brackets MAKE-ENDMAKE
MAKE      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
-- corner point                       We select Corner-point gridding
--    grid   nx  ny  nz                option and specify the number of
     CORNER  78  78  9   /            grid blocks along every axis.

INCLUDE                               We load the grid in corner-point format
   'EXAMPLE-H3.GRDECL' /              from include file.

FAULTS
```

```
-- --name--  ------box------  -direction tag-    We specify fault plane
   FAULT1    40 78 39 39 1 9   J+             /  named FAULT1.
-- --------  ---------------  ---------------
/

MULTFLT
-- --name--  -multiplier-               We multiply transmissibility across
   FAULT1     0.0         /             the fault plane by factor 0.0.
-- --------  ------------
/

BOUNDARY
-- --name--  ------box------  --direction tags--  --type--  --mode--
    123      6*                 I- I+ J- J+ 2*    SAMESIZE  ACTBASED /
    223      40 78 39 39 1 9    J+ 5*             SAMESIZE  ALL      /
    223      40 78 40 40 1 9    J- 5*             SAMESIZE  ALL      /
-- --------  ---------------  ------------------  --------  --------
/
               Using the keyword BOUNDARY
               - we encompass the reservoir with auxiliary grid blocks
                 marked as FLUXNUM=123. Further, we use these blocks for
                 modelling fluid influx from the aquifer;
               - we create auxiliary grid blocks on either side of the fault
                 plane marked as FLUXNUM=223. Further, we use these grid blocks
                 to fix pressure at the fault plane.

WELSPECS
-- -------- -- ---- ---- --------
-- well        i-   j-   datum
--     name     ind  ind   depth
-- -------- -- ---- ---- --------       We specify well INJECTOR in the column
   INJECTOR 1*  30   48  /              i=30, j=48.
-- -------- -- ---- ---- --------
/

COMPDAT
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
   INJECTOR   1*  1*   6    9   OPEN  1*        1*        0.15    /
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
/                                    We specify that the well is completed
                                     in layer k=6-9. The wellbore diameter
                                     is 0.15 meters.

DZWELL                               We define that the well is visible in
  250 25 /                           ParaView up to 250 meters above the top
                                     layer and up to 25 meters below the
                                     bottom layer.

```

```
ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

PORO                                    We load porosity from include file.
   ASCII 'EXAMPLE-H3.PORO' /

PERMX                                   We load X-axis permeability from
   ASCII 'EXAMPLE-H3.PERM' /            include file.

COPY
-- --from-- ---to--- ------box------    We copy X-axis permeability into
   PERMX    PERMY    /                  Y-axis and Z-axis permeability,
   PERMX    PERMZ    /
-- -------- -------- ---------------
/

REGALL    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
                                        The keyword REGALL enable application
                                        operations on arrays both to domain and
                                        boundary grid blocks.
EQUALREG
--          new
-- property value    marker    marker value     We set FIPNUM=11 in all
-- -------- -------- --------- ---------        boundary grid blocks.
   FIPNUM   11       FLUXNUM   123       /
   FIPNUM   11       FLUXNUM   223       /
   FIPNUM   11       FLUXNUM   223       /
-- -------- -------- --------- ---------
/

RPTGRID
   DEPTH XCOORD YCOORD PORO
   PERMX PERMY PERMZ
   I-IJKRES J-IJKRES K-IJKRES /         We report these properties from the
                                        GRID section.

PROPS     ####################### PROPS section begins here ####################

LOADEOS
   '../../INCLUDE/CO2H2O_V3.0.EOS' /    We load CO2-H2O mixture properties file.

          Rock properties are specified within brackets ROCK-ENDROCK
ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

ROCKDH                                  We specify that
  2500  1.0 /                           rock density is 2500 kg/m3,
                                        rock heat capacity is 1.0 kJ/kg/K

```

```
ENDROCK   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The relative permeabilities are specified within brackets SAT-ENDSAT
SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

SATTAB
--  sH2O     krH2O      krCO2
    0.2     0.000000   1.000000 /       We specify H2O-rich phase relative
    0.3     0.000316   0.737758 /       permeability (krH2O) and CO2-rich phase
    0.4     0.005056   0.499535 /       relative permeability (krCO2) as
    0.5     0.025600   0.302400 /       function of water saturation (sH2O).
    0.6     0.080908   0.155832 /       These are the Brooks and Corey curves
    0.7     0.197530   0.061728 /       with Sl_min=0.2, Sg_Min=0.05.
    0.8     0.409600   0.014400 /
    0.9     0.758834   0.000572 /
    0.95    1.000000   0.000000 /
/

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

RPTSATTA                                We report saturation functions
/                                       for visualization in ParaView.

PHASES
-- name   pres  enth  CO2 H2O                     We define two phases for
   LH2O   1.0    5.0  0.0 1.0  /                  output. LH2O - H2O-rich phase,
   SCO2   10.0  10.0  1.0 0.0  /                  SCO2 - CO2-rich phase.
/                                                 The critical saturations
                                                  are consistent with the
                                                  SATTAB keyword.

INIT      ###################### INIT section begins here #####################

OPERAREG
-- modified argument           marker    opera    paramet  paramet  paramet
-- property property marker    value       tion      er1      er2      er3
-- -------- -------- --------- --------- -------- -------- -------- ---------
   PRES     DEPTH    SATNUM    1         MULTA    0.31     0.01              /
   TEMPC    DEPTH    SATNUM    1         MULTA    10       0.03              /
-- -------- -------- --------- --------- -------- -------- -------- ---------
/
                                        Using the keyword OPERAREG, we specify
                                        initial approximation of hydrostatic
                                        equilibrium. We specify:
                                        -   PRES=0.31+0.01*DEPTH MPa;
                                        -   TEMP=10+0.03*DEPTH C.

```

```
183
184 EQUALREG
185 --          new
186 -- property value    marker    marker value
187 -- -------- -------- --------- ---------       Initial CO2 molar fraction
188    COMP1T    0.0      SATNUM    1        /     is equal to 0 in the whole
189 -- -------- -------- --------- ---------       domain.
190 /
191
192 EQUALS                                  We set FIPNUM=10 in the domain grid
193   FIPNUM 10 /                           blocks.
194 /
195
196 RPTSUM
197   PRESSURE TEMPC ENTHT COMP1T COMP2T
198                    SAT#LH2O SAT#SCO2
199           I-IJKRES J-IJKRES K-IJKRES /  We save these properties distribution
200                                         in the summary files at every report
201                                         step.
202 RPTFIP                                  For every fluid-in-place region we
203   MASS#1 MS1#SCO2 MS1#LH2O /            save in summaries total CO2 mass,
204                                         CO2 mass in CO2-rich phase and CO2
205                                         mass in H2O-rich phase(dissolved in
206                                         water).
207 RPTWELL                                 For the well we report CO2 injection
208    WMIR#1 WMIT#1 WBHP WBHTC /           rate and total, bottom-hole pressure
209                                         and temperature.
210
211 SCHEDULE   ##################### SCHEDULE section begins here ####################
212
213 WEEKTOL                                 We enable WEEKTOL modelling option
214                                         (week convergence tolerance).
215
216 ILUTFILL                                ILUT fill factor is 5.0.
217    5 /
218 ILUTDROP                                ILUT drop factor is 0.001.
219   1E-3 /
220
221           ======================================================================
222
223 TUNING                                  The next time step is 10 days. Maximal
224     1*  100  10 /                       time step is limited by 100 days.
225
226 TSTEP                                   We simulate without CO2 injection for
227    365.25 /                             1 year to evaluate initial hydrostatic
228                                         equilibrium. The flow is simulated both
229                                         in the domain and boundary grid blocks.
230           ======================================================================
```

```
231
232 EQUALREG
233 --          new
234 -- property value    marker    marker value
235 -- -------- -------- --------- ---------      Now, we fix parameters (pressure
236    ACTNUM   2        FLUXNUM   123       /    and temperature) at all boundary
237    ACTNUM   2        FLUXNUM   223       /    grid blocks.
238 -- -------- -------- --------- ---------
239 /
240
241 WELLINJE
242 -- -------- ---- ---- -------- -------- -------- -------- -------- --------
243 --   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
244 --                        rate     rate            rate              dimens
245 -- -------- ---- ---- -------- -------- -------- -------- -------- --------
246    INJECTOR OPEN MASS     0      1296      50    /
247 -- -------- ---- ---- -------- -------- -------- -------- -------- --------
248 /
249                                      Using the keyword WELLINJE, we start
250                                      CO2 injection. Injection rate is
251                                      1296 t/day. Maximal bottom hole
252                                      pressure is 50 MPa.
253
254 EQUALNAM                             We specify that the injected CO2
255   PRES    25. INJECTOR /             temperature is 80 C at 25 MPa.
256   TEMPC   80           /
257   COMP1T  1.0          /
258   COMP2T  0.0          /
259 /
260
261 INITNAM                             We recalculate primary variables based
262    INJECTOR /                        on the secondary variables in the stock
263                                      tank associated with the INJECTOR.
264
265 TUNING                              The next time step is 1 day. The
266     1. 92.  1. /                     maximal time step is 92 days.
267
268 TSTEP                               We advance simulation to 25 years
269    25*365.25 /                       reporting every year.
270
271           ================================================================
272
273 WELLSTOP                            Now, we stop the CO2 injection.
274    INJECTOR /
275 /
276
277 TUNING                              The next time step is 1 day. The
278     1.  92  1. /                     maximal time step is 92 days.
```

```
279
280 TSTEP                                  We advance simulation for 25 years
281    25*365.25 /                         reporting every year.
282
283 POST      ######################## POST section begins here ####################
284
285 RPTPOST
286    TIMEY MASS#1 MS1#SCO2 MS1#LH2O /    We save consolidated time series report
287 POSTFPCE                               for the fluid-in-place region 10.
288    10 /
289 /
290
291 RPTPOST
292    NOTHING
293    TIMEY  WMIR#1 WMIT#1 WBHP WBHTC /   We save consolidated time series report
294 POSTWELL                               for the well.
295   '*' /
296 /
297
298 CONVERT                                We convert the output to ParaView
299                                         compatible format.
300
301 END        ############################################################################
```

9.16. Benchmark study for CO_2 storage (GASSTORE)

This problem is discussed in the paper: Class H. et al. 2009 A benchmark study on problems related to CO_2 storage in geological formations. Comput. Geosci. 13(4):409-434. DOI:10.1007/s10596-009-9146-x. See a description of the problem, input data files and visualisations under *the link*.

EXAMPLE-H3-GASSTORE.RUN

```
1  ################################################################################
2
3                            BENCHMARK STUDY FOR CO2 STORAGE
4
5      This is problem 3.1 from
6      Class H. et al. 2009 A benchmark study on problems related to CO2 storage
7         in geological formations. Comput. Geosci. 13(4):409-434.
8          DOI:10.1007/s10596-009-9146-x
9
10     See description at: http://dx.doi.org/10.1007/s10596-009-9146-x
11
12 --------------------------------------------------------------------------------
13
14     Last update on 24.10.2015
15     Program Version: 2015.F01
16
17 ################################################################################
18
19
20 RUNSPEC   ################### RUNSPEC section begins here ######################
21
22 FAST                                  We use FAST option.
23
24 STARTTIM                             The initial time is -365.25 days=
25    -365.25 /                        = -1 year.\
26
27 GASSTORE  ^^^^^^ We switch the program to the GASSTORE EOS module ^^^^^^^^^^^^^^
28
29                                      We enable
30 WATER                               - water phase
31 CO2-SEQ                             - CO2 component (and gas phase)
32 DISGAS                              - CO2 dissolution in water
33 VAPWAT                              - water vapour in the gas phase
34
35 ISOTHERM  We use the Isothermal modelling option
36
37 METRIC    ^^^^^^^^^^^^ We use METRIC units ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
38
```

```
GRID      ###################### GRID section begins here ######################

          The grid is specified within brackets MAKE-ENDMAKE
MAKE      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
-- corner point                          We select Corner-point gridding
--    grid   nx  ny  nz                  option and specify the number of
     CORNER  78  78  9   /               grid blocks along every axis.

INCLUDE                                  We load the grid in corner-point format
   'EXAMPLE-H3.GRDECL' /                 from include file.

FAULTS
-- --name--  ------box------  -direction tag-    We specify fault plane
   FAULT1    40 78 39 39 1 9   J+             /  named FAULT1.
-- --------  ---------------  ---------------
/

MULTFLT
-- --name--  -multiplier-                 We multiply transmissibility across
   FAULT1     0.0         /               the fault plane by factor 0.0.
-- --------  ------------
/

BOUNDARY
-- --name--  ------box------  --direction tags--  --type--  --mode--
    123      6*                 I- I+ J- J+ 2*    SAMESIZE  ACTBASED /
    223      40 78 39 39 1 9    J+ 5*             SAMESIZE  ALL      /
    223      40 78 40 40 1 9    J- 5*             SAMESIZE  ALL      /
-- --------  ---------------  ------------------  --------  --------
/
               Using the keyword BOUNDARY
               - we encompass the reservoir with auxiliary grid blocks
                 marked as FLUXNUM=123. Further, we use these blocks for
                 modelling fluid influx from the aquifer;
               - we create auxiliary grid blocks on either side of the fault
                 plane marked as FLUXNUM=223. Further, we use these grid blocks
                 to fix pressure at the fault plane.

WELSPECS
-- -------- -- ---- ---- --------
-- well        i-   j-   datum
--     name     ind  ind   depth
-- -------- -- ---- ---- --------       We specify well INJECTOR in the column
   INJECTOR 1*  30   48  /              i=30, j=48.
-- -------- -- ---- ---- --------
/

COMPDAT
```

```
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
   INJECTOR   1*  1*   6    9   OPEN  1*        1*        0.15    /
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
/                                    We specify that the well is completed
                                     in layer k=6-9. The wellbore diameter
                                     is 0.15 meters.

DZWELL                               We define that the well is visible in
  250 25 /                           ParaView up to 250 meters above the top
                                     layer and up to 25 meters below the
                                     bottom layer.

ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

PORO                                 We load porosity from include file.
   ASCII 'EXAMPLE-H3.PORO' /

PERMX                                We load X-axis permeability from
   ASCII 'EXAMPLE-H3.PERM' /         include file.

COPY
-- --from-- ---to--- ------box------ We copy X-axis permeability into
   PERMX    PERMY    /               Y-axis and Z-axis permeability,
   PERMX    PERMZ    /
-- -------- -------- ---------------
/

RPTGRID
   DEPTH XCOORD YCOORD PORO
   PERMX PERMY PERMZ
   I-IJKRES J-IJKRES K-IJKRES /      We report these properties from the
                                     GRID section.

PROPS     ####################### PROPS section begins here ####################

SATOPT                               We change the method of interpolation
  2* 1 /                             of saturation functions beyond the
                                     the critical values.

INCLUDE                              We load the porperties of CO2 from the
  'CO2TAB.METRIC.INC' /              include file.

          Rock properties are specified within brackets ROCK-ENDROCK
ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

ROCKDH                               We specify that
  2500  1.0 /                        rock density is 2500 kg/m3,
```

```
                                        rock heat capacity is 1.0 kJ/kg/K

ENDROCK   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The relative permeabilities are specified within brackets SAT-ENDSAT
SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

SATTAB
--  sH2O     krH2O      krCO2
    0.2     0.000000   1.000000 /       We specify H2O-rich phase relative
    0.3     0.000316   0.737758 /       permeability (krH2O) and CO2-rich phase
    0.4     0.005056   0.499535 /       relative permeability (krCO2) as
    0.5     0.025600   0.302400 /       function of water saturation (sH2O).
    0.6     0.080908   0.155832 /       These are the Brooks and Corey curves
    0.7     0.197530   0.061728 /       with Sl_min=0.2, Sg_Min=0.05.
    0.8     0.409600   0.014400 /
    0.9     0.758834   0.000572 /
    0.95    1.000000   0.000000 /
    1.0     1.0        0.0      /
/

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

RPTSATTA                                We report saturation functions
/                                       for visualization in ParaView.

INIT      ####################### INIT section begins here #####################

REGALL    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
                                        The keyword REGALL enable application
                                        operations on arrays both to domain and
                                        boundary grid blocks.

          The initial equilibrium is specified within brackets EQL-ENDEQL
EQL       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
/

EQUILGS
-- -------- -------- --------
-- datum    datum     GWC
--   depth  pressure   depth
-- -------- -------- --------
   2500     253.1     2000 /
-- -------- -------- --------

                    Using the keyword EQUILGS we specify that
                    - pressure is 253.1 bar at depth 2500 m;
```

```
                    - gas-water contact is above the top layer
                      (at depth 2000 m).

TEMPCVD
   2500  85  /      We specify that the temperature is 85C at 2500 m depth and
   3000  100 /      100 C at 3000 m depth.
/

XGMVD
   2500  0.0 /      We specify that the initial gas mass farction in brine
   3000  0.0 /      is 0.
/

YWMVD
   2500  0.0 /      We specify that the initial water mass farction in gas
   3000  0.0 /      is 0 (this data must be specified because the datum depth
/                   differs from the GWC depth).

ENDEQL    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

EQUALNAM                                 We specify that the injected CO2
  PRES    250. INJECTOR /                 temperature is 80 C at 25 MPa.
  TEMPC   80            /
  SGAS    1.0           /
  YWM     0.0           /                No water vapour in injected gas.
/

RPTSUM
  PRES TEMPC SGAS SLIQ XGM YWM /         We report these properties for the grid
                                         blocks.
RPTWELL
   WBHP WBHTC WGIR  WGIT
              WGIRM WGITM /              We report these properties for the
                                         wells.
RPTFIELD
   FGIP  FGIPL  FGIPG  FGIR  FGIT
   FGIPM FGIPLM FGIPGM FGIRM FGITM /     We report these properties for the
                                         field.

SCHEDULE   ##################### SCHEDULE section begins here ####################

WEEKTOL                                  We enable WEEKTOL modelling option
                                         (week convergence tolerance).

ILUTFILL                                 ILUT fill factor is 5.0.
   5 /
ILUTDROP                                 ILUT drop factor is 0.001.
  1E-3 /
```

```
231
232          ================================================================
233
234 TUNING                                  The next time step is 10 days. Maximal
235     1*  100  10 /                       time step is limited by 100 days.
236
237 TSTEP                                   No flow/injection during the first year.
238    365.25 /                             We check the initial equilibration
239                                         option.
240          ================================================================
241
242 EQUALREG
243 --          new
244 -- property value    marker    marker value
245 -- -------- -------- --------- ---------      Now, we fix parameters (pressure
246    ACTNUM   2        FLUXNUM   123       /    and temperature) at all boundary
247    ACTNUM   2        FLUXNUM   223       /    grid blocks.
248 -- -------- -------- --------- ---------
249 /
250
251 WELLINJE
252 -- -------- ---- ---- -------- -------- -------- -------- -------- --------
253 --   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
254 --                        rate     rate            rate              dimens
255 -- -------- ---- ---- -------- -------- -------- -------- -------- --------
256    INJECTOR OPEN MASS     0      1296      500     1E5 /
257 -- -------- ---- ---- -------- -------- -------- -------- -------- --------
258 /
259                                         Using the keyword WELLINJE, we start
260                                         CO2 injection. Injection rate is
261                                         1296 t/day. Maximal bottom hole
262                                         pressure is 50 MPa.
263
264 TUNING                                  The next time step is 1 day. The
265     1. 92.  1. /                        maximal time step is 92 days.
266
267 TSTEP                                   We advance simulation to 25 years
268    25*365.25 /                          reporting every year.
269
270          ================================================================
271
272 WELLSTOP                                Now, we stop the CO2 injection.
273    INJECTOR /
274 /
275
276 TUNING                                  The next time step is 1 day. The
277     1.  92  1. /                        maximal time step is 92 days.
278
```

```
279 TSTEP                                   We advance simulation for 25 years
280    25*365.25 /                          reporting every year.
281
282 POST      ######################## POST section begins here ####################
283
284 RPTPOST
285    TIMEY
286    FGIP  FGIPL  FGIPG  FGIR  FGIT
287    FGIPM FGIPLM FGIPGM FGIRM FGITM /    We save consolidated time series report
288 POSTFLD                                 for the field data.
289 /
290
291 RPTPOST
292    NOTHING  TIMEY
293    WBHP WBHTC WGIR  WGIT
294               WGIRM WGITM /             We save consolidated time series report
295 POSTWELL                                for the well.
296   '*' /
297 /
298
299 CONVERT                                 We convert the output to ParaView
300                                         compatible format.
301
302 END        ###########################################################################
```

9.17. Benchmark study for CO_2 storage (BLACKOIL)

This problem is discussed in the paper: Class H. et al. 2009 A benchmark study on problems related to CO_2 storage in geological formations. Comput. Geosci. 13(4):409-434. DOI:10.1007/s10596-009-9146-x. See a description of the problem, input data files and visualisations under *the link*.

EXAMPLE-H3-BLACKOIL-1.RUN

```
1  ###############################################################################
2
3                            BENCHMARK STUDY FOR CO2 STORAGE
4
5      This is problem 3.1 from
6      Class H. et al. 2009 A benchmark study on problems related to CO2 storage
7         in geological formations. Comput. Geosci. 13(4):409-434.
8          DOI:10.1007/s10596-009-9146-x
9
10     See description at: http://dx.doi.org/10.1007/s10596-009-9146-x
11
12 -------------------------------------------------------------------------------
13
14     Last update on 14.11.2015
15     Program Version: 2015.G01
16
17     Note: In this model, OIL stands for the water component, and GAS stands
18           for the CO2 component.
19
20 ###############################################################################
21
22
23 RUNSPEC   ################### RUNSPEC section begins here ######################
24
25 FAST                                  We use FAST option.
26
27 STARTTIM                              The initial time is -365.25 days=
28    -365.25 /                          = -1 year.
29
30 BLACKOIL  ^^^^^^ We switch the program to the GASSTORE EOS module ^^^^^^^^^^^^^
31
32                                       We enable
33 OIL                                   - oil phase (stands for water)
34 GAS                                   - gas phase (stands for CO2)
35 DISGAS                                - gas dissolution in oil (CO2 dissolves
36                                         in water)
37
38 METRIC    ^^^^^^^^^^^^ We use METRIC units ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
```

```
39
40 GRID      ###################### GRID section begins here ######################
41
42          The grid is specified within brackets MAKE-ENDMAKE
43 MAKE      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
44 -- corner point                          We select Corner-point gridding
45 --    grid   nx  ny  nz                  option and specify the number of
46      CORNER  78  78  9   /              grid blocks along every axis.
47
48 INCLUDE                                 We load the grid in corner-point format
49    'EXAMPLE-H3.GRDECL' /                from include file.
50
51 FAULTS
52 -- --name--  ------box------  -direction tag-    We specify fault plane
53    FAULT1    40 78 39 39 1 9   J+             /  named FAULT1.
54 -- --------  ---------------  ---------------
55 /
56
57 MULTFLT
58 -- --name--  -multiplier-               We multiply transmissibility across
59    FAULT1     0.0         /             the fault plane by factor 0.0.
60 -- --------  ------------
61 /
62
63 BOUNDARY
64 -- --name--  ------box------  --direction tags--  --type--  --mode--
65     123      6*                 I- I+ J- J+ 2*    SAMESIZE  ACTBASED /
66     223      40 78 39 39 1 9    J+ 5*             SAMESIZE  ALL      /
67     223      40 78 40 40 1 9    J- 5*             SAMESIZE  ALL      /
68 -- --------  ---------------  ------------------  --------  --------
69 /
70               Using the keyword BOUNDARY
71               - we encompass the reservoir with auxiliary grid blocks
72                 marked as FLUXNUM=123. Further, we use these blocks for
73                 modelling fluid influx from the aquifer;
74               - we create auxiliary grid blocks on either side of the fault
75                 plane marked as FLUXNUM=223. Further, we use these grid blocks
76                 to fix pressure at the fault plane.
77
78 WELSPECS
79 -- -------- -- ---- ---- --------
80 -- well        i-   j-   datum
81 --     name     ind  ind   depth
82 -- -------- -- ---- ---- --------       We specify well INJECTOR in the column
83    INJECTOR 1*  30   48  /              i=30, j=48.
84 -- -------- -- ---- ---- --------
85 /
86
```

```
COMPDAT
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
   INJECTOR   1*  1*   6    9   OPEN  1*        1*        0.15    /
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
/                                    We specify that the well is completed
                                     in layer k=6-9. The wellbore diameter
                                     is 0.15 meters.

DZWELL                               We define that the well is visible in
  250 25 /                           ParaView up to 250 meters above the top
                                     layer and up to 25 meters below the
                                     bottom layer.

ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

PORO                                 We load porosity from include file.
   ASCII 'EXAMPLE-H3.PORO' /

PERMX                                We load X-axis permeability from
   ASCII 'EXAMPLE-H3.PERM' /         include file.

COPY
-- --from-- ---to--- ------box------    We copy X-axis permeability into
   PERMX    PERMY    /                 Y-axis and Z-axis permeability,
   PERMX    PERMZ    /
-- -------- -------- ---------------
/

RPTGRID
   DEPTH XCOORD YCOORD PORO
   PERMX PERMY PERMZ
   I-IJKRES J-IJKRES K-IJKRES /      We report these properties from the
                                     GRID section.

PROPS     ######################## PROPS section begins here ####################

          Fluid properties are specified within brackets ROCK-ENDROCK
EOS       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
/

DENSITY
   998.313 1* 1.840 /                Oil density at stock tank conditions
                                     is 998.313 kg/m3, and gas density is
                                     1.84 kg/m3.

                                     Using the PVTO keyword we specify the
                                     properties of live oil.

```

```
PVTO
--  gas-oil ratio   pressure          FVF         viscosity
    3.7942823E-02  1.0000000E+00  1.0389080E+00  2.9454073E-01
                   2.5000000E+01  1.0377399E+00  2.9518849E-01
                   5.0000000E+01  1.0365231E+00  2.9586243E-01
                   7.5000000E+01  1.0353063E+00  2.9653562E-01
                   1.0000000E+02  1.0340895E+00  2.9720810E-01
                   1.2500000E+02  1.0328727E+00  2.9787996E-01
                   1.5000000E+02  1.0316559E+00  2.9855124E-01
                   1.7500000E+02  1.0304391E+00  2.9922201E-01
                   2.0000000E+02  1.0292223E+00  2.9989233E-01
                   2.2500000E+02  1.0280055E+00  3.0056224E-01
                   2.5000000E+02  1.0267887E+00  3.0123182E-01
                   2.7500000E+02  1.0255719E+00  3.0190111E-01
                   3.0000000E+02  1.0243551E+00  3.0257016E-01
                   3.2500000E+02  1.0231383E+00  3.0323903E-01
                   3.5000000E+02  1.0219215E+00  3.0390776E-01
                   3.7500000E+02  1.0207047E+00  3.0457640E-01 /
    6.0856824E+00  2.5000000E+01  1.0468221E+00  2.9518849E-01
                   5.0000000E+01  1.0456053E+00  2.9586243E-01
                   7.5000000E+01  1.0443885E+00  2.9653562E-01
                   1.0000000E+02  1.0431717E+00  2.9720810E-01
                   1.2500000E+02  1.0419549E+00  2.9787996E-01
                   1.5000000E+02  1.0407381E+00  2.9855124E-01
                   1.7500000E+02  1.0395213E+00  2.9922201E-01
                   2.0000000E+02  1.0383045E+00  2.9989233E-01
                   2.2500000E+02  1.0370878E+00  3.0056224E-01
                   2.5000000E+02  1.0358710E+00  3.0123182E-01
                   2.7500000E+02  1.0346542E+00  3.0190111E-01
                   3.0000000E+02  1.0334374E+00  3.0257016E-01
                   3.2500000E+02  1.0322206E+00  3.0323903E-01
                   3.5000000E+02  1.0310038E+00  3.0390776E-01
                   3.7500000E+02  1.0297870E+00  3.0457640E-01 /
    1.1374607E+01  5.0000000E+01  1.0535480E+00  2.9586243E-01
                   7.5000000E+01  1.0523312E+00  2.9653562E-01
                   1.0000000E+02  1.0511144E+00  2.9720810E-01
                   1.2500000E+02  1.0498976E+00  2.9787996E-01
                   1.5000000E+02  1.0486808E+00  2.9855124E-01
                   1.7500000E+02  1.0474640E+00  2.9922201E-01
                   2.0000000E+02  1.0462472E+00  2.9989233E-01
                   2.2500000E+02  1.0450304E+00  3.0056224E-01
                   2.5000000E+02  1.0438136E+00  3.0123182E-01
                   2.7500000E+02  1.0425968E+00  3.0190111E-01
                   3.0000000E+02  1.0413800E+00  3.0257016E-01
                   3.2500000E+02  1.0401632E+00  3.0323903E-01
                   3.5000000E+02  1.0389464E+00  3.0390776E-01
                   3.7500000E+02  1.0377297E+00  3.0457640E-01 /
    1.5717859E+01  7.5000000E+01  1.0588537E+00  2.9653562E-01
```

```
                  1.0000000E+02  1.0576369E+00  2.9720810E-01
                  1.2500000E+02  1.0564201E+00  2.9787996E-01
                  1.5000000E+02  1.0552033E+00  2.9855124E-01
                  1.7500000E+02  1.0539865E+00  2.9922201E-01
                  2.0000000E+02  1.0527697E+00  2.9989233E-01
                  2.2500000E+02  1.0515529E+00  3.0056224E-01
                  2.5000000E+02  1.0503361E+00  3.0123182E-01
                  2.7500000E+02  1.0491193E+00  3.0190111E-01
                  3.0000000E+02  1.0479025E+00  3.0257016E-01
                  3.2500000E+02  1.0466857E+00  3.0323903E-01
                  3.5000000E+02  1.0454689E+00  3.0390776E-01
                  3.7500000E+02  1.0442522E+00  3.0457640E-01 /
   1.9205146E+01  1.0000000E+02  1.0628740E+00  2.9720810E-01
                  1.2500000E+02  1.0616572E+00  2.9787996E-01
                  1.5000000E+02  1.0604404E+00  2.9855124E-01
                  1.7500000E+02  1.0592236E+00  2.9922201E-01
                  2.0000000E+02  1.0580068E+00  2.9989233E-01
                  2.2500000E+02  1.0567900E+00  3.0056224E-01
                  2.5000000E+02  1.0555732E+00  3.0123182E-01
                  2.7500000E+02  1.0543564E+00  3.0190111E-01
                  3.0000000E+02  1.0531396E+00  3.0257016E-01
                  3.2500000E+02  1.0519228E+00  3.0323903E-01
                  3.5000000E+02  1.0507060E+00  3.0390776E-01
                  3.7500000E+02  1.0494892E+00  3.0457640E-01 /
   2.1946685E+01  1.2500000E+02  1.0657743E+00  2.9787996E-01
                  1.5000000E+02  1.0645575E+00  2.9855124E-01
                  1.7500000E+02  1.0633407E+00  2.9922201E-01
                  2.0000000E+02  1.0621239E+00  2.9989233E-01
                  2.2500000E+02  1.0609071E+00  3.0056224E-01
                  2.5000000E+02  1.0596903E+00  3.0123182E-01
                  2.7500000E+02  1.0584735E+00  3.0190111E-01
                  3.0000000E+02  1.0572567E+00  3.0257016E-01
                  3.2500000E+02  1.0560399E+00  3.0323903E-01
                  3.5000000E+02  1.0548231E+00  3.0390776E-01
                  3.7500000E+02  1.0536063E+00  3.0457640E-01 /
   2.4100486E+01  1.5000000E+02  1.0677920E+00  2.9855124E-01
                  1.7500000E+02  1.0665752E+00  2.9922201E-01
                  2.0000000E+02  1.0653584E+00  2.9989233E-01
                  2.2500000E+02  1.0641416E+00  3.0056224E-01
                  2.5000000E+02  1.0629248E+00  3.0123182E-01
                  2.7500000E+02  1.0617080E+00  3.0190111E-01
                  3.0000000E+02  1.0604912E+00  3.0257016E-01
                  3.2500000E+02  1.0592744E+00  3.0323903E-01
                  3.5000000E+02  1.0580576E+00  3.0390776E-01
                  3.7500000E+02  1.0568408E+00  3.0457640E-01 /
   2.5856639E+01  1.7500000E+02  1.0692125E+00  2.9922201E-01
                  2.0000000E+02  1.0679957E+00  2.9989233E-01
                  2.2500000E+02  1.0667789E+00  3.0056224E-01
```

```
                    2.5000000E+02  1.0655621E+00  3.0123182E-01
                    2.7500000E+02  1.0643453E+00  3.0190111E-01
                    3.0000000E+02  1.0631285E+00  3.0257016E-01
                    3.2500000E+02  1.0619117E+00  3.0323903E-01
                    3.5000000E+02  1.0606949E+00  3.0390776E-01
                    3.7500000E+02  1.0594781E+00  3.0457640E-01 /
    2.7352480E+01   2.0000000E+02  1.0702421E+00  2.9989233E-01
                    2.2500000E+02  1.0690253E+00  3.0056224E-01
                    2.5000000E+02  1.0678085E+00  3.0123182E-01
                    2.7500000E+02  1.0665917E+00  3.0190111E-01
                    3.0000000E+02  1.0653749E+00  3.0257016E-01
                    3.2500000E+02  1.0641581E+00  3.0323903E-01
                    3.5000000E+02  1.0629413E+00  3.0390776E-01
                    3.7500000E+02  1.0617245E+00  3.0457640E-01 /
    2.8649228E+01   2.2500000E+02  1.0709727E+00  3.0056224E-01
                    2.5000000E+02  1.0697559E+00  3.0123182E-01
                    2.7500000E+02  1.0685391E+00  3.0190111E-01
                    3.0000000E+02  1.0673223E+00  3.0257016E-01
                    3.2500000E+02  1.0661055E+00  3.0323903E-01
                    3.5000000E+02  1.0648887E+00  3.0390776E-01
                    3.7500000E+02  1.0636719E+00  3.0457640E-01 /
    2.9779722E+01   2.5000000E+02  1.0714536E+00  3.0123182E-01
                    2.7500000E+02  1.0702368E+00  3.0190111E-01
                    3.0000000E+02  1.0690200E+00  3.0257016E-01
                    3.2500000E+02  1.0678032E+00  3.0323903E-01
                    3.5000000E+02  1.0665864E+00  3.0390776E-01
                    3.7500000E+02  1.0653696E+00  3.0457640E-01 /
    3.0770636E+01   2.7500000E+02  1.0717249E+00  3.0190111E-01
                    3.0000000E+02  1.0705081E+00  3.0257016E-01
                    3.2500000E+02  1.0692913E+00  3.0323903E-01
                    3.5000000E+02  1.0680745E+00  3.0390776E-01
                    3.7500000E+02  1.0668577E+00  3.0457640E-01 /
    3.1644299E+01   3.0000000E+02  1.0718202E+00  3.0257016E-01
                    3.2500000E+02  1.0706034E+00  3.0323903E-01
                    3.5000000E+02  1.0693866E+00  3.0390776E-01
                    3.7500000E+02  1.0681698E+00  3.0457640E-01 /
    3.2418933E+01   3.2500000E+02  1.0717667E+00  3.0323903E-01
                    3.5000000E+02  1.0705499E+00  3.0390776E-01
                    3.7500000E+02  1.0693331E+00  3.0457640E-01 /
    3.3108892E+01   3.5000000E+02  1.0715860E+00  3.0390776E-01
                    3.7500000E+02  1.0703692E+00  3.0457640E-01 /
    3.3725815E+01   3.7500000E+02  1.0712957E+00  3.0457640E-01 /
/

PVDG
--    pressure         FVF         viscosity
    1.0000000E+00  1.2797705E+00  1.8397191E-02 /   Using the PVDG keyword we
    2.5000000E+01  4.8122672E-02  1.8843426E-02 /   specify the properties of
```

```
    5.0000000E+01  2.2400218E-02  1.9569501E-02 /   dry gas.
    7.5000000E+01  1.3789654E-02  2.0682755E-02 /
    1.0000000E+02  9.4760185E-03  2.2377593E-02 /
    1.2500000E+02  6.9192121E-03  2.4934342E-02 /
    1.5000000E+02  5.3027814E-03  2.8614381E-02 /
    1.7500000E+02  4.2888684E-03  3.3308777E-02 /
    2.0000000E+02  3.6692250E-03  3.8398987E-02 /
    2.2500000E+02  3.2827887E-03  4.3312356E-02 /
    2.5000000E+02  3.0267616E-03  4.7836774E-02 /
    2.7500000E+02  2.8456334E-03  5.1965946E-02 /
    3.0000000E+02  2.7102097E-03  5.5752763E-02 /
    3.2500000E+02  2.6044461E-03  5.9255376E-02 /
    3.5000000E+02  2.5190001E-03  6.2523450E-02 /
    3.7500000E+02  2.4480955E-03  6.5597010E-02 /
/

ENDEOS    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          Rock properties are specified within brackets ROCK-ENDROCK
ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

ROCKDH                                 We specify that
  2500  1.0 /                          rock density is 2500 kg/m3,
                                       rock heat capacity is 1.0 kJ/kg/K

ENDROCK   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The relative permeabilities are specified within brackets SAT-ENDSAT
SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

SATOPT                                 We use an alternative method for
   2*  1 /                             interpolation of saturation functions

SGOF
-- sgas      krgas      kroil
   0.0      0.0        1.0      /      We define the gas relative permeability
   0.05     0.000000   1.000000 /      (krgas) and the oil relative
   0.1      0.000572   0.758834 /      permeability (kroil) as the function of
   0.2      0.014400   0.409600 /      the gas saturation (sgas).
   0.3      0.061728   0.197530 /
   0.4      0.155832   0.080908 /
   0.5      0.302400   0.025600 /
   0.6      0.499535   0.005056 /
   0.7      0.737758   0.000316 /
   0.8      1.000000   0.000000 /
/
```

```

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

RPTSATTA                              We report saturation functions
/                                     for visualization in ParaView.

INIT      ####################### INIT section begins here #####################

REGALL    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
                                      The keyword REGALL enable application
                                      operations on arrays both to domain and
                                      boundary grid blocks.

          The initial equilibrium is specified within brackets EQL-ENDEQL
EQL       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
/

EQUIL
-- -------- -------- -------- -------- --------
-- datum    datum                      GOC
--   depth  pressure                   depth
-- -------- -------- -------- -------- --------
   2500     253.1     1*       1*       2000 /
-- -------- -------- -------- -------- --------

                    Using the keyword EQUIL we specify that
                    - pressure is 253.1 bar at depth 2500 m;
                    - gas-oil contact is above the top layer
                      (at depth 2000 m).

RSVD
  2500 0.0 /                          We specify that at the initial moment
  3000 0.0 /                          of time there is no dissolved gas in
/                                     oil.

ENDEQL    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

RPTSUM
  PRES SGAS SOIL RS  /                We report these properties for the grid
                                      blocks.
RPTWELL
   WBHP WGIR WGIT WGIRM WGITM /       We report these properties for the
                                      wells.
RPTFIELD
   FGIPM FGIPLM FGIPGM
   FOIPM FOIPLM FOIPGM
   FGIRM FGITM
   FGIP  FGIPL  FGIPG /               We report these properties for the
```

```
                                        field.

SCHEDULE   ##################### SCHEDULE section begins here ####################

WEEKTOL                                 We enable WEEKTOL modelling option
                                        (week convergence tolerance).

ILUTFILL                                ILUT fill factor is 5.0.
   5 /
ILUTDROP                                ILUT drop factor is 0.001.
  1E-3 /

VARS
  PRES DMAX 50 /                        Maximum pressure change is 50 bars.
  PRES MAXV 400 /                       Maximum pressure is 400 bars.
/

          ========================================================================

TUNING                                  The next time step is 10 days. Maximal
    1*  100  10 /                       time step is limited by 100 days.

TSTEP                                   No flow/injection during the first year.
   365.25 /                             We check the initial equilibration
                                        option.
          ========================================================================

EQUALREG
--          new
-- property value    marker    marker value
-- -------- -------- --------- ---------      Now, we fix parameters (pressure
   ACTNUM   2        FLUXNUM   123       /    and temperature) at all boundary
   ACTNUM   2        FLUXNUM   223       /    grid blocks.
-- -------- -------- --------- ---------
/

WELLINJE
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
--   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
--                         rate     rate            rate              dimens
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
   INJECTOR OPEN MASS     0      1296      500     1*      GAS /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
/
                                        Using the keyword WELLINJE, we start
                                        CO2 injection. Injection rate is
                                        1296 t/day. Maximal bottom hole
                                        pressure is 500 bar. We redefine
```

```
                                    parameters in the stock tank to the
                                    GAS state.

TUNING                              The next time step is 1 day. The
    1. 92.  1. /                    maximal time step is 92 days.

TSTEP                               We advance simulation to 25 years
   25*365.25 /                      reporting every year.

          ================================================================

WELLSTOP                            Now, we stop the CO2 injection.
   INJECTOR /
/

TUNING                              The next time step is 1 day. The
    1.  92  1. /                    maximal time step is 92 days.

TSTEP                               We advance simulation for 25 years
   25*365.25 /                      reporting every year.

POST      ####################### POST section begins here ####################

RPTPOST
   TIMEY
   FGIPM FGIPLM FGIPGM
   FOIPM FOIPLM FOIPGM
   FGIRM FGITM
   FGIP  FGIPL  FGIPG  /            We save consolidated time series report
POSTFLD                             for the field data.
/

RPTPOST
   NOTHING  TIMEY
   WBHP WGIR WGIT WGIRM WGITM /     We save consolidated time series report
POSTWELL                            for the well.
  '*' /
/
CONVERT                             We convert the output to ParaView
                                    compatible format.

END        ###########################################################################
```

9.18. Benchmark study for CO_2 storage (BLACKOIL; VAPOIL option)

This problem is discussed in the paper: Class H. et al. 2009 A benchmark study on problems related to CO_2 storage in geological formations. Comput. Geosci. 13(4):409-434. DOI:10.1007/s10596-009-9146-x. See a description of the problem, input data files and visualisations under *the link*.

EXAMPLE-H3-BLACKOIL-2.RUN

```
1  ##############################################################################
2
3                           BENCHMARK STUDY FOR CO2 STORAGE
4
5      This is problem 3.1 from
6      Class H. et al. 2009 A benchmark study on problems related to CO2 storage
7         in geological formations. Comput. Geosci. 13(4):409-434.
8          DOI:10.1007/s10596-009-9146-x
9
10     See description at: http://dx.doi.org/10.1007/s10596-009-9146-x
11
12 ------------------------------------------------------------------------------
13
14     Last update on 14.11.2015
15     Program Version: 2015.G01
16
17     Note: In this model, OIL stands for the water component, and GAS stands
18           for the CO2 component.
19
20 ##############################################################################
21
22
23 RUNSPEC   ################### RUNSPEC section begins here ######################
24
25 FAST                                  We use FAST option.
26
27 STARTTIM                              The initial time is -365.25 days=
28    -365.25 /                          = -1 year.
29
30 BLACKOIL  ^^^^^^ We switch the program to the GASSTORE EOS module ^^^^^^^^^^^^^
31
32                                       We enable
33 OIL                                   - oil phase (stands for water)
34 GAS                                   - gas phase (stands for CO2)
35 DISGAS                                - gas dissolution in oil (CO2 dissolves
36                                         in water)
```

```
VAPOIL                                  - vaposized oil (water vapour in gas
                                          phase)

METRIC    ^^^^^^^^^^^^ We use METRIC units ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

GRID      ###################### GRID section begins here ######################

          The grid is specified within brackets MAKE-ENDMAKE
MAKE      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
-- corner point                         We select Corner-point gridding
--    grid   nx  ny  nz                 option and specify the number of
     CORNER  78  78  9   /              grid blocks along every axis.

INCLUDE                                 We load the grid in corner-point format
   'EXAMPLE-H3.GRDECL' /                from include file.

FAULTS
-- --name--  ------box------  -direction tag-    We specify fault plane
   FAULT1    40 78 39 39 1 9   J+             /  named FAULT1.
-- --------  ---------------  ---------------
/

MULTFLT
-- --name--  -multiplier-               We multiply transmissibility across
   FAULT1     0.0         /             the fault plane by factor 0.0.
-- --------  ------------
/

BOUNDARY
-- --name--  ------box------  --direction tags--  --type--  --mode--
    123      6*                 I- I+ J- J+ 2*    SAMESIZE  ACTBASED /
    223      40 78 39 39 1 9    J+ 5*             SAMESIZE  ALL      /
    223      40 78 40 40 1 9    J- 5*             SAMESIZE  ALL      /
-- --------  ---------------  ------------------  --------  --------
/
               Using the keyword BOUNDARY
               - we encompass the reservoir with auxiliary grid blocks
                 marked as FLUXNUM=123. Further, we use these blocks for
                 modelling fluid influx from the aquifer;
               - we create auxiliary grid blocks on either side of the fault
                 plane marked as FLUXNUM=223. Further, we use these grid blocks
                 to fix pressure at the fault plane.

WELSPECS
-- -------- -- ---- ---- --- --------
-- well        i-   j-
--     name     ind  ind     fluxnum
-- -------- -- ---- ---- --- --------   We specify well INJECTOR in the column
```

```
   INJECTOR 1*  30   48  12*  33   /    i=30, j=48. FLUXNUM=33 in the pipe
-- -------- -- ---- ---- --- --------   segment asscociated with well.
/

COMPDAT
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
   INJECTOR   1*  1*   6    9   OPEN  1*        1*        0.15    /
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
/                                     We specify that the well is completed
                                      in layer k=6-9. The wellbore diameter
                                      is 0.15 meters.

DZWELL                                We define that the well is visible in
  250 25 /                            ParaView up to 250 meters above the top
                                      layer and up to 25 meters below the
                                      bottom layer.

ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

PORO                                  We load porosity from include file.
   ASCII 'EXAMPLE-H3.PORO' /

PERMX                                 We load X-axis permeability from
   ASCII 'EXAMPLE-H3.PERM' /          include file.

COPY
-- --from-- ---to--- ------box------  We copy X-axis permeability into
   PERMX    PERMY    /                Y-axis and Z-axis permeability,
   PERMX    PERMZ    /
-- -------- -------- ---------------
/

RPTGRID
   DEPTH XCOORD YCOORD PORO
   PERMX PERMY PERMZ
   I-IJKRES J-IJKRES K-IJKRES /       We report these properties from the
                                      GRID section.

PROPS     ######################## PROPS section begins here ####################

          Fluid properties are specified within brackets ROCK-ENDROCK
EOS       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
/

DENSITY
   998.313 1* 1.840 /                 Oil density at stock tank conditions
                                      is 998.313 kg/m3, and gas density is
                                      1.84 kg/m3.
```

```
133
134                                     Using the PVTO keyword we specify the
135                                     properties of live oil.
136
137 PVTO
138 --  gas-oil ratio   pressure          FVF         viscosity
139     3.7942823E-02  1.0000000E+00  1.0389080E+00  2.9454073E-01
140                    2.5000000E+01  1.0377399E+00  2.9518849E-01
141                    5.0000000E+01  1.0365231E+00  2.9586243E-01
142                    7.5000000E+01  1.0353063E+00  2.9653562E-01
143                    1.0000000E+02  1.0340895E+00  2.9720810E-01
144                    1.2500000E+02  1.0328727E+00  2.9787996E-01
145                    1.5000000E+02  1.0316559E+00  2.9855124E-01
146                    1.7500000E+02  1.0304391E+00  2.9922201E-01
147                    2.0000000E+02  1.0292223E+00  2.9989233E-01
148                    2.2500000E+02  1.0280055E+00  3.0056224E-01
149                    2.5000000E+02  1.0267887E+00  3.0123182E-01
150                    2.7500000E+02  1.0255719E+00  3.0190111E-01
151                    3.0000000E+02  1.0243551E+00  3.0257016E-01
152                    3.2500000E+02  1.0231383E+00  3.0323903E-01
153                    3.5000000E+02  1.0219215E+00  3.0390776E-01
154                    3.7500000E+02  1.0207047E+00  3.0457640E-01 /
155     6.0856824E+00  2.5000000E+01  1.0468221E+00  2.9518849E-01
156                    5.0000000E+01  1.0456053E+00  2.9586243E-01
157                    7.5000000E+01  1.0443885E+00  2.9653562E-01
158                    1.0000000E+02  1.0431717E+00  2.9720810E-01
159                    1.2500000E+02  1.0419549E+00  2.9787996E-01
160                    1.5000000E+02  1.0407381E+00  2.9855124E-01
161                    1.7500000E+02  1.0395213E+00  2.9922201E-01
162                    2.0000000E+02  1.0383045E+00  2.9989233E-01
163                    2.2500000E+02  1.0370878E+00  3.0056224E-01
164                    2.5000000E+02  1.0358710E+00  3.0123182E-01
165                    2.7500000E+02  1.0346542E+00  3.0190111E-01
166                    3.0000000E+02  1.0334374E+00  3.0257016E-01
167                    3.2500000E+02  1.0322206E+00  3.0323903E-01
168                    3.5000000E+02  1.0310038E+00  3.0390776E-01
169                    3.7500000E+02  1.0297870E+00  3.0457640E-01 /
170     1.1374607E+01  5.0000000E+01  1.0535480E+00  2.9586243E-01
171                    7.5000000E+01  1.0523312E+00  2.9653562E-01
172                    1.0000000E+02  1.0511144E+00  2.9720810E-01
173                    1.2500000E+02  1.0498976E+00  2.9787996E-01
174                    1.5000000E+02  1.0486808E+00  2.9855124E-01
175                    1.7500000E+02  1.0474640E+00  2.9922201E-01
176                    2.0000000E+02  1.0462472E+00  2.9989233E-01
177                    2.2500000E+02  1.0450304E+00  3.0056224E-01
178                    2.5000000E+02  1.0438136E+00  3.0123182E-01
179                    2.7500000E+02  1.0425968E+00  3.0190111E-01
180                    3.0000000E+02  1.0413800E+00  3.0257016E-01
```

```
                  3.2500000E+02  1.0401632E+00  3.0323903E-01
                  3.5000000E+02  1.0389464E+00  3.0390776E-01
                  3.7500000E+02  1.0377297E+00  3.0457640E-01 /
   1.5717859E+01  7.5000000E+01  1.0588537E+00  2.9653562E-01
                  1.0000000E+02  1.0576369E+00  2.9720810E-01
                  1.2500000E+02  1.0564201E+00  2.9787996E-01
                  1.5000000E+02  1.0552033E+00  2.9855124E-01
                  1.7500000E+02  1.0539865E+00  2.9922201E-01
                  2.0000000E+02  1.0527697E+00  2.9989233E-01
                  2.2500000E+02  1.0515529E+00  3.0056224E-01
                  2.5000000E+02  1.0503361E+00  3.0123182E-01
                  2.7500000E+02  1.0491193E+00  3.0190111E-01
                  3.0000000E+02  1.0479025E+00  3.0257016E-01
                  3.2500000E+02  1.0466857E+00  3.0323903E-01
                  3.5000000E+02  1.0454689E+00  3.0390776E-01
                  3.7500000E+02  1.0442522E+00  3.0457640E-01 /
   1.9205146E+01  1.0000000E+02  1.0628740E+00  2.9720810E-01
                  1.2500000E+02  1.0616572E+00  2.9787996E-01
                  1.5000000E+02  1.0604404E+00  2.9855124E-01
                  1.7500000E+02  1.0592236E+00  2.9922201E-01
                  2.0000000E+02  1.0580068E+00  2.9989233E-01
                  2.2500000E+02  1.0567900E+00  3.0056224E-01
                  2.5000000E+02  1.0555732E+00  3.0123182E-01
                  2.7500000E+02  1.0543564E+00  3.0190111E-01
                  3.0000000E+02  1.0531396E+00  3.0257016E-01
                  3.2500000E+02  1.0519228E+00  3.0323903E-01
                  3.5000000E+02  1.0507060E+00  3.0390776E-01
                  3.7500000E+02  1.0494892E+00  3.0457640E-01 /
   2.1946685E+01  1.2500000E+02  1.0657743E+00  2.9787996E-01
                  1.5000000E+02  1.0645575E+00  2.9855124E-01
                  1.7500000E+02  1.0633407E+00  2.9922201E-01
                  2.0000000E+02  1.0621239E+00  2.9989233E-01
                  2.2500000E+02  1.0609071E+00  3.0056224E-01
                  2.5000000E+02  1.0596903E+00  3.0123182E-01
                  2.7500000E+02  1.0584735E+00  3.0190111E-01
                  3.0000000E+02  1.0572567E+00  3.0257016E-01
                  3.2500000E+02  1.0560399E+00  3.0323903E-01
                  3.5000000E+02  1.0548231E+00  3.0390776E-01
                  3.7500000E+02  1.0536063E+00  3.0457640E-01 /
   2.4100486E+01  1.5000000E+02  1.0677920E+00  2.9855124E-01
                  1.7500000E+02  1.0665752E+00  2.9922201E-01
                  2.0000000E+02  1.0653584E+00  2.9989233E-01
                  2.2500000E+02  1.0641416E+00  3.0056224E-01
                  2.5000000E+02  1.0629248E+00  3.0123182E-01
                  2.7500000E+02  1.0617080E+00  3.0190111E-01
                  3.0000000E+02  1.0604912E+00  3.0257016E-01
                  3.2500000E+02  1.0592744E+00  3.0323903E-01
                  3.5000000E+02  1.0580576E+00  3.0390776E-01
```

```
                    3.7500000E+02  1.0568408E+00  3.0457640E-01 /
    2.5856639E+01   1.7500000E+02  1.0692125E+00  2.9922201E-01
                    2.0000000E+02  1.0679957E+00  2.9989233E-01
                    2.2500000E+02  1.0667789E+00  3.0056224E-01
                    2.5000000E+02  1.0655621E+00  3.0123182E-01
                    2.7500000E+02  1.0643453E+00  3.0190111E-01
                    3.0000000E+02  1.0631285E+00  3.0257016E-01
                    3.2500000E+02  1.0619117E+00  3.0323903E-01
                    3.5000000E+02  1.0606949E+00  3.0390776E-01
                    3.7500000E+02  1.0594781E+00  3.0457640E-01 /
    2.7352480E+01   2.0000000E+02  1.0702421E+00  2.9989233E-01
                    2.2500000E+02  1.0690253E+00  3.0056224E-01
                    2.5000000E+02  1.0678085E+00  3.0123182E-01
                    2.7500000E+02  1.0665917E+00  3.0190111E-01
                    3.0000000E+02  1.0653749E+00  3.0257016E-01
                    3.2500000E+02  1.0641581E+00  3.0323903E-01
                    3.5000000E+02  1.0629413E+00  3.0390776E-01
                    3.7500000E+02  1.0617245E+00  3.0457640E-01 /
    2.8649228E+01   2.2500000E+02  1.0709727E+00  3.0056224E-01
                    2.5000000E+02  1.0697559E+00  3.0123182E-01
                    2.7500000E+02  1.0685391E+00  3.0190111E-01
                    3.0000000E+02  1.0673223E+00  3.0257016E-01
                    3.2500000E+02  1.0661055E+00  3.0323903E-01
                    3.5000000E+02  1.0648887E+00  3.0390776E-01
                    3.7500000E+02  1.0636719E+00  3.0457640E-01 /
    2.9779722E+01   2.5000000E+02  1.0714536E+00  3.0123182E-01
                    2.7500000E+02  1.0702368E+00  3.0190111E-01
                    3.0000000E+02  1.0690200E+00  3.0257016E-01
                    3.2500000E+02  1.0678032E+00  3.0323903E-01
                    3.5000000E+02  1.0665864E+00  3.0390776E-01
                    3.7500000E+02  1.0653696E+00  3.0457640E-01 /
    3.0770636E+01   2.7500000E+02  1.0717249E+00  3.0190111E-01
                    3.0000000E+02  1.0705081E+00  3.0257016E-01
                    3.2500000E+02  1.0692913E+00  3.0323903E-01
                    3.5000000E+02  1.0680745E+00  3.0390776E-01
                    3.7500000E+02  1.0668577E+00  3.0457640E-01 /
    3.1644299E+01   3.0000000E+02  1.0718202E+00  3.0257016E-01
                    3.2500000E+02  1.0706034E+00  3.0323903E-01
                    3.5000000E+02  1.0693866E+00  3.0390776E-01
                    3.7500000E+02  1.0681698E+00  3.0457640E-01 /
    3.2418933E+01   3.2500000E+02  1.0717667E+00  3.0323903E-01
                    3.5000000E+02  1.0705499E+00  3.0390776E-01
                    3.7500000E+02  1.0693331E+00  3.0457640E-01 /
    3.3108892E+01   3.5000000E+02  1.0715860E+00  3.0390776E-01
                    3.7500000E+02  1.0703692E+00  3.0457640E-01 /
    3.3725815E+01   3.7500000E+02  1.0712957E+00  3.0457640E-01 /
/
```

```
                                        Using the PVTG keyword we specify the
                                        properties of wet gas.

PVTG
--    pressure     oil-gas ratio       FVF         viscosity
    1.0000000E+00  4.6992416E-03  4.5431950E+00  1.8397191E-02
                   0.0000000E+00  4.5431950E+00  1.8397191E-02 /
    2.5000000E+01  3.0794089E-05  4.8926811E-02  1.8843426E-02
                   0.0000000E+00  4.8926811E-02  1.8843426E-02 /
    5.0000000E+01  1.7510412E-05  2.2613062E-02  1.9569501E-02
                   0.0000000E+00  2.2613062E-02  1.9569501E-02 /
    7.5000000E+01  1.3630656E-05  1.3891650E-02  2.0682755E-02
                   0.0000000E+00  1.3891650E-02  2.0682755E-02 /
    1.0000000E+02  1.2186467E-05  9.5386824E-03  2.2377593E-02
                   0.0000000E+00  9.5386824E-03  2.2377593E-02 /
    1.2500000E+02  1.1863351E-05  6.9637549E-03  2.4934342E-02
                   0.0000000E+00  6.9637549E-03  2.4934342E-02 /
    1.5000000E+02  1.2210082E-05  5.3379161E-03  2.8614381E-02
                   0.0000000E+00  5.3379161E-03  2.8614381E-02 /
    1.7500000E+02  1.2899778E-05  4.3188904E-03  3.3308777E-02
                   0.0000000E+00  4.3188904E-03  3.3308777E-02 /
    2.0000000E+02  1.3627126E-05  3.6963576E-03  3.8398987E-02
                   0.0000000E+00  3.6963576E-03  3.8398987E-02 /
    2.2500000E+02  1.4238443E-05  3.3081528E-03  4.3312356E-02
                   0.0000000E+00  3.3081528E-03  4.3312356E-02 /
    2.5000000E+02  1.4717046E-05  3.0509336E-03  4.7836774E-02
                   0.0000000E+00  3.0509336E-03  4.7836774E-02 /
    2.7500000E+02  1.5089915E-05  2.8689347E-03  5.1965946E-02
                   0.0000000E+00  2.8689347E-03  5.1965946E-02 /
    3.0000000E+02  1.5385450E-05  2.7328367E-03  5.5752763E-02
                   0.0000000E+00  2.7328367E-03  5.5752763E-02 /
    3.2500000E+02  1.5625081E-05  2.6265288E-03  5.9255376E-02
                   0.0000000E+00  2.6265288E-03  5.9255376E-02 /
    3.5000000E+02  1.5823856E-05  2.5406300E-03  6.2523450E-02
                   0.0000000E+00  2.5406300E-03  6.2523450E-02 /
    3.7500000E+02  1.5992246E-05  2.4693403E-03  6.5597010E-02
                   0.0000000E+00  2.4693403E-03  6.5597010E-02 /
/

ENDEOS    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>


          Rock properties are specified within brackets ROCK-ENDROCK
ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

ROCKDH                                  We specify that
  2500  1.0 /                           rock density is 2500 kg/m3,
```

```
                                        rock heat capacity is 1.0 kJ/kg/K

ENDROCK   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The relative permeabilities are specified within brackets SAT-ENDSAT
SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

SATOPT                                  We use an alternative method for
   2*  1 /                              interpolation of saturation functions

SGOF
-- sgas      krgas      kroil
   0.0      0.0        1.0      /       We define the gas relative permeability
   0.05     0.000000   1.000000 /       (krgas) and the oil relative
   0.1      0.000572   0.758834 /       permeability (kroil) as the function of
   0.2      0.014400   0.409600 /       the gas saturation (sgas).
   0.3      0.061728   0.197530 /
   0.4      0.155832   0.080908 /
   0.5      0.302400   0.025600 /
   0.6      0.499535   0.005056 /
   0.7      0.737758   0.000316 /
   0.8      1.000000   0.000000 /
/

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

RPTSATTA                                We report saturation functions
/                                       for visualization in ParaView.

INIT      ####################### INIT section begins here #####################

REGALL    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
                                        The keyword REGALL enable application
                                        operations on arrays both to domain and
                                        boundary grid blocks.

          The initial equilibrium is specified within brackets EQL-ENDEQL
EQL       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
/

EQUIL
-- -------- -------- -------- -------- --------
-- datum    datum                       GOC
--   depth  pressure                    depth
-- -------- -------- -------- -------- --------
   2500     253.1     1*       1*       2000 /
-- -------- -------- -------- -------- --------
```

```

                    Using the keyword EQUIL we specify that
                    - pressure is 253.1 bar at depth 2500 m;
                    - gas-oil contact is above the top layer
                      (at depth 2000 m).

RSVD
  2500 0.0 /                            We specify that at the initial moment
  3000 0.0 /                            of time there is no dissolved gas in
/                                       oil.

RVVD
  1500 0.0 /                            No water vapour in gas above the
  2000 0.0 /                            contact.
/

ENDEQL    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

REGALL    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
EQUALREG
-- -------- -------- -------- --------
    PRES      300    FLUXNUM   33    /  We define parameters in the pipe
    SGAS      1.0    /                  segment associated with the well.
    RV        0.0    /
-- -------- -------- -------- --------
/

RPTSUM
  PRES SGAS SOIL RS RV  /               We report these properties for the grid
                                        blocks.
RPTWELL
   WBHP WGIR WGIT WGIRM WGITM /         We report these properties for the
                                        wells.
RPTFIELD
   FGIPM FGIPLM FGIPGM
   FOIPM FOIPLM FOIPGM
   FGIRM FGITM
   FGIP  FGIPL  FGIPG  /                We report these properties for the
                                        field.

SCHEDULE   #################### SCHEDULE section begins here ####################

WEEKTOL                                 We enable WEEKTOL modelling option
                                        (week convergence tolerance).

ILUTFILL                                ILUT fill factor is 5.0.
   5 /
ILUTDROP                                ILUT drop factor is 0.001.
```

```
  1E-3 /

VARS
  PRES DMAX 50 /                        Maximum pressure change is 50 bars.
  PRES MAXV 400 /                       Maximum pressure is 400 bars.
/

          ================================================================

TUNING                                  The next time step is 10 days. Maximal
    1*  100  10 /                       time step is limited by 100 days.

TSTEP                                   No flow/injection during the first year.
   365.25 /                             We check the initial equilibration
                                        option.
          ================================================================

EQUALREG
--          new
-- property value    marker    marker value
-- -------- -------- --------- ---------      Now, we fix parameters (pressure
   ACTNUM   2        FLUXNUM   123       /    and temperature) at all boundary
   ACTNUM   2        FLUXNUM   223       /    grid blocks.
-- -------- -------- --------- ---------
/

WELLINJE
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
--   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
--                         rate     rate            rate              dimens
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
   INJECTOR OPEN MASS     0      1296      500     1*      GAS /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
/
                                        Using the keyword WELLINJE, we start
                                        CO2 injection. Injection rate is
                                        1296 t/day. Maximal bottom hole
                                        pressure is 500 bar. We redefine
                                        parameters in the stock tank to the
                                        GAS state.

TUNING                                  The next time step is 1 day. The
    1. 92.  1. /                        maximal time step is 92 days.

TSTEP                                   We advance simulation to 25 years
   25*365.25 /                          reporting every year.

          ================================================================
```

```
469
470 WELLSTOP                                  Now, we stop the CO2 injection.
471    INJECTOR /
472 /
473
474 TUNING                                    The next time step is 1 day. The
475     1.  92  1. /                          maximal time step is 92 days.
476
477 TSTEP                                     We advance simulation for 25 years
478    25*365.25 /                            reporting every year.
479
480 POST      ######################## POST section begins here ######################
481
482 RPTPOST
483    TIMEY
484    FGIPM FGIPLM FGIPGM
485    FOIPM FOIPLM FOIPGM
486    FGIRM FGITM
487    FGIP  FGIPL  FGIPG  /                  We save consolidated time series report
488 POSTFLD                                   for the field data.
489 /
490
491 RPTPOST
492    NOTHING  TIMEY
493    WBHP WGIR WGIT WGIRM WGITM /           We save consolidated time series report
494 POSTWELL                                  for the well.
495   '*' /
496 /
497
498 CONVERT                                   We convert the output to ParaView
499                                           compatible format.
500
501 END        ###########################################################################
```

9.19. Example-H4 (BINMIXT)

This problem is discussed in the paper: Eigestad G.T., Dahle H.K. et al. 2009 Geologic modeling and simulation of CO_2 injection in the Johansen formation. Comput. Geosci. 2009. 13, P.435-450. DOI: 10.1007/s10596-009-9153-y. See a description of the problem, input data files and visualisations under *the link*.

EXAMPLE-H4.RUN

```
1  ###############################################################################
2
3                      CO2 INJECTION IN JOHANSEN FORMATION
4
5      See description at:
6              http://www.sintef.no/Projectweb/MatMoRA/Downloads/Johansen/
7
8  -------------------------------------------------------------------------------
9
10     Last update on 24.05.2015
11     Program Version: 2015.C09
12
13 ###############################################################################
14
15 -------------------------------------------------------------------------------
16
17         The sataset description:
18
19         This is MUFITS RUN-file for the NPD5 model of the Johansen formation.
20
21         Data sets used:
22                 Grid            NPD5.grdecl
23                 Faults          NPD5_Faults.txt
24                 Porosity        NPD5_Porosity.txt
25                 Permeability    NPD5_Permeability.txt
26                 Rel. Perm.      Scaled_RelPermWater.txt & Scaled_RelPermCO2.txt
27
28         All files can be loaded at:
29               http://www.sintef.no/Projectweb/MatMoRA/Downloads/Johansen/
30               "A sector model with heterogeneous rock properties" (NPD5 model)
31
32         The geological model has 11 layers:
33               layers 1-5   -  caprock, shale above Johansen;
34               layers 6-10  -  high permeable layers (Johansen formation);
35               layers 11    -  shale.
36
37 -------------------------------------------------------------------------------
```

```
38
39 RUNSPEC   ################### RUNSPEC section begins here ######################
40           In this section, the general options and parameters of the simulation
41              are specified.
42
43 FAST                                  We use FAST option.
44
45 GRID       ##################### GRID section begins here ######################
46           In this section, we must define the geometrical and logical structures
47              of the reservoir (computational grid, boundaries, faults, wells,
48              point sources, etc.).
49
50           The grid is specified within brackets MAKE-ENDMAKE
51 MAKE       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
52    CORNER 100 100 11 /
53           The opening bracket MAKE has the following associated data:
54              item 1 - the grid type mnemonic:
55                 CART   - cartresian grid;
56                 CORNER - corner-point grid (default option);
57                 RADIAL - radial grid;
58              item 2 - the number of cells along the i-indexation axis (x-axis);
59              item 3 - the number of cells along the j-indexation axis (y-axis);
60              item 4 - the number of cells along the k-indexation axis (z-axis).
61
62 MAPUNITS                              We use metric units for map axes.
63    'METRES' /
64
65 MAPAXES
66    503234.063181 6761075.871490
67    503234.063181 6760075.871490
68    504234.063181 6760075.871490 /    We specify map axes.
69
70 GRIDUNIT
71   'METRES' 'MAP' /                   We use metric units for the grid. Grid
72                                       is specified relative to the map axes.
73
74 INCLUDE
75    'NPD5.grdecl' /
76           The formatted corner-point grid file (the file extension
77              .GRDECL) is loaded using the keyword INCLUDE. The asscociated
78              data of this keyword is the system path to the grid file. MUFITS
79              recognizes the usual keywords in the file like: COORD, ZCORN,
80              ACTNUM, SPECGRID (the 1st three data items must be the same as
81              specified by the keyword MAKE; is used for the grid dimension
82              checking), MAPAXES, MAPUNITS (must be METRES), GRIDUNITS
83              (must be METRES).
84
85           ======================================================================
```

```
86
87  SKIP      <<<<<<<<<<< All keyword until ENDSKIP are skipped <<<<<<<<<<<<<<<<<<<<
88
89            Uncomment the following local grid refinement specification
90            to have a better resolution of the plume.
91
92  CARFIN    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
93  -- name   imin-imax   jmin-jmax   kmin-kmax   ni   nj   nk  parent_grid
94     LGR1    62   72     17   39     5    8     22   46   16 /
95  CARFIN    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
96     LGR2    36   57     45   60     5    8     44   32   16 /
97  CARFIN    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
98     LGR3    1    35     25   50     5    8     70   52   16 /
99  CARFIN    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
100    LGR4    44   61     1    39     5    8     36   78   16 /
101 CARFIN    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
102    LGR5    65   84     40   52     5    8     40   26   16 /
103 CARFIN    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
104    LGR6    73   84     29   39     5    8     24   22   16 /
105 CARFIN    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
106    LGR2A   8    23     15   32     3    11    32   36   16      LGR2 /
107 ENDFIN    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
108           The LGRs of cartesian grid type are specified by the keyword CARFIN.
109              The LGR is specified by the ijk-indexes. The data associated with
110              this keyword are as follows:
111              item 1 (name) - the name of LGR. Father, we can reference to the
112                 LGR by its name 'LGR1';
113              items 2,3 (imin-imax) - the minimal and the maximal i-index of the
114                 parent grid, to which the LGR is applied;
115              items 4,5 (jmin-jmax) - the minimal and the maximal j-index of the
116                 parent grid, to which the LGR is applied;
117              items 6,7 (kmin-kmax) - the minimal and the maximal k-index of the
118                 parent grid, to which the LGR is applied;
119              item 8 (ni) - the new number of cells along the i-indexation axis
120                 (ni>imax-imin);
121              item 9 (nj) - the new number of cells along the j-indexation axis
122                 (nj>jmax-jmin);
123              item 10 (nk) - the new number of cells along the k-indexation axis
124                 (nk>kmax-kmin);
125              item 11 (parent_grid) - the name of the parent grid. By the
126                 default option, the parent grid is the whole reservoir.
127           The grid LGR2A is nested in the grid LGR2.
128           After the keyword CARFIN is executed, MUFITS automatically creates the
129              the LGR and makes this grid active. The keyword ENDFIN makes the
130              parent grid of the reservoir active. This keyword does not have
131              associated data.
132
133 ENDSKIP   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

```
134
135          ================================================================
136
137 FAULTS
138    ASCII 'NPD5_Faults.txt' /
139          The faults are specified by the keyword FAULTS. This keyword has
140             tabular associated data:
141             item 1 - the fault ID (8 byte character); we can farther
142                reference to the fault by this ID;
143             items 2-7 - define cells, which have a face belonging
144                to the fault. The cells are defined using the ijk-indexation
145                box.
146             items 8 - defines, which faces of the cells form the fault.
147                This item must be any of X-,X+,Y-,Y+,Z-,Z+,I-,I+,J-,J+,K-,K+.
148          The keyword FAULTS can be tuned up to load the fault specification
149             from an include file. If the 1st argument of this keyword is the
150             mnemonic ASCII then  the loading is invoked. In this case the 2nd
151             argument is the system path to the include file. The mnemonic ASCII
152             signifies that the include file is the formatted file.
153          In the include file the following faults are defined: FAULT3, FAULT5,
154             FAULT6, FAULT9, FAULT10.
155
156 MULTFLT
157 -- --name--  -multiplier-
158    FAULT3     0.1 /
159    FAULT5     0.1 /
160    FAULT6     0.1 /
161    FAULT9     0.1 /
162    FAULT10    0.1 /
163 -- --------  ------------
164 /
165          The transmissibility multiplier across the fault is specified by
166             the keyword MULTFLT. This keyword has tabular associated data.
167             item 1 - the fault ID defined by the keyword FAULTS;
168             item 2 - the transmissibility multiplier.
169          According to the computational problem formulation the transmissibility
170             across every fault must be multiplied by 0.1.
171
172          ================================================================
173
174 BOUNDARY
175 -- --name--  ------box------  --direction tags--  --type--  --mode--
176     123      6*                 I- I+ J- J+ 2*     SAMESIZE /
177 -- --------  --------------  -----------------  --------  --------
178 /
179          Before defining the boundary conditions in the section INIT, we must
180             define the boundaries itself within these brackets MAKE/ENDMAKE.
181             The boundaries are defined using the keyword BOUNDARY, which has
```

```
182             tabular associated data: every row for a new boundary. The data
183             items are as follows:
184             item 1 - the boundary ID (an integer). The auxiliary cells,
185                which MUFITS creates for the boundary are marked out by the
186                condition FLUXNUM=ID.
187             items 2-7 - define a region of the reservoir (i=[imin,imax],
188                j=[jmin,jmax], k=[kmin,kmax]), for which the boundary is
189                defined. By default, this region is the whole active grid (e.g.
190                in the first data row i=[1,78], j=[1,78], k=[1,9]).
191             items 8-13 - the direction tags (any of X-,X+,Y-,Y+,Z-,Z+,I-,I+,
192                J-,J+,K-,K+). These tags define, which faces of the cells in
193                the region form the boundary.
194             item 14 - the type of the cells, which MUFITS creates for the
195                boundary (8 byte character). The possible values are:
196                   SAMESIZE - MUFITS creates the cells, which have the same
197                      geometrical volume and distance to the interface as the
198                      cell in the region defined by the items 2-7, to which
199                      the created cell is connected (default value).
200                   INFTHIN - MUFITS creates infinitely thin cells.
201             item 15 - the creation mode (8 byte character). The possible
202                values are:
203                   ACTBASED - MUFITS creates the auxiliary cells for every
204                      cell in the region defined by the items 2-7, which does
205                      not have an active cell connected to its face defined
206                      by the items 8-13 (default value).
207                   ALL - MUFITS creates the auxiliary cells for every cell
208                      in the region defined by the items 2-7 regardless the
209                      distribution of the property ACTNUM.
210          Using the keyword BOUNDARY we define the outer boundary of the
211             reservoir. We use the default mode ACTBASED in order not to
212             influence the internal regions of the reservoir.
213
214          ================================================================
215
216 GETWELLS                              We import the CO2 injection well
217                                       specification from the section SCHEDULE.
218
219 DZWELL                                The well is visible in ParaView up to
220   500 50 /                            500 metres above the top layer and
221                                       up to 50 meters below the bottom layer.
222
223          ================================================================
224
225 ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
226
227 PORO
228    ASCII 'NPD5_Porosity.txt' /
229          The keyword PORO can be tuned up to load the porosity from an include
```

```
            file. If the 1st data item of this keyword is the mnemonic ASCII
            or BINARY then the loading is invoked. In this case, the 2nd data
            item is file name, from which the porosity is loaded. The mnemonic
            ASCII signifies that the porosity file is a formatted file. The
            mnemonic BINARY signifies that the the porosity file is a binary
            file.
         According to the earlier specification of the arrays loading mode
            (the keyword LOADING), the file NPD5_Porosity.txt must contain
            the porosity value for every cell of the reservoir. The values are
            ordered in that i-index is running in the 1st place, j-index
            is running in the 2nd place, k-index is running in the 3rd
            place.

PERMX
   ASCII 'NPD5_Permeability.txt' /
         The syntax of the keyword PERMX is identical to the syntax of the
            keyword PORO except that the keyword is for loading the x-direction
            permeabilities. The permeability dimension is mD (millidarcy).

COPY
-- --from-- ---to--- ------box------
   PERMX    PERMY    /                 We copy X-direction permeability into
   PERMX    PERMZ    /                 Y- and Z-direction permeability.
-- -------- -------- ---------------
/

MULTIPLY
-- --property-- --multiplier-- ------box------   We reduce Z-direction
      PERMZ           0.1      /                 permeability by the factor
-- ------------ -------------- ---------------   0.1.
/
         According to the computational problem formulation, the following
            relations are hold:

                  PERMY=PERMX
                  PERMZ=0.1*PERMX   (the permeability ratio is 10)

         This specification can be done by copying the property PERMX to
            the properties PERMY and PERMZ. This operation can be done by the
            keyword COPY. This keyword has the following tabular associated
            data:
            item 1 - the mnemonic of the property, from which values are
               copied;
            item 2 - the mnemonic of the property, to which values are
               copied;
            item 3-8 - the input box in the active grid. The operation is
               applied only to the cells belonging to the box.
         Farther, we multiply the property PERMZ by 0.1. This operation can be
```

```
278             done by the keyword MULTIPLY. This keyword has the following
279             tabular associated data:
280             item 1 - the mnemonic of the modified property;
281             item 2 - the multiplier value;
282             item 3-8 - the input box in the active grid. The operation is
283                applied only to the cells belonging to the box.
284
285          Note that we don't define the petrophysical properties in the
286             auxiliary cells, which MUFITS creates for the boundaries. If the
287             petrophysical properties in these cells are not defined after the
288             section GRID then MUFITS copies these properties from the
289             cells, to which the auxiliary cells are connected.
290
291          ============================================================================
292
293 REGALL   ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
294          The switch REGALL does not have associated data. This keyword targets
295             MUFITS to perform operations on array for all cells in the
296             simulation but not only for the cells in the current input box.
297             This keyword affects the data processing by the following keyword
298             EQUALREG, which is related to the auxiliary cells of the boundary
299             condition. It also affects the boundary conditions specification
300             in the section INIT.
301
302 EQUALS
303    FIPNUM 2 /                          Now, FIPNUM=2 in all domain grid blocks.
304 /
305
306 EQUALREG
307 --          new
308 -- property value    marker    marker value
309 -- -------- -------- --------- ---------     Now, FIPNUM=11 in all boundary
310    FIPNUM   11        FLUXNUM   123       /  grid blocks.
311 -- -------- -------- --------- ---------
312 /
313          The fluid-in-place regions are defined by the property FIPNUM. By
314             default, FIPNUM=1 in every cell. If we want to calculate the fluxes
315             between the reservoir and its boundaries then we have to redefine
316             the property FIPNUM in all auxiliary cells. We use the keyword
317             EQUALREG to set FIPNUM=11 in all auxiliary cells. Thus, the
318             reservoir itself is marked out as FIPNUM=2 while its boundaries
319             are marked out as FIPNUM=11.
320
321          ============================================================================
322
323 RPTGRID
324    MPINUM DEPTH PORO PERMX PERMY PERMZ MPINUM XCOORD YCOORD
325                                  I-IJKRES J-IJKRES K-IJKRES /
```

```
          We specify the properties we want to save in the file
             'EXAMPLE-H4.GRID.SUM' after the section REGIONS. The file will
             contain the properties MPINUM, DEPTH, PERMX, etc.

PROPS     ######################## PROPS section begins here ####################
          In this section, we must define the thermophysical properties of
             binary mixture, host rock as well as the relative permeability
             curves.

LOADEOS
   '../../INCLUDE/CO2H2O_V3.0.EOS' /
          The keyword LOADEOS is used for loading the mixture properties from
             the EOS-file. This keyword has as associated data the system path
             to the file.

          Rock properties are specified within brackets ROCK-ENDROCK
ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /
          The opening bracket ROCK has one associated data item, which is the
             ROCKNUM region ID. We leave it equal to 1 by default.

ROCKDH
-- grain density         heat capacity
--  [kg/m^3]              [kJ/kg/K]
      2500.                 1.0 /
          The easiest way for the rock properties specification is by using the
             keyword ROCKDH

ENDROCK   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The relative permeabilities are specified within brackets SAT-ENDSAT
SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
  /
          The opening bracket SAT has one associated data item, which is the
             SATNUM region ID. By default, leave it equal to 1.

SATTAB
--  sH2O    krH2O   krCO2
   0.100   0.0000  0.4284 /
   0.150   0.0020  0.3800 /
   0.204   0.0084  0.3292 /
   0.258   0.0281  0.2820 /      sH2O   - H2O-rich phase saturation
   0.313   0.0592  0.2382 /      krH2O  - H2O-rich phase relative permeability
   0.367   0.1017  0.1978 /      krCO2  - CO2-rich phase relative permeability
   0.421   0.1555  0.1610 /
   0.475   0.2206  0.1276 /
   0.529   0.2971  0.0976 /
   0.583   0.3850  0.0712 /
```

```
   0.638   0.4842  0.0482 /
   0.692   0.5948  0.0286 /
   0.746   0.7167  0.0126 /
   0.800   0.8500  0.0000 /
/
          The keyword SATTAB is followed by the tabular data for the relative
             permeabilites.
             The 1st column is the liquid phase saturation, which is water
             (sH2O). The 2nd column is the liquid phase relative
             permeability (krH2O). The 3rd column is the gaseous phase
             relative permeability, which is the relative permeability
             of supercritical CO2 phase.
          The relative permeabilities represent an assembly of the data from
             the files Scaled_RelPermWater.txt and Scaled_RelPermCO2.txt.

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

PHASES
-- name   pres  enth  CO2 H2O  crit._saturation
   LH2O   1.0    5.0  0.0 1.0      0.1         /
   SCO2   10.0  10.0  1.0 0.0      0.2         /
/
          The keyword PHASES defines new MNEMONICS for data output. This keyword
             defines tabular data and every row of these data defines an
             individual phase. The 1st column is the phase name (no more
             than 4 characters). The following data items defines typical
             thermodynamic parameters of the phase. The 2nd column is the
             pressure, the 3rd is the enthalpy, the 4th and 5th are the molar
             composition of binary mixture. The 6th column is the critical
             saturation of the phase, which affects the properties MM1#NAME,
             MM2#NAME, MC1#NAME, MC2#NAME. By default, this critical saturation
             is equal to 0.
          We define two phases. The 1st is the liquid water (LH2O). The 2nd is
             the supercritical CO2 phase.

RPTSATTA                                  We report relative permeabilities for
/                                         visualisation in ParaView.

INIT      ######################## INIT section begins here #####################
          In this section, we must define the initial and boundary conditions.
             We must define the required number of parameters for every cell
             associated with the simulation.

OPERAREG
-- modified argument           marker    opera    paramet  paramet  paramet
-- property property marker    value       tion      er1      er2      er3
-- -------- -------- --------- --------- -------- -------- -------- ---------
   PRES     DEPTH    SATNUM    1         MULTA    0.31     0.01              /
```

```
   TEMPC    DEPTH    SATNUM    1         MULTA    0        0.03               /
-- -------- -------- --------- --------- -------- -------- -------- ---------
/
          We use the keyword OPERAREG to specify the initial distribution of
             pressure and temperature. This keyword has tabular associated
             data. The keyword data items are as follows:
             item 1 - the modified property mnemonic;
             item 2 - the argument property mnemonic;
             item 3 - the marker mnemonic;
             item 4 - the marker value;
             item 5 - the data transformation mnemonic. This argument specifies
                which operation is applied to the data;
             items 6-8 - the parameters of the operation.
          If the operation is MULTA then MUFITS do the following:

              'modified property'='parameter1'+'parameter2'*'argument property'
              for all cells marked out as 'marker' is equal to 'marker value'.

          The first data row associated with the keyword OPERAREG defines the
             following hydrostatic distribution of the pressure:

                PRES=0.31+0.01*DEPTH.

          Here, we suppose that the water density is 1000 kg/m^3, and that
             the pressure at the reference depth 3100 meters is 31.31 MPa.
          The second data row associated with the keyword OPERAREG defines the
             following distribution of the temperature:

                TEMPC=0+0.03*DEPTH.

          Here, we suppose that the geothermal gradient is 3 degrees per 100
             meters, and that the temperature is 100 C at the reference depth
             3000 meters.

EQUALREG
--          new
-- property value    marker    marker value
-- -------- -------- --------- ---------
   COMP1T   0.0      SATNUM    1         /
-- -------- -------- --------- ---------
/
          Using the keyword EQUALREG we specify that the initial CO2 content
             is equal to 0. Thus, the reservoir is saturated by pure water.

          ============================================================================

RPTSUM
  PRESSURE TEMPC ENTHT COMP1T COMP2T
```

```
    I-IJKRES J-IJKRES K-IJKRES
        SAT#LH2O  SAT#SCO2
        CP1#LH2O  CP1#SCO2
        CP2#LH2O  CP2#SCO2
        MS1#LH2O  MS1#SCO2
        MC1#SCO2  MM1#SCO2 /
          The properties we want to save in the summary files during the
             simulation can be specified using keyword RPTSOL.
          We want to save the properties DEPTH, PHST, PRES etc.
          Note that SAT#LH2O is the saturation of the phase LH2O which was
             defined using the keyword PHASES in the section PROPS.
          Note that the keyword RPTSOL has an arbitrary number (up to 10000)
              of associated data items. The data items are the property
              mnemonics. The mnemonics with the special meaning are as follows:
                 CLEAN - targets MUFITS to delete all data items in the list
                    RPTSOL;
                 INSERT - the subsequent mnemonics are added to the list RPTSOL
                   (default option);
                 DELETE - the subsequent mnemonics are deleted from the list
                    RPTSOL.
           The input of the same type is for the keywords RPTGRID, PHYSICS and
              REPORTS in this RUN-file.

RPTWELL                                  For the well we report CO2 injection
   WMIR#1 WMIT#1 WBHP WBHTC /            rate and total, bottom-hole pressure
                                         and temperature.

RPTFIP
   PRESSURE MASS#1 MASS#2 FLUX1 FLUX2 FLUXE MS1#LH2O MS1#SCO2 MC1#SCO2 MM1#SCO2 /
          The output for the fluid-in-place regions is initiated by the keyword
             RPTFIP. MUFITS will save in the summary files:
                PRESSURE - the average pressure in every region FIPNUM;
                FLUX1    - the current flux of the 1st component between the
                           regions FIPNUM;
                FLUX2    - the current flux of the 2nd component between the
                           regions FIPNUM;
                FLUXE    - the current energy flux between the
                           regions FIPNUM;
                MS1#LH2O - the total mass of CO2 in water phase in every region
                           FIPNUM;
                MS1#SCO2 - the total mass of CO2 in CO2-RICH PHASE in every
                           region FIPNUM
                etc. (see the document "Reference manual").

SCHEDULE   ##################### SCHEDULE section begins here ####################

WEEKTOL                                  We enable option WEEKTOL (week
                                         convergence tolerances).
```

```
518
519 REPORTS                                  We add linear solver reports to the
520   LINSOL /                               reports list.
521
522 ILUTFILL                                 Linear solver setup:
523    5.0 /                                 - ilut fill factor is equal to 3.0
524 ILUTDROP                                 - ilut drop off factor is 0.001.
525    1E-4
526
527
528 VARS                                     The maximal pressure change in Newton
529    PRES DMAX 3.0 /                       iteration is 3.0 MPa.
530 /
531
532           ========================================================================
533
534           The schedule description:
535
536               name        duration                  description
537              ------------------------------------------------------------
538              Period1  -   1    year     -   hydrostatic conditions evaluation;
539              Period2  -   100  years    -   the period of injection;
540              Period3  -   2500 years    -   the post-injection period.
541
542           The injection volume is 400 Mt of CO2 in 100 years.
543           The injection rate = 10951.4031 t/day=3*3650.4677 t/day; Therefore the
                 injection rate for every source is 3650.4677 t/day.
544
545
546           The injection temperature = 80C = 353K if the pressure is equal to
                 25 MPa
547
548
549                                  The Period1 begins
550           ========================================================================
551 TUNING
552 -- timestep setup
553 -- limit   max    next   min
554     5      1000    365 /
555           The timestep control can be tuned up using the keyword TUNING. The
556              data items related to the timestep control are:
557                 item 1 - the timestep limit [days]. The next timestep can't
558                    be greater than this value;
559                 item 2 - the maximal timestep [days];
560                 item 3 - the explicitly specified next timestep [days];
561                 item 4 - the minimal timestep [days].
562
563 TSTEP
564    365.25 /
565           We proceed the simulation over 1 year (365.25 days) in the
```

```
566             simulation schedule in order to evaluate good hydrostatics.
567
568                                 The Period2 begins
569           ========================================================================
570
571 REGALL
572 EQUALREG
573 --          new
574 -- property value    marker    marker value
575 -- -------- -------- --------- ---------
576    ACTNUM   2        FLUXNUM   123       /
577 -- -------- -------- --------- ---------
578 /
579           After the hydrostatics is evaluated, we can fix parameters at the
580              boundaries. This is done by using the keyword EQUALREG. The
581              property ACTNUM is set to 2 in all cells marked out as
582              FLUXNUM=123.
583
584 WELSPECS
585 -- -------- -- ---- ---- --------
586 -- well        i-   j-   datum
587 --     name     ind  ind   depth
588 -- -------- -- ---- ---- --------       We specify well INJECTOR in the column
589    INJECTOR 1*  58   48  /              i=30, j=48.
590 -- INJECTOR 1*  64   47  /              or i=64, j=47 for the scenario INJ2.
591 -- -------- -- ---- ---- --------
592 /
593
594 COMPDAT
595 -- -------- ---- ---- ---- ---- ---- -------- -------- --------
596 -- well     i-   j-   k-   k-   mode satnum   explicit   well
597 --     name  ind  ind  min  max                 tran   diameter
598 -- -------- ---- ---- ---- ---- ---- -------- -------- --------
599    INJECTOR   1*  1*   6    10   OPEN  1*        1*      0.15    /
600 -- -------- ---- ---- ---- ---- ---- -------- -------- --------
601 /
602                                        Using the keyword COMPDAT we specify
603                                        that the CO2 injection well is
604                                        completed in layers 6,7,8,9 and 10.
605                                        The well diameter is 0.15 metres.
606
607 WELLINJE
608 -- -------- ---- ---- -------- -------- -------- -------- -------- --------
609 --   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
610 --                         rate     rate            rate              dimens
611 -- -------- ---- ---- -------- -------- -------- -------- -------- --------
612    INJECTOR OPEN MASS     0    10951.4     60      1E6 /
613 -- -------- ---- ---- -------- -------- -------- -------- -------- --------
```

```
/
                                        We the the mass injection rate is
                                        10951.4 t/day. The maximal bottom-
                                        hole pressure is 60 MPa.


EQUALNAM                                We specify that the injected CO2
  PRES    25. INJECTOR /                temperature is 80 C at 25 MPa.
  TEMPC   80           /
  COMPT#1 1.0          /                We inject pure CO2.
  COMPT#2 0.0          /
/

INITNAM                                 We recalculate primary variables based
   INJECTOR /                           on the secondary variables in the stock
                                        tank associated with the INJECTOR.

TUNING
-- timestep setup
-- limit   max    next   min
    1.     365.5   10. /

TSTEP
   50*730.5 /
          We proceed the simulation over 100 years (50*730.5 days) in the
             simulation schedule. The output is saved after every 2 years in
             the schedule.

                                 The Period3 begins
          ========================================================================

WELLSTOP                                Now, we stop the CO2 injection.
   INJECTOR /
/

TUNING
-- timestep setup
-- limit   max    next   min
    1.     1840    10. /

TSTEP
   50*3652.5 /
          We proceed the simulation over 500 years (50*3652.5 days) in the
             simulation schedule. The output is saved after every 10 years in
             the schedule.

TUNING
-- timestep setup
```

```
-- limit   max    next   min
    1*    3653    1* /

TSTEP
   100*7305.00 /
          We proceed the simulation over 2000 years (100*7305.00 days) in the
             simulation schedule. The output is saved after every 20 years in
             the schedule.

POST      ######################## POST section begins here ######################

RPTPOST
   NOTHING
   TIMEY  WMIR#1 WMIT#1 WBHP WBHTC /    We save consolidated time series report
POSTWELL                                for the well.
  '*' /
/

RPTPOST
  NOTHING TIMEY MASS#1
  MS1#LH2O MS1#SCO2 MC1#SCO2 MM1#SCO2 / We save consolidated time series report
POSTFPCE                                for the region FIPNUM=2.
  2 /
/

RPTPOST
   NOTHING TIMEY FLUX1 /                We save consolidated time series report
POSTFPCO                                for the boundary between FIPNUM regions
   2 11 /                               2 and 11.
/

CONVERT                                 We convert the output to ParaView
                                        compatible format.

END        ###############################################################################

MUFITS does not read the following records.
```

9.20. Radial flow from a CO_2 injection well (GASSTORE)

This problem is discussed in the paper: Pruess, K. et al. 2004. Code Intercomparison Builds Confidence in Numerical Simulation Models for Geologic Disposal of CO_2. Energy, 29(9-10): 1431-1444. DOI:10.1016/j.energy.2004.03.077. See a description of the problem, input data files and visualisations under *the link*.

GASSTORE-TEST2.RUN

```
################################################################################

                        RADIAL FLOW FROM A CO2 INJECTION WELL

    The problem is from the paper:
        Pruess, K. et al. 2004. Code Intercomparison Builds Confidence in
        Numerical Simulation Models for Geologic Disposal of CO2. Energy,
        29(9-10): 1431-1444. DOI:10.1016/j.energy.2004.03.077.
        Pruess, K. et al. 2002. Multiphase flow dynamics during CO2 injection
        into saline aquifers// Envirom. Geol. 42, 282-295.
        DOI:10.1007/s00254-001-0498-3

    See description at:
        http://dx.doi.org/10.1016/j.energy.2004.03.077;
        http://dx.doi.org/10.1007/s00254-001-0498-3
        http://esd1.lbl.gov/files/research/projects/tough/
                            documentation/TOUGH2-ECO2N_V2.0_Users_Guide.pdf

--------------------------------------------------------------------------------

    Last update on 12.11.2015
    Program Version: 2015.G01

RUNSPEC   ################### RUNSPEC section begins here ######################

METRIC    ^^^^^^^^^^^^ We use METRIC units ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
                                       Note, this keywords affects the units
                                       in many keywords below.

GASSTORE  ^^^^^^ We switch the program to the GASSTORE EOS module ^^^^^^^^^^^^^

                                       We enable
NACL                                   - salt component;
CO2-SEQ                                - CO2 component;
WATER                                  - water component;
```

```
DISGAS                                  - CO2 dissolution in brine;
VAPWAT                                  - water vapour in the gas phase.

CAPPRES                                 We enable capillary pressure.
ROCKCOMP                                We enable rock compressibility.

ISOTHERM                                We use the Isothermal modelling option.

FAST                                    We use FAST option.

AUXARRAY                                We introduce an auxiliary array
  #SIMVAR /                             #SIMVAR, where we will store values of
/                                       the similarity variable.

GRID      ###################### GRID section begins here ######################

          The grid is specified within brackets MAKE-ENDMAKE
MAKE      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
-- cartesian                            We select Radial gridding
--    grid   nr   ntheta  nz            option and specify the number of
     RADIAL 1270    1     1   /         grid blocks along every direction.

RTZBOUND
  0 6000  -0.0314159 0.0314159 0 100 /
                                        We specify the domain extent in meters
                                        along R and Z axes, and in radians
                                        along Theta direction. Thus, we
                                        consider the 0.01 fraction of the
                                        full circle.

DRV
  1100*1
  1  2  3  4  5  6  7  8  9  10
  11 12 13 14 15 16 17 18 19 20
  20 20 30 40 50 60 70 80 90 100
  110  120  130  140  150  160  170  180  190  200
  200  200  300  400  500  600  700  800  900  1000
  1100 1200 1300 1400 1500 1600 1700 1800 1900 2000
  2000 2000 3000 4000 5000 6000 7000 8000 9000 10000
  10*10000
  10*20000
  10*30000
  10*40000
  10*50000
  10*60000
  10*70000
  10*80000
  10*90000
```

```
  10*100000 /                          Using the DRV keyword we specify the
                                       grid blocks extensions along axis R.
                                       We have a uniform grid for the first
                                       1100 meters, and farther the extension
                                       of the grid blocks increases.


SRCSPECG                               We introduce a point source INJECTOR
  'INJECTOR' 1 1 1 /                   in the grid block i=1, j=1, k=1.
/

ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

EQUALS                                 We equal
  PORO   0.12  /                        porosity to 0.12;
  PERMI  100  /                        permeablity to 100 mD.
/

MULTIPLY                               Because we specified grid only for
   CELLVOL   100.0 /                   0.01 fraction of the full circle we
   AREA      100.0 /                   multiply cell volumes and interface
/                                      areas (both in R and Z directions)
                                       by the factor 100.

PROPS     ############################################################################

INCLUDE                                We load the properties of CO2 from
  'CO2TAB.METRIC.INC' /                include file.

          Rock properties are specified within brackets ROCK-ENDROCK
ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
/
ROCKECL
-- -refpres- --compr--                 We specify the reference pressure (bar)
      120      4.5E-5 /                 and the rock compressibility (1/bar).

ENDROCK   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The relative permeabilities and capillary pressure are specified
          within brackets SAT-ENDSAT
SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
/

SGLF
-- sgas     krgas          krliq          pcgl
   0.00     0.0            1.0            0.0          /  We define the relative
   0.01     0.0            0.6284139343   0.0248270428 /  permeability of the
   0.025    0.0            0.4690103837   0.0413776570 /  gas phase (krgas), the
```

```
   0.05     0.0            0.3243551786   0.0616685866 /  relative permeability
   0.1      0.0008753194   0.1754738386   0.0942244008 /  of the liquid (krliq)
   0.2      0.0217429361   0.0560863397   0.1525231868 /  and gas-liquid
   0.3      0.0919085466   0.0163414095   0.2147871648 /  capillary pressure
   0.4      0.2281782851   0.0036709332   0.2901918260 /  (pcgl) as the function
   0.5      0.4339133784   0.0004812265   0.3906042421 /  of the gas saturation
   0.6      0.6990301461   0.0000159286   0.5385130956 /  (sgas).
   0.7      1.0            0.0            0.7873987963 /
/

SSFN
-- ssol  krf                            We define the reduction factor for
   0.0   1.0 /                          the rock permeability (krf) as the
   0.01  0.9 /                          the faction of the solid phase
   0.05  0.7 /                          saturation (ssol).
   0.1   0.5 /
   0.2   0.0 /
/

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

INIT      ######################## INIT section begins here #####################

EQUALS                               We equal in the whole domain:
  PRES   120  /                      - pressure to 120 bar;
  TEMPC  45   /                      - temperature to 45 C;
  SLIQ   1.0  /                      - saturation of the liquid phase to 1;
  XSM    0.15 /                      - salt mass fraction in brine to 0.15;
  XGM    0.0  /                      - gas mass fraction in brine to 0.
/

EQUALS                               We fix parameters in the grid block
  ACTNUM 2 1270 /                    i=1270, j=1, k=1. (The farthest block
/                                    from the well.

                                     We specify parameters in the stock tank
EQUALNAM                             associated with the INJECTOR source:
  PRES    120 'INJECTOR' /           - pressure to 120 bar;
  TEMPC   45  /                      - temperature to 45 C;
  SGAS    1.0 /                      - saturation of the gas phase to 1;
  YWM     0.0 /                      - water vapour mass fraction in the
/                                      gas phase to 0.

                                     Here, we calculate the values of the
                                     similarity variable corresponding to
OPERATE                              the time 1000 days (8.64e7 seconds):
   #SIMVAR XCOORD    COPY /          - #SIMVAR:=XCOORD;
   #SIMVAR #SIMVAR  MULTARR /        - #SIMVAR:=#SIMVAR*#SIMVAR;
```

```
   #SIMVAR 1*        DIVIDE  8.64E7 /   - #SIMVAR:=#SIMVAR/8.64E7.
/

RPTSUM
  #SIMVAR PRES TEMPC XGM YWM
  SLIQ SGAS XSMR SSOL ASCII/            We report these properties in the grid
                                        blocks;

RPTSRC                                  We report these properties for the
  SMIR#1 SMIR#2 SMIT#1 SMIT#2 /         INJECTOR. The 1st component is WATER,
                                        the 2nd component is CO2.

SCHEDULE   ##################### SCHEDULE section begins here ###################

REPORTS                                 No reports in the LOG-file.
  NOTHING /

VARS                                    The maximal alteration of pressure
  PRES DMAX  10 /                       in a Newton iteration is 10 bars.
/

SRCINJE                                 We specify that the injecton rate is
  'INJECTOR' MASS  1*  400 1* 8640 /    8640 ton/day=100 kg/sec, and the
/                                       maximum pressure in the grid block i=1,
                                        j=1, k=1 is 400 bars.

TUNING                                  The maximun and next time step is
   4*0.005 /                            0.005 days.

TIME                                    We advance simulation to t=0.3 days.
   0.3 /

TUNING                                  We increase the maximum time step to
  1* 0.05 /                             0.05 days.

TIME                                    We advance simulation to t=3 days.
   1 3 /

TUNING                                  We increase the maximum time step to
  1* 0.5 /                              0.5 days.

TIME                                    We advanxe simulation to t=20 days.
 10 20 /

TUNING                                  We increase the maximum time step to
  1* 2 /                                2 days.

TIME                                    We advance simulation to 300 days.
```

```
   50 100 300 /

TUNING                                  We increase the maximum time step to
  1* 5 /                                5 days.

TIME                                    We advance simulation to 1000 days.
   500 1000 /

POST      ######################## POST section begins here ######################

CONVERT                                 We convert the output to ParaView
                                        compatible format.

RPTPOST                                 We define the properties output from the
  TIME SMIR#1 SMIR#2 SMIT#1 SMIT#2 /    following POSTSRC keyword.
POSTSRC                                 We save consolidated time series
  INJECTOR /                            report for the INJECTOR.
/

END       ###############################################################################
```

9.21. Heat sweep in a vertical fracture (T2EOS1)

This problem is discussed in the paper: Pruess K. et al. 1984 Thermal effects of reinjection in geothermal reservoirs with major vertical fractures. J. Pet. Tech. 36(10): 1567-1578. DOI:10.2118/12099-PA. See a description of the problem, input data files and visualisations under *the link*.

T2EOS1-TEST1.RUN

```
################################################################################

                        HEAT SWEEP IN A VERTICAL FRACTURE

    The problem is from SPE-12099-PA paper:
    Pruess K. et al. 1984 Thermal effects of reinjection in geothermal
       reservoirs with major vertical fractures. J. Pet. Tech. 36(10):
       1567-1578. DOI:10.2118/12099-PA

    See description at: http://dx.doi.org/10.2118/12099-PA

--------------------------------------------------------------------------------

    Last update on 17.05.2015
    Program Version: 2015.C08

################################################################################

--------------------------------------------------------------------------------

   Note:
   We type all comments in lowercase letters while all keywords and mnemonics
      must be in uppercase letters.

--------------------------------------------------------------------------------

   Note:
-- any line beginning with '!' or '--' is a comment line;
   any line not beginning with an uppercase letter outside keyword
!  instruction is also a comment line.

--------------------------------------------------------------------------------

   Note:
   Any data line must be terminated by the slash '/' sign.

--------------------------------------------------------------------------------

```

```
   Note:
   We denote by the repeated symbols the following structural elements of this
      RUN-file:
      '###############' - delimits the sections of the RUN-file.
      '<<<<<<<<<<<<<<' - designates an opening bracket.
      '>>>>>>>>>>>>>>' - designates a closing bracket.
      '^^^^^^^^^^^^^^' - designates the keyword, which affects the order of the
                         data input for all the following keywords (both in the
                         current and in the following sections).
      '==============' - merge the keywords and the comments in paragraphs.

------------------------------------------------------------------------------


RUNSPEC   ################### RUNSPEC section begins here ######################

T2EOS1    ^^^^^^ We switch the program to the T2EOS1 EOS module ^^^^^^^^^^^^^^^^

                                       We enable:
HCROCK                                  - heat conduction in rock material;
HCFLUID                                 - heat conduction in reservoir fluid.

STARTTIM                               The start time is -365.25 days.
  -365.25 /

GRID      ###################### GRID section begins here ######################

          The grid is specified within brackets MAKE-ENDMAKE
MAKE      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
-- cartesian                           We select Cartesian gridding
--    grid   nx  ny  nz                option and specify the number of
      CART   12  20  10  /             grid blocks along every axis.


XYZBOUND
-- xmin-xmax   ymin-ymax   zmin-zmax    We specify the domain extent in meters.
   0    240        2*      0    200 /   The reservoir is height is 200 meters.

DYV
  0.04 0.01 0.02 0.04 0.08 0.1 0.2 0.4 0.6 1 2 3 5 8 10 12 16 20 30 50 /
                                       We specify the grid blocks extensions
                                       along Y axis. The grid is refined close
                                       to the Y=0 m plane.

          We futher assume that the grid blocks
           - j=1 correspond to the fracture;
           - j=2,20 correspond to confining impermeable rocks (semi-analytical
             method is not used for to describe heat conduction in confining
```

```
            rocs).

BOX                                     We make active j=[2,20] region of the
  2* 2 /                                domain.
TYPENUM                                 We set agrid blocks j=[2,20]
  1000000*2 /                           impermeable where only heat conduction
ENDBOX                                  equation is solved.

SRCSPECG                                We intriduce point source named INJE in
   INJE  1 1 8 /                        the grid block i=1, j=1, k=8.
/

WELSPECS
-- -------- -- ---- ---- --------       We introduce well PRODUCER with wellhead
-- well        i-   j-   datum          in the column i=12, j=1. The datum depth
--     name        ind  ind   depth        for the bottom-hole pressure is 50
-- -------- -- ---- ---- --------       meters.
   PRODUCER 1*  12   1    50      /
-- -------- -- ---- ---- --------
/

COMPDAT
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
-- well     i-   j-   k-   k-   mode satnum   explicit   well
--     name  ind  ind  min  max                 tran   diameter
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
   PRODUCER   1*   1*  3    3   OPEN    1*      4E-12     1*    /
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
/
                                        Using keyword COMPDAT we complete the
                                        the well only in the layer 3 (the well
                                        is communicating only with the layer 3).
                                        The productivity index is 4e-12 m3.

ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

EQUALS                                  We equal in the whole domain:
  PORO      0.5    /                    - porosity to 0.5;
  PERMX     200000 /                    - permeability 200000 mD;
  HCONDCFX  2.1    /                    - rock heat conduction coefficient
/                                         along axis X to 2.1 W/m/K.

COPY                                    We copy:
  PERMX    PERMZ /                      - X axis permeability into Y axis perm.
  HCONDCFX HCONDCFY /                   - X axis heat conduction coefficient
  HCONDCFX HCONDCFZ /                     into Y and Z axis heat cond. coeff.
/

```

```
     The following two keywords allow us to take into account that confining
     rocks are on either side of the fracture, whereas the grid corresponds
     to situation when the confining rocks are only one  side of the fracture.
     The heat losses in the confining rocks are two-times underestimated subject
     the used grid.
     To make the simulation correct, we will multiply the impermeable grid block
     volumes and Y axis heat conduction coefficient by the factor 2.


MULTIREG                                We multiply impermeable grid block
  CELLVOL 2.0 TYPENUM 2 /               volumes by the factor 2.
/

MULTIPLY                                We multiply Y direction rock heat
  HCONDCFY 2.0 /                        conduction coefficient by the factor 2.
/

RPTGRID
  PORO PERMX PERMZ
  HCONDCFX HCONDCFY HCONDCFZ /          We report from the GRID section
                                        porosity, permeabilities and rock heat
                                        conduction coefficient.

PROPS     ######################## PROPS section begins here ####################

          Rock properties are specified within brackets ROCK-ENDROCK
ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

ROCKDH                                  We specify that
  2650  1.0 /                           rock density is 2650 kg/m3,
                                        rock heat capacity is 1.0 kJ/kg/K

ENDROCK  >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

         The relative permeabilities are specified within brackets SAT-ENDSAT
SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

SATTAB
-- swat krwat krvap                    The water relative permeability (krwat)
   0.0  0.0   1.0  /                   and vapor relative permeability (krvap)
   1.0  1.0   0.0  /                   are linear functions of water
/                                       saturation (swat).

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

INIT      ######################## INIT section begins here #####################
```

```
183
184 EQUALS                                  We equal in the whole domain
185   PRES   10  /                          - pressure to 10 MPa;
186   TEMPC  300 /                          - temperature to 300 C.
187 /
188
189 EQUALNAM
190   PRES  10   'INJE' /                   We specify that the injected fluid
191   TEMPC 100         /                   temperature is 100 C at 10 MPa.
192 /
193
194 EQUALREG                                 We set FIPNUM=10 in all 'fracture' grid
195   FIPNUM 10 TYPENUM 1 /                  blocks.
196 /
197                                         We save at every report step:
198 RPTSUM                                  - pressure, temperatue and water
199   PRES TEMPC SWAT/                        saturation;
200 RPTSUM                                  - j -index;
201   J-IJKRES /
202 RPTFIP                                  - average pressure;
203   PRES /
204 RPTSRC                                  - point source mass injection rate and
205   SMIR#1 SMIT#1 /                         source mass injection total;
206 RPTWELL                                 - well bottom-hole temperature and
207    WBHTC WBHP WMPR#1 WMPT#1 /             pressure, mass production rate and
208                                           mass production total.
209
210 SCHEDULE   ##################### SCHEDULE section begins here ###################
211
212 TSTEP                                   We advance simulation to 0 days to
213   365.25 /                              evaluate hydrostatic equilibrium. The
214                                         injection/production is not active
215                                         during this period.
216
217 SRCINJE                                 We specify that point source injection
218   INJE MASS 1* 20. 1* 345.6 /           rate is 345.6 ton/day=4 kg/sec, the
219 /                                       maximal grid block pressure is 20 MPa.
220
221 WELLPROD                                We specify that production well is
222   PRODUCER OPEN BHP 2* 9.65 /           operating at constant bottom hole
223 /                                       pressure 9.65 MPa.
224
225 TUNING                                  The next time step is 0.1 days.
226   2* 0.1 /
227
228 TSTEP                                   We advance simulation to 5 years.
229   10*3.6525 49*36.525 /
230
```

```
POST      ######################## POST section begins here ######################

CONVERT                                  We convert the output to ParaView
                                         compatible format.

RPTPOST
  TIME TEMPC /                           We save consolidated time series report
POSTBLOC                                 for the grid block i=12, j=1, k=3.
  12 1 3 /                               Temperature in the grid block is
/                                        reported.

RPTPOST
  NOTHING TIME PRES /                    We save consolidated time series
POSTFPCE                                 report for average pressure in fracture.
  10 /
/

RPTPOST
  NOTHING TIME WBHTC
          WBHP WMPR#1 WMPT#1 /           We save consolidated time series report
POSTWELL                                 for the production well parameters.
  PRODUCER /
/

RPTPOST
  NOTHING TIME SMIR#1 SMIT#1 /           We save consolidated time series report
POSTSRC                                  for the point source parameters.
  'INJE' /
/

END       ###############################################################################
```

9.22. Five-spot geothermal production/injection (T2EOS1)

This problem is discussed in the paper: Pruess K. et al. A practical method for modelling fluid and heat flow in fractured porous media// SPE. J. 1985. 25(1). 14-26. DOI: 10.2118/10509-PA. See a description of the problem, input data files and visualisations under *the link*.

T2EOS1-TEST2-250.RUN

```
################################################################################

                    FIVE-SPOT GEOTHERMAL PRODUCTION/INJECTION

    The problem is from the paper:
    Pruess, K. and Narasimhan, T.N. A practical method for modelling fluid
       and heat flow in fractured porous media// SPE. J. 1985. 25(1). 14-26.
       DOI: 10.2118/10509-PA.

    See description at: http://dx.doi.org/10.2118/10509-PA

    This is the second scenario in that the distance between fractures is 250
    meters.

--------------------------------------------------------------------------------

    Last update on 12.05.2017
    Program Version: 2017.A01

################################################################################

RUNSPEC   ################### RUNSPEC section begins here ######################

T2EOS1    ^^^^^^^^^^^^^ We switch the program to T2EOS1 module ^^^^^^^^^^^^^^^^

METRIC    ^^^^^^^^^^^^^^^^^^^^^ We use METRIC units ^^^^^^^^^^^^^^^^^^^^^^^^^^

HCROCK                                  We activate heat conduction term for
                                        rocks.
HCFLUID                                 We activate heat conduction tem for
                                        fluid.

DUALPORO                                We activate the double porosity option.

MINC                                    We activate the MINC option. The maximum
   4  SPHERICAL  0.08 0.20 0.35 0.35 /  number of nested matrix media per
```

```
                                        grid block is 4. The geometry of the
                                        nested media is SPHERICAL. The
                                        fractional volume of the first MINC
                                        media (closest to frature media) is
                                        0.08. The fractional volume of the
                                        second media is 0.20, etc. The
                                        simulator normalizes these numbers to
                                        make its sum equal to 1.

FAST                                    We activate the FAST option in order to
                                        accelerate the simulation run.

AUXARRAY                                We create an auxiliary array named
  #XCOORD /                                    #XCOORD.
/

GRID      ###################### GRID section begins here ######################

          The grid is specified within brackets MAKE-ENDMAKE
MAKE      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
-- cartesian                            We select Cartesian gridding
--    grid   nx  ny  nz                 option and specify the number of
      CART   11  11   1  /              grid blocks along every axis. Note,
                                        that because of the DUALPORO keyword
                                        the program doubles the nz parameter.
                                        The grid blocks iz=1 correspond to the
                                        matrix media, and the grid blocks iz=2
                                        correspond to the fractures media.

XYZBOUND                                We specify that that xmin=0 meters,
-- xmin-xmax  ymin-ymax  zmin-zmax      ymin=0 meters, the top surface of the
  0.0  707.1  0.0  707.1 1000 1305. /   reservoir is at 1000 meters depth. The
                                        reservoir thickness is 305 meters. The
                                        number zmin=1000 is irrelevant for the
                                        simulation results.

DXV                                     We specify grid blocks extension along
  35.355 9*70.71 35.355 /               X axis.
DYV                                     We specify grid blocks extension along
  35.355 9*70.71 35.355 /               Y axis

TYPENUM                                 We specify that the matrix grid blocks
   121*2 /                              are impermeable in that only heat
                                        conduction is simulated.

WELSPECS
-- -------- -- ---- ---- --------
-- well        i-   j-   datum         We specify four wells in the corners of
```

```
--     name     ind  ind   depth        the domain.
-- -------- -- ---- ---- --------
   PROD1    1*  1    1   /
   PROD2    1*  11   11  /
   INJE1    1*  1    11  /
   INJE2    1*  11   1   /
-- -------- -- ---- ---- --------
/

COMPDAT
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
-- well     i-   j-   k-   k-   mode satnum   explicit   well
--     name  ind  ind  min  max                 tran   diameter
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
   PROD1     1*   1*   2    2   OPEN  1*         1E4    /
   PROD2     1*   1*   2    2   OPEN  1*         1E4    /
   INJE1     1*   1*   2    2   OPEN  1*         1E4    /
   INJE2     1*   1*   2    2   OPEN  1*         1E4    /
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
/                                    We specify that all well are completed
                                     only in fractures media. Since the PI
                                     index is not specified in the problem
                                     statement, we set it to an arbitrary
                                     large number 1E4. This number is high
                                     enough that the BHP pressure does not
                                     significantly differ from pressure in
                                     the corresponding block.

DZWELL                               This keyword specifies parameters for
  150 50 /                           wells visualization in ParaView.

ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

EQUALS
-- -------- -------- -- ---------
-- mnemonic value       kmin-kmax
-- -------- -------- -- ---------
   PORO     0.5      4*    2*2   /     By using k index (k=2) we refer to
   PERMX    6.0                  /     fracture media.
-- -------- -------- -- ---------
   LX       250.     4*    2*1   /     By using k index (k=1) we refer to
-- -------- -------- -- ---------      matrix media (all 4 matrix media).
/
                                     Here, we specify that
                                     - porosity in fractures is 0.5;
                                     - X direction permeability in
                                       fractures is 6 mD.;
                                     - fractures spacing along X axis is
```

```
                                          250 meters.

MULTIPLY
-- -------- -------- -- ---------
-- mnemonic value       kmin-kmax
-- --------- -------- -- ---------
   PERMX     1.01325  4*  2*2     /    We multiply permeability by 1.01325 in
-- --------- -------- -- ---------     order to make it equal to 6e-15 m^2.
   CELLVOL   0.98     4*  2*1     /    We specify that the fractional volume of
   CELLVOL   0.02     4*  2*2     /    matrix media is 0.98 of the original
-- --------- -------- -- ---------     grid block volume. We specify that the
/                                      fractional volume of fractures is 0.02
                                       of the original grid block volume.

EQUALS                                 We specify that rock heat conductivity
   HCONDCFX 2.1 /                      along X direction is 2.1 W/m/K both in
/                                      fractures and matrix.

COPY
-- --------- -------- -- ---------
-- mnemfrom  mnemto      kmin-kmax      We copy:
-- --------- -------- -- ---------     in fractures:
   PERMX     PERMY    4*  2*2     /       X-permeability into Y-permeability
   PERMX     PERMZ                /       X-permeability into Z-permeability
-- --------- -------- -- ---------     in matrix:
   LX        LY       4*  2*1     /       X-spacing into Y-spacing of fractures
   LX        LZ                   /       X-spacing into Z-spacing of fractures
-- --------- -------- -- ---------     both in fractures and in matrix:
   HCONDCFX  HCONDCFY 4*  1 2     /       X-heat conductivity into Y-conduct-ty
   HCONDCFX  HCONDCFZ 4*  1 2     /       X-heat conductivity into Z-conduct-ty
-- --------- -------- -- ---------
/

RPTGRID
  TYPENUM PORO PERMX PERMY PERMZ
  LX LY LZ SIGMAV CELLVOL /            We report these properties from the
                                       grid section. Note, the MINCNUM array
                                       is by default reported in double
                                       porosity simulations.

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! IMPORTANT !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
          Here, MUFITS creates the MINC media. In farther sections the index
             k=1 refers to all matrix media (MINCNUM=1,2,3,4);
             k=2 refers to fractures (MINCNUM=0);
             k=3 refers to the firts matrix media (MINCNUM=1);
             k=4 refers to the second matrix media (MINCNUM=2);
             k=5 refers to the third matrix media (MINCNUM=3);
             k=6 refers to the fourth matrix media (MINCNUM=4).
```

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

PROPS     ####################### PROPS section begins here ####################

          Rock properties are specified within brackets ROCK-ENDROCK
ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

ROCKDH                                  We specify that
  2650  1.0 /                           rock density is 2650 kg/m3,
                                        rock heat capacity is 1.0 kJ/kg/K

ENDROCK   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The saturation functions are specified within brackets SAT-ENDSAT
SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
  /

SATTLIQ
-- swat   krwat      krvap
   0.30   0.0        1.0        /       We specify Corey relative permeablity
   0.35   0.000035   0.847029   /       functions with swat-min=0.3 and
   0.40   0.000560   0.699030   /       svap-min=0.05. The first column is
   0.45   0.002836   0.560204   /       the water saturation (swat). The
   0.50   0.008963   0.433913   /       second column is the water relative
   0.55   0.021882   0.322677   /       permeability (krwat), and the third
   0.60   0.045376   0.228178   /       column is the vapor relative
   0.65   0.084065   0.151255   /       permeability (krvap).
   0.70   0.143412   0.091908   /
   0.75   0.229718   0.049297   /
   0.80   0.350127   0.021742   /
   0.85   0.512622   0.006722   /
   0.90   0.726024   0.000875   /
   0.95   1.0        0.0        /
/

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

INIT      ####################### INIT section begins here #####################

EQUALS                                  We equate in all grid blocks (both
-- -------- --------                     matrix and fracture)
   TEMPC    300     /                   - temperature to 300 C;
   SWAT     0.99    /                   - water saturation to 0.99.
-- -------- --------
/

EQUALNAM                                We speicfy parameters of the injected
```

```
-- ------- -------- --------           fluid. Total enthalpy of the fluid is
   PRES     85.93   INJE*    /         0.5 MJ/kg at 85.93 bar.
   ENTHMTOT 0.5              /
-- ------- -------- --------
/

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

COPY                                  Here, within <<< >>> brackets, we
-- --------- -------- -- ---------     calculate the #XCOORD auxiliary data.
   XCOORD    #XCOORD  4*  1 2     /    This data is relevant only for
-- --------- -------- -- ---------     postprocessing of the results in the
/                                     POST section. It does not influence
                                      simulation results.
EQUALS
-- -------- -------- ---------
   #XCOORD  0.       2*1  4*  /
   #XCOORD  707.1    2*11 4*  /
-- -------- -------- ---------
/

MULTIPLY
-- -------- -------- -- ---------
   #XCOORD  -1       4*  1 2     /
-- -------- -------- -- ---------
/

ADD
-- -------- -------- -- ---------
   #XCOORD  707.1    4*  1 2     /
-- -------- -------- -- ---------
/
>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

RPTSUM                                We report these properties for every
   PRES TEMPC SWAT SVAP #XCOORD /     grid block (both matrix and fracture)
                                      at every report time.
RPTWELL
   WBHP   WBHTC  WMPR#1 WMPT#1
   WMIR#1 WMIT#1  /                   We report these properties for the
                                      wells.

SCHEDULE   ##################### SCHEDULE section begins here ####################

REPORTS
   NOTHING /                          No calculation/convergence reports in
                                      in the LOG file.

```

```
WELLPROD
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
--   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
--                        rate     rate            rate              dimens
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
   PROD*    OPEN MASS   1*       648      1.0       1E6 /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
/                                       We specify that each production well
                                        produces fluid at the rate 648 ton/day
                                        (7.5 kg/sec) at the minimum bottom-
                                        hole pressure 1.0 bar. This pressure
                                        constraint is not reached in the
                                        simulation.

WELLINJE
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
--   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
--                        rate     rate            rate              dimens
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
   INJE*    OPEN MASS   1*       648      200       1E6 /
-- -------- ---- ---- -------- -------- -------- -------- -------- --------
/                                       We specify that each injection well
                                        injects fluid at the rate 648 ton/day
                                        (7.5 kg/sec) at the maximum bottom-
                                        hole pressure 200 bar. This pressure
                                        constraint is not reached in the
                                        simulation.

TUNING                                  The maximum time step is 30 days.
  1* 30 /

TSTEP                                   We simulate the process reporting every
  73*182.625 /                          182.625 days (0.5 year).


POST      ######################## POST section begins here ######################

CONVERT                                 We convert the simulation results to
                                        ParaView format.

RPTPOST                                 We create report for every production
  TIME WBHP WBHTC WMPR#1 WMPT#1 /       well.
POSTWELL
   PROD* /
/

RPTPOST
  NOTHING TIME WBHP WBHTC
```

```
  WMIR#1 WMIT#1 /                          We create report for every injection
POSTWELL                                   well.
   INJE* /
/

RPTPOST
   NOTHING #XCOORD TEMPC /
POSTSPEC                                   We save temperature distribution in
   1* FRACTURE.CSV  73 /                   the fractures media (k=2)
POSTBLK                                    between injector and producer at
   11  1  2 /                             the final simulation time (report
   10  1  2 /                             step #73).
    9  1  2 /
    8  1  2 /
    7  1  2 /
    6  1  2 /
    5  1  2 /
    4  1  2 /
    3  1  2 /
    2  1  2 /
    1  1  2 /
/

END        ###############################################################################
```

9.23. Test for heat conduction

See a description of the problem, input data files and visualisations under *the link*.

HC.RUN

```
################################################################################

                            TEST FOR HEAT CONDUCTION

    This is the self-similar one-dimensional discontinuity break-down problem
    for the heat conduction equation. The fluid flow is not considered in this
    problem. The break-down problem has analytical solution which can be used
    for testing purposes.

--------------------------------------------------------------------------------

    Last update on 16.05.2015
    Program Version: 2015.C08

################################################################################

   The break-down problem description:
   ===================================

   Rock properties:
       Porosity:          0
       Density:           R=2500 kg/m3
       Heat capacity:     C=1 kJ/kg/K
       Head conductivity: L=2 W/m/K

   Domain extension: 100 meters,   x=[-50,50] meters.

   Initial conditions (for temperature):
       T=Tl=300 (K) if x<0;
       T=Tr=400 (K) if x>0.

   Analytical solution:
   ====================

       T(x,t)=(Tl+Tr)/2+(Tr-Tl)/2*erf(x/2/sqrt(a*t)),   a=L/R/C

   Temperatures at 1000 days obtained using the analytical solution:

                         Coordinate (m) | Temperature (K)
                         -------------------------------
                                2.05    |  356.920
                                4.05    |  363.474
```

```
                                6.05    |  369.657
                                8.05    |  375.322
                               10.05    |  380.366
                               20.05    |  395.593
                             ------------------------------


RUNSPEC   ################### RUNSPEC section begins here ######################

HCROCK                                  We enable heat conduction processes in
                                        rocks.

NOCASC                                  We disable cascade method.

GRID      ##################### GRID section begins here ######################

          The grid is specified within brackets MAKE-ENDMAKE
MAKE      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
-- cartesian                            We select Cartesian gridding
--    grid   nx  ny  nz                 option and specify the number of
      CART    1   1  1000  /           grid blocks along every axis.
                                        The grid is 1*1*1000 grid blocks. We
                                        locate domain along Z-axis for using
                                        the property DEPTH as the location
                                        coordinate.

XYZBOUND
-- xmin-xmax   ymin-ymax   zmin-zmax    We specify the domain extent in meters.
   0    10     0    10     -50  50 /    The reservoir extension alon Z-axis is
                                        [-50,50] meters.

TYPENUM                                 We specify that all 1000 grid blocks
   1000*2 /                             are impermeable in which only heat
                                        conduction equation is solved.

ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

EQUALS                                  We equal rock heat conductivity to
  HCONDCF 2 /                           2.0 W/m/K.
/

PROPS     ###################### PROPS section begins here ####################

          Rock properties are specified within brackets ROCK-ENDROCK
ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /
ROCKDH                                  We equal rock density to 2500 kg/m3
   2500.  1.0 /                         and rock heat capacity to 1 kJ/kg/K.
```

```

ENDROCK   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

INIT      ####################### INIT section begins here #####################

EQUALS
   TEMP 300  4*  1   500  /            We equal T=300 K in the region z<0;
   TEMP 400  4*  501 1000 /            We equal T=400 K in the region z>0.
/

RPTSUM                                 We specify that grid blocks center depth
   ASCII DEPTH TEMP /                  (DEPTH) and temperature (TEMP) are saved
                                       in the summary files at every report
                                       step. The formated output is requested
                                       (ASCII).

SCHEDULE   #################### SCHEDULE section begins here ###################

TUNING                                 The initial and maximal time step is
   3*1 /                               a day.

TSTEP                                  We advance the simulation to 100 days.
   100 /

TUNING                                 Further the maximal time step is 10
   1*  10 /                            days.

TSTEP                                  We advance simulation to 1000 days
   9*100 /                             reporting every 100 days.

POST      ####################### POST section begins here #####################

RPTPOST
   DEPTH TEMP /
POSTSPEC
   1* TEMP-TO-COMPARE.CSV  10 /        We report temperature in selected grid
POSTBLK                                blocks at 1000 days to compare with the
    1  1  521 /                        analytical solution. The output file is
    1  1  541 /                        TEMP-TO-COMPARE.CSV.
    1  1  561 /
    1  1  581 /
    1  1  601 /
    1  1  701 /
/

CONVERT                                We convert the output to ParaView
                                       compatible format.
```

```
139 END             ###########################################################################
```

9.24. CO_2 discharge along a fault zone (GASSTORE)

This problem is discussed in the paper: Pruess, K. et al. 2002. Multiphase flow dynamics during CO_2 injection into saline aquifers// Envirom. Geol. 42, 282-295. DOI:10.1007/s00254-001-0498-3. See a description of the problem, input data files and visualisations under *the link*.

GASSTORE-TEST3.RUN

```
################################################################################

                          CO2 DISCHARGE ALONG A FAULT ZONE

    The problem is from the paper:
        Pruess, K. et al. 2002. Multiphase flow dynamics during CO2 injection
        into saline aquifers// Envirom. Geol. 42, 282-295.
        DOI:10.1007/s00254-001-0498-3

    See description at:
        http://dx.doi.org/10.1007/s00254-001-0498-3
        http://esd1.lbl.gov/files/research/projects/tough/
                            documentation/TOUGH2-ECO2N_V2.0_Users_Guide.pdf

--------------------------------------------------------------------------------

    Last update on 12.11.2015
    Program Version: 2015.G01

RUNSPEC   ################### RUNSPEC section begins here ######################

METRIC    ^^^^^^^^^^^^ We use METRIC units ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
                                       Note, this keywords affects the units
                                       in many keywords below.

GASSTORE  ^^^^^^ We switch the program to the GASSTORE EOS module ^^^^^^^^^^^^^^

                                       We enable
CO2-SEQ                                - CO2 component;
WATER                                  - water component;
DISGAS                                 - CO2 dissolution in brine;
VAPWAT                                 - water vapour in the gas phase.

CAPPRES                                We enable capillary pressure.
ROCKCOMP                               We enable rock compressibility.

ISOTHERM                               We use the Isothermal modelling option.

```

```
FAST                                    We use FAST option.

AUXARRAY                                We create an auxiliary array for
  #ELEVTN /                             elevation which will be defined in the
/                                       GRID section.

GRID      ###################### GRID section begins here ######################

          The grid is specified within brackets MAKE-ENDMAKE
MAKE      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
-- cartesian                            We select Cartesian gridding
--    grid   nx  ny  nz                 option and specify the number of
      CART   1   1   100  /             grid blocks along every axis.

XYZBOUND
  0 1  0 25  1000 1500 /                We specify the domain extent in meters.
                                        The top boundary is at the depth 1000 m,
                                        the bottom bounary is at the depth
                                        1500 meters.

BOUNDARY                                We create the auxiliary blocks for
  2 6* K- 5* INFTHIN 1* 3* 1* 2 /       - the top boundary (with FLUXNUM=2);
  3 6* K+ 5* INFTHIN 1* 3* 1* 2 /       - the bottom boundary (with FLUXNUM=3).
/                                       Parameters in all these blocks are
                                        fixed (ACTNUM=2).

ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

EQUALS                                  We equal
  PORO   0.35  /                          porosity to 0.35;
  PERMZ  100   /                          permeablity to 100 mD.
/

MULTIPLY                                We multiply permeability by 0.9869 thus
  PERMZ  0.9868 /                       converting units from mD to m^2.
/

COPY
  ZCOORD #ELEVTN /                      We calculate the values of the auxiliary
/                                       array.
ADD
  #ELEVTN 1500 /
/

REGALL
EQUALREG                                We will use different relative
  SATNUM 3 FLUXNUM 3 /                  permeabilities at the bottom boundary.
/                                       Here, we set SATNUM=3 in the boundary
```

```
                                    grid blocks at the bottom boundary.

PROPS     ----------------------------------------------------------------------

INCLUDE                             We load the properties of CO2 from
  'CO2TAB.METRIC.INC' /             include file.

          Rock properties are specified within brackets ROCK-ENDROCK
ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
/
ROCKECL
-- -refpres- --compr--              We specify the reference pressure (bar)
      148      4.5E-5 /             and the rock compressibility (1/bar).

ENDROCK   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

SATOPT                              We use a different method for
  2* 1 /                            interpolation of relative permeabilities

          The relative permeabilities and capillary pressure are specified
          within brackets SAT-ENDSAT
SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
/

SGLF
--     sgas             krgas             krliq             pcgl
   0.0000000000000   0.0000000000000   1.0000000000000   0.0000000000000 /
   0.0000010000000   0.0000000000000   0.9939098509000   0.0000000000000 /
   0.0000100000000   0.0000000000000   0.9826015192000   0.0000000000000 /
   0.0001000000000   0.0000000000000   0.9505288472000   0.0000000000000 /
   0.0010000000000   0.0000000000000   0.8612718342000   0.0000000000000 /
   0.0050000000000   0.0000000000000   0.7212560377000   0.0182670819400 /
   0.0100000000000   0.0000000000000   0.6284139343000   0.0285511460500 /
   0.0200000000000   0.0000000000000   0.5116452375000   0.0433853488800 /
   0.0300000000000   0.0000000000000   0.4325412049000   0.0552929644800 /
   0.0500000000000   0.0000000000000   0.3243551786000   0.0754894759700 /
   0.0510000000000   0.0000000072771   0.3200441375000   0.0764254231300 /
   0.0520000000000   0.0000000581716   0.3158127124000   0.0773563603800 /
   0.0550000000000   0.0000009068310   0.3035724543000   0.0801210460300 /
   0.0600000000000   0.0000072266377   0.2845589409000   0.0846454691700 /
   0.0750000000000   0.0001116032354   0.2360310158000   0.0977498078100 /
   0.1000000000000   0.0008753194912   0.1754738386000   0.1187135430000 /
   0.1500000000000   0.0067224536910   0.0992570417300   0.1602824964000 /
   0.2000000000000   0.0217429361800   0.0560863397600   0.2046326457000 /
   0.2500000000000   0.0492979937700   0.0309335737500   0.2549401556000 /
   0.3000000000000   0.0919085466200   0.0163414095300   0.3148784924000 /
   0.3500000000000   0.1512552081000   0.0081014546990   0.3897547913000 /
   0.4000000000000   0.2281782851000   0.0036709332850   0.4882724450000 /
```

```
    0.4500000000000   0.3226777774000   0.0014626982680   0.6262919938000 /
    0.5000000000000   0.4339133784000   0.0004812265543   0.8364145747000 /
    0.5500000000000   0.5602044746000   0.0001163376876   1.1977621530000 /
    0.6000000000000   0.6990301461000   0.0000159286397   1.9611024110000 /
    0.6300000000000   0.7869788313000   0.0000027861053   3.0087418390000 /
    0.6500000000000   0.8470291657000   0.0000005390436   4.4958829830000 /
    0.6600000000000   0.8773748259000   0.0000001814561   5.8647078270000 /
    0.6700000000000   0.9078844017000   0.0000000445979   8.2586192380000 /
    0.6800000000000   0.9385189033000   0.0000000061729  13.3745016900000 /
    0.6850000000000   0.9538704492000   0.0000000015177  18.8266918100000 /
    0.6900000000000   0.9692379958000   0.0000000002101  30.4813222200000 /
    0.6920000000000   0.9753883212000   0.0000000000708  39.7364527400000 /
    0.6930000000000   0.9784640229000   0.0000000000369  46.5690697600000 /
    0.6940000000000   0.9815400274000   0.0000000000174  55.9300878300000 /
    0.6950000000000   0.9846162914000   0.0000000000072  69.4591893800000 /
    0.6955000000000   0.9861545075000   0.0000000000043  78.7225222900000 /
    0.6958000000000   0.9870774608000   0.0000000000031  85.4478908400000 /
    0.6960000000000   0.9876927723000   0.0000000000024  90.5479158600000 /
    0.6962000000000   0.9883080908000   0.0000000000019  96.2381399100000 /
    0.6963210000000   0.9886803616000   0.0000000000016  100              /
    0.7               1.0               0.0              100              /
/
                                          By using the SGLF keyword we define
                                          the relative permeability of gas
                                          (krgas), the relative permeability of
                                          brine (krliq), the capillary pressure
                                          (pcgl) as the functions of the gas
                                          saturation (sgas).

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   3 /    We use a different set of relative permeabilities at the bottom
          boundary.

SGLF
--       sgas              krgas              krliq              pcgl
    0.0000000000000   0.0000000000000   1.0000000000000   0.0000000000000 /
    0.0000010000000   0.0000000000000   0.9939098509000   0.0000000000000 /
    0.0000100000000   0.0000000000000   0.9826015192000   0.0000000000000 /
    0.0001000000000   0.0000000000000   0.9505288472000   0.0000000000000 /
    0.0010000000000   0.0000000000000   0.8612718342000   0.0000000000000 /
    0.0050000000000   0.0000000000000   0.7212560377000   0.0000000000000 /
    0.0100000000000   0.0000000000000   0.6284139343000   0.0000000000000 /
    0.0200000000000   0.0000000000000   0.5116452375000   0.0000000000000 /
    0.0300000000000   0.0000000000000   0.4325412049000   0.0000000000000 /
    0.0500000000000   0.0000000000000   0.3243551786000   0.0000000000000 /
    0.0510000000000   0.0000000072771   0.3200441375000   0.0000000000000 /
```

```
    0.0520000000000   0.0000000581716   0.3158127124000   0.0000000000000 /
    0.0550000000000   0.0000009068310   0.3035724543000   0.0000000000000 /
    0.0600000000000   0.0000072266377   0.2845589409000   0.0000000000000 /
    0.0750000000000   0.0001116032354   0.2360310158000   0.0000000000000 /
    0.1000000000000   0.0008753194912   0.1754738386000   0.0000000000000 /
    0.1500000000000   0.0067224536910   0.0992570417300   0.0000000000000 /
    0.2000000000000   0.0217429361800   0.0560863397600   0.0000000000000 /
    0.2500000000000   0.0492979937700   0.0309335737500   0.0000000000000 /
    0.3000000000000   0.0919085466200   0.0163414095300   0.0000000000000 /
    0.3500000000000   0.1512552081000   0.0081014546990   0.0000000000000 /
    0.4000000000000   0.2281782851000   0.0036709332850   0.0000000000000 /
    0.4500000000000   0.3226777774000   0.0014626982680   0.0000000000000 /
    0.5000000000000   0.4339133784000   0.0004812265543   0.0000000000000 /
    0.5500000000000   0.5602044746000   0.0001163376876   0.0000000000000 /
    0.6000000000000   0.6990301461000   0.0000159286397   0.0000000000000 /
    0.6300000000000   0.7869788313000   0.0000027861053   0.0000000000000 /
    0.6500000000000   0.8470291657000   0.0000005390436   0.0000000000000 /
    0.6600000000000   0.8773748259000   0.0000001814561   0.0000000000000 /
    0.6700000000000   0.9078844017000   0.0000000445979   0.0000000000000 /
    0.6800000000000   0.9385189033000   0.0000000061729   0.0000000000000 /
    0.6850000000000   0.9538704492000   0.0000000015177   0.0000000000000 /
    0.6900000000000   0.9692379958000   0.0000000002101   0.0000000000000 /
    0.6920000000000   0.9753883212000   0.0000000000708   0.0000000000000 /
    0.6930000000000   0.9784640229000   0.0000000000369   0.0000000000000 /
    0.6940000000000   0.9815400274000   0.0000000000174   0.0000000000000 /
    0.6950000000000   0.9846162914000   0.0000000000072   0.0000000000000 /
    0.6955000000000   0.9861545075000   0.0000000000043   0.0000000000000 /
    0.6958000000000   0.9870774608000   0.0000000000031   0.0000000000000 /
    0.6960000000000   0.9876927723000   0.0000000000024   0.0000000000000 /
    0.6962000000000   0.9883080908000   0.0000000000019   0.0000000000000 /
    0.6963210000000   0.9886803616000   0.0000000000016   0.0000000000000 /
    0.7               1.0               0.0               0.0000000000000 /
/
                                          By using the SGLF keyword we define
                                          the relative permeability of gas
                                          (krgas), the relative permeability of
                                          brine (krliq), the capillary pressure
                                          (pcgl) as the functions of the gas
                                          saturation (sgas).

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

INIT      ####################### INIT section begins here #####################

                                          Here, we define parameters at the
REGALL                                    bottom boundary. We change EQLNUM thus
EQUALREG                                  the following EQL-ENDEQL will not affect
  EQLNUM 3  FLUXNUM 3 /                   the boundary grid block.
```

```
  PRES  240 /                          - the pressure is 240 bar;
  TEMPC  45 /                          - the temperature is 45 C;
  SGAS  1.0 /                          - the gas saturation is 1.0;
  YWM   0.0 /                          - the water mass fraction in gas is 0.
/

          The initial equilibrium is specified within brackets EQL-ENDEQL
EQL       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
/

EQUILGS
-- -------- -------- --------          Using the keyword EQUILGS we specify that
-- datum    datum     GWC               - pressure is 100 bar at the depth 1000
--   depth  pressure   depth               meters;
-- -------- -------- --------          - gas-water contact is above the top
     1000     100     500    /            boundary (at depth 500 m).
-- -------- -------- --------

TEMPCVD
  1000 45 /                            We specify that the temperature is 45C
  1500 45 /                            in the domain.
/

XGMVD
  1000 0.0 /                           We specify that the gas mass fraction in
  1500 0.0 /                           brine is 0.0 in the domain
/

YWMVD
   0  0.0 /                           We specify that the water mass fraction
 500  0.0 /                           in gas is 0 in the gas zone above the
/                                     gas-water contact.

ENDEQL    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

RPTSUM
  PRES TEMPC SGAS SLIQ XGM YWM
  #ELEVTN /                            We report these properties for the grid
                                       blocks.

REGALL
COPYREG
  FLUXNUM FIPNUM ROCKNUM 1/            We copy FLUXNUM in the FIPNUM.
/

RPTFIP                                 We report fluxes between different
  FLUX#1 FLUX#2 /                      FIPNUM regions.

```

```
279
280 SCHEDULE   ##################### SCHEDULE section begins here ###################
281
282 REPORTS                                  No reports in the LOG-file.
283   NOTHING /
284
285 VARS                                     The maximal alteration of pressure
286   PRES DMAX  30 /                       in a Newton iteration is 10 bars.
287 /
288
289 TUNING
290  1* 0.0001 1* 0.00001 /                 Here, we gradually increase timestep
291 TIME                                     advanxing simulation further in time.
292  0.001 0.002 0.004 /
293
294 TUNING
295  1* 0.001 /
296 TIME
297   0.01 0.02 0.04 0.1 /
298
299 TUNING
300  1* 0.01 /
301 TIME
302 0.2 0.4 1 /
303
304 TUNING
305  1* 0.1 /
306 TIME
307    2 3 4 5 6 7 8 9 10 11.574 /
308
309 TUNING
310  1* 1 /
311 TIME
312  15 20 30 40 50 60 70 80 90 100 115.74  /
313
314 TUNING
315   1* 10 /
316 TIME
317   150 200 300 400 500 600 700 800 900 1000 1157.4 /
318
319 TUNING
320   1* 100 /
321 TIME
322   1500 2000 3000 4000 5000 6000 7000 8000 9000 10000 11574 /
323
324 TUNING
325   1* 200 /
326 TIME
```

```
  15000 20000 30000 40000 /

TUNING
  1* 1000 /
TIME
  50000 60000 70000 80000 90000 100000 115740 /

POST      ######################## POST section begins here ######################

CONVERT                                  We convert the output to ParaView
                                         compatible format.

RPTPOST
  TIMESEC FLUX#1 FLUX#2 /                We save consolidated files for the
POSTFPCO                                 fluxes
  1 2 /                                  - at the top boundary;
  3 1 /                                  - at the bottom boundary.
/

RPTPOST
 NOTHING #ELEVTN SGAS PRES XGM YWM TIME /
/
POSTSPEC                                 We output distributions at time 1e7 sec.
  1* GASSTORE-TEST3.TIME1E7S.CSV 31 /
POSTBLK
  1 1 100 /
  1 1 99  /
  1 1 98  /
  1 1 97  /
  1 1 96  /
  1 1 95  /
  1 1 94  /
  1 1 93  /
  1 1 92  /
  1 1 91  /
  1 1 90  /
  1 1 89  /
  1 1 88  /
  1 1 87  /
  1 1 86  /
  1 1 85  /
  1 1 84  /
  1 1 83  /
  1 1 82  /
  1 1 81  /
  1 1 80  /
  1 1 79  /
  1 1 78  /
```

```
  1 1 77  /
  1 1 76  /
  1 1 75  /
  1 1 74  /
  1 1 73  /
  1 1 72  /
  1 1 71  /
  1 1 70  /
  1 1 69  /
  1 1 68  /
  1 1 67  /
  1 1 66  /
  1 1 65  /
  1 1 64  /
  1 1 63  /
  1 1 62  /
  1 1 61  /
  1 1 60  /
  1 1 59  /
  1 1 58  /
  1 1 57  /
  1 1 56  /
  1 1 55  /
  1 1 54  /
  1 1 53  /
  1 1 52  /
  1 1 51  /
  1 1 50  /
  1 1 49  /
  1 1 48  /
  1 1 47  /
  1 1 46  /
  1 1 45  /
  1 1 44  /
  1 1 43  /
  1 1 42  /
  1 1 41  /
  1 1 40  /
  1 1 39  /
  1 1 38  /
  1 1 37  /
  1 1 36  /
  1 1 35  /
  1 1 34  /
  1 1 33  /
  1 1 32  /
  1 1 31  /
  1 1 30  /
```

```
  1 1 29  /
  1 1 28  /
  1 1 27  /
  1 1 26  /
  1 1 25  /
  1 1 24  /
  1 1 23  /
  1 1 22  /
  1 1 21  /
  1 1 20  /
  1 1 19  /
  1 1 18  /
  1 1 17  /
  1 1 16  /
  1 1 15  /
  1 1 14  /
  1 1 13  /
  1 1 12  /
  1 1 11  /
  1 1 10  /
  1 1  9  /
  1 1  8  /
  1 1  7  /
  1 1  6  /
  1 1  5  /
  1 1  4  /
  1 1  3  /
  1 1  2  /
  1 1  1  /
/

POSTSPEC
  1* GASSTORE-TEST3.TIME1E9S.CSV 53 / We output distributions at time 1e9 sec.
POSTBLK
  1 1 100 /
  1 1 99  /
  1 1 98  /
  1 1 97  /
  1 1 96  /
  1 1 95  /
  1 1 94  /
  1 1 93  /
  1 1 92  /
  1 1 91  /
  1 1 90  /
  1 1 89  /
  1 1 88  /
  1 1 87  /
```

```
  1 1 86  /
  1 1 85  /
  1 1 84  /
  1 1 83  /
  1 1 82  /
  1 1 81  /
  1 1 80  /
  1 1 79  /
  1 1 78  /
  1 1 77  /
  1 1 76  /
  1 1 75  /
  1 1 74  /
  1 1 73  /
  1 1 72  /
  1 1 71  /
  1 1 70  /
  1 1 69  /
  1 1 68  /
  1 1 67  /
  1 1 66  /
  1 1 65  /
  1 1 64  /
  1 1 63  /
  1 1 62  /
  1 1 61  /
  1 1 60  /
  1 1 59  /
  1 1 58  /
  1 1 57  /
  1 1 56  /
  1 1 55  /
  1 1 54  /
  1 1 53  /
  1 1 52  /
  1 1 51  /
  1 1 50  /
  1 1 49  /
  1 1 48  /
  1 1 47  /
  1 1 46  /
  1 1 45  /
  1 1 44  /
  1 1 43  /
  1 1 42  /
  1 1 41  /
  1 1 40  /
  1 1 39  /
```

```
519   1 1 38  /
520   1 1 37  /
521   1 1 36  /
522   1 1 35  /
523   1 1 34  /
524   1 1 33  /
525   1 1 32  /
526   1 1 31  /
527   1 1 30  /
528   1 1 29  /
529   1 1 28  /
530   1 1 27  /
531   1 1 26  /
532   1 1 25  /
533   1 1 24  /
534   1 1 23  /
535   1 1 22  /
536   1 1 21  /
537   1 1 20  /
538   1 1 19  /
539   1 1 18  /
540   1 1 17  /
541   1 1 16  /
542   1 1 15  /
543   1 1 14  /
544   1 1 13  /
545   1 1 12  /
546   1 1 11  /
547   1 1 10  /
548   1 1  9  /
549   1 1  8  /
550   1 1  7  /
551   1 1  6  /
552   1 1  5  /
553   1 1  4  /
554   1 1  3  /
555   1 1  2  /
556   1 1  1  /
557 /
558
559 END       ###########################################################################
```

9.25. CO_2 injection into a 2D-layered brine formation (GASSTORE)

This problem is discussed in the paper: Pruess, K. et al. 2004. Code Intercomparison Builds Confidence in Numerical Simulation Models for Geologic Disposal of CO_2. Energy, 29(9-10): 1431-1444. DOI:10.1016/j.energy.2004.03.077. See a description of the problem, input data files and visualisations under *the link*.

GASSTORE-TEST4.RUN

```
################################################################################

               CO2 INJECTION IN A 2D-LAYERED BRINE FORMATION

    The problem is from the paper:
        Pruess, K. et al. 2004. Code Intercomparison Builds Confidence in
        Numerical Simulation Models for Geologic Disposal of CO2. Energy,
        29(9-10): 1431-1444. DOI:10.1016/j.energy.2004.03.077.

    See description at:
        http://dx.doi.org/10.1016/j.energy.2004.03.077;
        http://esd1.lbl.gov/files/research/projects/tough/
                            documentation/TOUGH2-ECO2N_V2.0_Users_Guide.pdf

    Note:
        According to the formulation of the problem, the permeability at
        the interface between every two grid blocks is upstream weighted.
        MUFITS uses a harmonic weighted permeability which is a conventional
        approximation for an industrial simulator. Therefore, we redefine
        the permeability in the vertical direction by the MULTZ- multiplier
        assuming that the flow is in the upward direction.

--------------------------------------------------------------------------------

    Last update on 12.11.2015
    Program Version: 2015.G01

RUNSPEC   ################### RUNSPEC section begins here ######################

METRIC    ^^^^^^^^^^^^ We use METRIC units ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
                                       Note, this keywords affects the units
                                       in many keywords below.

GASSTORE  ^^^^^^ We switch the program to the GASSTORE EOS module ^^^^^^^^^^^^^^

```

```
                                        We enable
NACL                                    - salt component;
CO2-SEQ                                 - CO2 component;
WATER                                   - water component;
DISGAS                                  - CO2 dissolution in brine;
VAPWAT                                  - water vapour in the gas phase.

CAPPRES                                 We enable capillary pressure.

ISOTHERM                                We use the Isothermal modelling option.

FAST                                    We use FAST option.

START                                   The date of beginning of the CO2
  1 OCT 1996 /                          injection is 01 october, 1996.

AUXARRAY                                We enable 2 auxiliary arrays
  #ELEVTN /                             - elevation;
  #XCOORD /                             - x coordinate.
/                                       Both arrays will be defined in the GRID
                                        section.

GRAVITY                                 The gravity acceleration is 9.81 m/sec^2
   9.81 /

GRID      ###################### GRID section begins here ########################

          The grid is specified within brackets MAKE-ENDMAKE
MAKE      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
-- cartesian                            We select Cartesian gridding
--    grid   nx  ny  nz                  option and specify the number of
      CART   29   1  34  /               grid blocks along every axis.

XYZBOUND
  2*  0 1  858 1000 /                   The top surface of the formation is at
                                        the depth 858 meters. The number 1000
                                        is redefined using the following
                                        keyword DZV. The domain extension
                                        along axis Oy is 1 meter.

DXV
   1   1   2   4   4   8  15  20  30  40
  50  50 100 150  50 150 300  50 475 500
 500 500 500 500 500 500 500 500 1E-3 / Here, we specify the extensions of the
                                        grid blocks along axis Ox.

DZV
   3   6  12   6   3   3   3   6  12   6
```

```
   3   3   3   6  12   6   3   3   3   6
  12   6   3   3   3   6  12   6 2.5   1
 2.5   6   6   7 /                      Here, we specify the thickness of
                                        every layer.


SRCSPECG
  'INJECTOR' 1 1 30 /                   The injection well is modelled by a
/                                       point source (named INJECTOR) which is
                                        located in the grid block i=1, j=1,
                                        k=30.

ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

EQUALS                                  By default, the SATNUM number is 1 in
  SATNUM 2  4*  2*6  /                  all grid blocks. Here, we equal SATNUM
  SATNUM 2  4*  2*12 /                  to 2 in the shale layers which are
  SATNUM 2  4*  2*18 /                  k=6; k=12; k=18 and k=24.
  SATNUM 2  4*  2*24 /
/

EQUALREG
-- -------- -------- -------- --------  We equal porosity to
   PORO     0.35     SATNUM   1 /       -  0.35 in the sand layers;
   PORO     0.1025   SATNUM   2 /       -  0.1025 in the shales.
-- -------- -------- -------- --------
/

EQUALREG
-- -------- -------- -------- --------  We equal permeability to
   PERMX    3000     SATNUM   1 /       - 3000 mD in the sand layers (X-dir)
   PERMZ    3000     SATNUM   1 /       - 3000 mD in the sand layers (Z-dir)
   PERMX    10       SATNUM   2 /       - 10 mD in the shales (X-dir)
   PERMZ    3000     SATNUM   2 /       - 3000 mD in the shales (Z-dir)
   MULTZ-  0.0033333 SATNUM   2 /       - we multiply the above value by
-- -------- -------- -------- --------    0.003333.
/

MULTIPLY                                We multiply permeability by 0.9869 thus
  PERMX 0.9869 /                        converting units from mD to m^2.
  PERMZ 0.9869 /
/

EQUALS
  ACTNUM 2  2*29 /                      We fix parameters in the boundary
/                                       grid blocks i=29.

COPY
```

```
  ZCOORD #ELEVTN /
  XCOORD #XCOORD /                    Here, we define the values of two
/                                     auxiliary arrays which are used for
ADD                                   data output.
  #ELEVTN 1042 /
/

RPTGRID                               We report these properties from the
  PORO PERMX PERMZ ACTNUM SATNUM /    GRID section.

PROPS     ######################## PROPS section begins here ####################

INCLUDE                               We load the properties of CO2 from
  'CO2TAB.METRIC.INC' /               include file.

          Rock properties are specified within brackets ROCK-ENDROCK
ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
/
ROCKDH                                The rock density is 2600 kg/m3, and the
   2600  0.92 /                       heat capacity is 0.92 kJ/kg/K. These
                                      numbers do not affect the simulation
                                      since it is running in the Isothermal
                                      mode.
ENDROCK   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

SATOPT                                We use a different method for
  2* 1 /                              interpolation of relative permeabilities

          The relative permeabilities and capillary pressure are specified
          within brackets SAT-ENDSAT
SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
/         Here, we specify the saturation functions for the region SATNUM=1,
          i.e. for the sand layers.

SGLF
--     sgas            krgas            krliq            pcgl
   0.000000000000   0.000000000000   1.000000000000   0.000000000000 /
   0.000001000000   0.000000000000   0.987479385400   0.000000000000 /
   0.000010000000   0.000000000000   0.968695014700   0.000000000000 /
   0.000100000000   0.000000000000   0.922270476300   0.000000000000 /
   0.001000000000   0.000000000000   0.810412236900   0.000000000000 /
   0.005000000000   0.000000000000   0.655485851400   0.002598035231 /
   0.010000000000   0.000000000000   0.560372644900   0.004254391500 /
   0.020000000000   0.000000000000   0.446701011000   0.006751168135 /
   0.030000000000   0.000000000000   0.372772184500   0.008820526801 /
   0.050000000000   0.000000000000   0.275111549000   0.012425808540 /
   0.051000000000   0.000000004738   0.271296445200   0.012595260470 /
   0.052000000000   0.000000037875   0.267557283600   0.012763989140 /
```

```
   0.055000000000   0.000000590617   0.256771407600   0.013266131240 /
   0.060000000000   0.000004709136   0.240105789900   0.014091145200 /
   0.075000000000   0.000072839506   0.198049174000   0.016501192060 /
   0.100000000000   0.000572839507   0.146500089300   0.020408807050 /
   0.150000000000   0.004424691356   0.082996814130   0.028285745690 /
   0.200000000000   0.014400000000   0.047583886410   0.036787448310 /
   0.250000000000   0.032869135810   0.026991883090   0.046453149880 /
   0.300000000000   0.061728395060   0.014912902340   0.057894785670 /
   0.350000000000   0.102400000000   0.007909588050   0.071951695230 /
   0.400000000000   0.155832098800   0.003962729625   0.089897683190 /
   0.450000000000   0.222498765400   0.001837449554   0.113817678700 /
   0.500000000000   0.302400000000   0.000766497053   0.147393851800 /
   0.550000000000   0.395061728400   0.000275630707   0.197750575000 /
   0.600000000000   0.499535802400   0.000079621867   0.280452234600 /
   0.650000000000   0.614400000000   0.000016203809   0.436070193400 /
   0.700000000000   0.737758024800   0.000001732079   0.805851789800 /
   0.750000000000   0.867239506100   0.000000038169   2.285560899000 /
   0.770000000000   0.920125440000   0.000000002298   4.919830283000 /
   0.780000000000   0.946704086800   0.000000000247   9.039249414000 /
   0.785000000000   0.960015840000   0.000000000051  13.917259450000 /
   0.788000000000   0.968008126500   0.000000000015  19.450202320000 /
   0.790000000000   0.973338042500   0.000000000005  25.568110170000 /
   0.792000000000   0.978669080900   0.000000000002  35.732680940000 /
   0.793000000000   0.981334951900   0.000000000001  43.657037670000 /
   0.794000000000   0.984001019900   0.000000000000  55.014231000000 /
   0.794500000000   0.985334119200   0.000000000000  62.684207110000 /
   0.795000000000   0.986667257200   0.000000000000  72.318162360000 /
   0.795300000000   0.987467157300   0.000000000000  79.351605310000 /
   0.795500000000   0.988000430700   0.000000000000  84.700076910000 /
   0.795700000000   0.988533709200   0.000000000000  90.677587800000 /
   0.795850000000   0.988933671300   0.000000000000  95.638012850000 /
   0.795971600000   0.989257909100   0.000000000000  100.0           /
   0.8              1.0              0.0             100.0           /
/
```

By using the SGLF keyword we define the relative permeability of gas (krgas), the relative permeability of brine (krliq), the capillary pressure (pcgl) as the functions of the gas saturation (sgas).

```
SSFN
-- ssol  krf
   0.0   1.0 /
   0.5   1.0 /
   1.0   0.0 /
/
```

We define the reduction factor for the rock permeability (krf) as the the faction of the solid phase saturation (ssol). Since krf=1 for small ssol the precipitation of salt does not affect the relative

```
                                        permeabilities of the fluid phases.

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
  2 /     Here, we specify the saturation functions for the region SATNUM=1,
          i.e. shales.

SGLF
--      sgas             krgas            krliq            pcgl
   0.000000000000   0.000000000000   1.000000000000   0.000000000000 /
   0.000001000000   0.000000000000   0.987479385400   0.000000000000 /
   0.000010000000   0.000000000000   0.968695014700   0.000000000000 /
   0.000100000000   0.000000000000   0.922270476300   0.000000000000 /
   0.001000000000   0.000000000000   0.810412236900   0.000000000000 /
   0.005000000000   0.000000000000   0.655485851400   0.044993906240 /
   0.010000000000   0.000000000000   0.560372644900   0.073679405870 /
   0.020000000000   0.000000000000   0.446701011000   0.116919671600 /
   0.030000000000   0.000000000000   0.372772184500   0.152757726700 /
   0.050000000000   0.000000000000   0.275111549000   0.215195566900 /
   0.051000000000   0.000000004738   0.271296445200   0.218130209300 /
   0.052000000000   0.000000037875   0.267557283600   0.221052326000 /
   0.055000000000   0.000000590617   0.256771407600   0.229748641600 /
   0.060000000000   0.000004709136   0.240105789900   0.244036592800 /
   0.075000000000   0.000072839506   0.198049174000   0.285774834600 /
   0.100000000000   0.000572839507   0.146500089300   0.353448613700 /
   0.150000000000   0.004424691356   0.082996814130   0.489864869500 /
   0.200000000000   0.014400000000   0.047583886410   0.637101060100 /
   0.250000000000   0.032869135810   0.026991883090   0.804495891800 /
   0.300000000000   0.061728395060   0.014912902340   1.002647126000 /
   0.350000000000   0.102400000000   0.007909588050   1.246090811000 /
   0.400000000000   0.155832098800   0.003962729625   1.556887251000 /
   0.450000000000   0.222498765400   0.001837449554   1.971144157000 /
   0.500000000000   0.302400000000   0.000766497053   2.552630952000 /
   0.550000000000   0.395061728400   0.000275630707   3.424730628000 /
   0.600000000000   0.499535802400   0.000079621867   4.856994007000 /
   0.650000000000   0.614400000000   0.000016203809   7.552053629000 /
   0.700000000000   0.737758024800   0.000001732079  13.956092450000 /
   0.730000000000   0.814883524000   0.000000243083  23.876580870000 /
   0.740000000000   0.840983040000   0.000000104078  30.101074860000 /
   0.750000000000   0.867239506100   0.000000038169  39.582339580000 /
   0.755000000000   0.880419040000   0.000000021379  46.365704030000 /
   0.758000000000   0.888341397500   0.000000014627  51.424821340000 /
   0.760000000000   0.893628649900   0.000000011184  55.331915820000 /
   0.763000000000   0.901567542900   0.000000007284  62.199801390000 /
   0.765000000000   0.906865183100   0.000000005365  67.609094800000 /
   0.768000000000   0.914818697200   0.000000003277  77.339833350000 /
   0.769000000000   0.917471646000   0.000000002752  81.113294900000 /
```

```
   0.770000000000   0.920125440000   0.000000002298  85.203764680000 /
   0.772000000000   0.925435459500   0.000000001572  94.496320770000 /
   0.773035000000   0.928184612100   0.000000001278  100.0           /
   0.8              1.0              0.0              100.0           /
/
                                        By using the SGLF keyword we define
                                        the relative permeability of gas
                                        (krgas), the relative permeability of
                                        brine (krliq), the capillary pressure
                                        (pcgl) as the functions of the gas
                                        saturation (sgas).

SSFN
-- ssol  krf                            We define the reduction factor for
   0.0   1.0 /                          the rock permeability (krf) as the
   0.5   1.0 /                          the faction of the solid phase
   1.0   0.0 /                          saturation (ssol). Since krf=1 for
/                                       small ssol the precipitation of salt
                                        does not affect the relative
                                        permeabilities of the fluid phases.

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

RPTSATTA                                We report the realtive permeabilities
1 1* 1000 /                             for the SATNUM=1 region.
RPTSATTA                                We report the relative permeabilities
2 1* 1000 /                             for the SATNUM=2 region.

INIT      ####################### INIT section begins here #####################

          The initial equilibrium is specified within brackets EQL-ENDEQL
EQL       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
/

EQUILGS
-- -------- -------- --------
-- datum    datum     GWC
--   depth  pressure   depth
-- -------- -------- --------
   1020     110        500 /
-- -------- -------- --------

                    Using the keyword EQUILGS we specify that
                    - pressure is 110 bar at the injection point at depth
                      1020 meters;
                    - gas-water contact is above the top layer
                      (at depth 500 m).

```

```
TEMPCVD
   800   37  /      We specify that the temperature is 37C in the formation.
   1000  37  /
/

XGMVD
   800   4.54104E-4 /      We specify that the initial gas mass fraction in brine
   1000  4.54104E-4 /      is 4.54104E-4.
/

XSMVD
   800   0.032 /           We specify that the initial salt mass fraction in
   1000  0.032 /           brine is 0.032.
/

YWMVD
   800   0.0 /      We specify that the initial water mass fraction in gas
   1000  0.0 /      is 0 (this data must be specified because the datum depth
/                   differs from the GWC depth).

ENDEQL    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

EQUALNAM                                We specify that the injected CO2
  PRES   110  INJECTOR /                temperature is 37C at 110 bar.
  TEMPC  37   /
  SGAS   1.0  /
  YWM    0.0  /                         No water vapour in gas.
/


RPTSUM
  PRES TEMPC XGM YWM SLIQ SGAS XSM SSOL
  FLUX#1 FLUX#2 #ELEVTN #XCOORD PCWG /  We report these properties for the grid
                                        blocks.

RPTSRC                                  We report these properties for the
  SMIR#2 SMIT#2 /                       INJECTOR.

RPTFIELD                                We report these properties for the
  FGIPM FGIPGM FGIPLM /                 field (formation).

SCHEDULE   ###################### SCHEDULE section begins here ###################

REPORTS
  NOTHING /                             No reports in LOG file.

VARS
  PRES DMAX 30 /                        Maximum pressure change is 30 bars.
```

```
/

ILUTFILL                                  Parameters of the linear solver.
  3 /
ILUTDROP
  1E-3 /

SRCINJE                                   We specify that the injection rate is
  'INJECTOR' MASS  1*  200 1* 13.6944 / 13.6944 ton/day=0.1585 kg/sec. The
/                                         maximum pressure in the "well" grid
                                          block (1,1,30) is 200 bars (this value
                                          is not reached during the simulation).

TUNING                                    The maximal time step is 2.5 days.
  1* 2.5 /

TIME
  0.001 0.0025 0.005 0.0075 0.01 0.015
  0.025 0.05 0.075 0.1 0.15 0.25 0.5 0.75 /  We report parameters at these
                                             times since beginning of the
                                             injection.

DATES
  2 OCT 1996 /
  3 OCT 1996 /
  4 OCT 1996 /
  6 OCT 1996 /
  8 OCT 1996 /
 15 OCT 1996 /
 23 OCT 1996 /
  1 NOV 1996 /
  8 NOV 1996 /
 15 NOV 1996 /
 23 NOV 1996 /
  1 DEC 1996 /
  8 DEC 1996 /                            We report parameters at these
 15 DEC 1996 /                            dates. The total simulation
 23 DEC 1996 /                            time is 2 years.
  1 JAN 1997 /
 15 JAN 1997 /
  1 FEB 1997 /
 15 FEB 1997 /
  1 MAR 1997 /
 15 MAR 1997 /
  1 APR 1997 /
  1 MAY 1997 /
  1 JUN 1997 /
  1 JUL 1997 /
```

```
  1 AUG 1997 /
  1 SEP 1997 /
  1 OCT 1997 /
  1 NOV 1997 /
  1 DEC 1997 /
  1 JAN 1998 /
  1 FEB 1998 /
  1 MAR 1998 /
  1 APR 1998 /
  1 MAY 1998 /
  1 JUN 1998 /
  1 JUL 1998 /
  1 AUG 1998 /
  1 SEP 1998 /
  1 OCT 1998 /
/

POST      ######################## POST section begins here ######################

CONVERT                                 We convert the output to ParaView
                                        compatible format.

RPTPOST
  TIMESEC PRES SLIQ SGAS SSOL #ELEVTN / We save these parameters variation
POSTBLOC                                with time in the grid blocks:
  1 1 30 /                              (1,1,30);
  2 1 30 /                              (2,1,30);      (i,j,k)
  1 1 29 /                              (1,1,29);
  1 1 31 /                              (1,1,31);
  1 1 19 /                              (1,1,19).
/

RPTPOST
  NOTHING TIMESEC FGIPM FGIPGM FGIPLM
  DAY MONTH YEAR /                      We save consolidated report for the
POSTFLD                                 field data.
/

RPTPOST
  NOTHING TIMESEC SMIR#2 SMIT#2 /       We save consolidated report for the
POSTSRC                                 INJECTOR.
  INJECTOR /
/

RPTPOST
  NOTHING  TIMESEC FLUX#1 FLUX#2 /
POSTBCON                                We save reports for the fluxes
   1 1 30 2*    2 1 30 /                  from (1,1,30) into (2,1,30);
```

```
   1 1 30 2*    1 1 29 /                  from (1,1,30) into (1,1,29);
   1 1 30 2*    1 1 31 /                  from (1,1,30) into (1,1,31);
   1 1 29 2*    1 1 30 /                  from (1,1,29) into (1,1,30).
/

POSTSPEC
  1* GASSTORE-TEST4.SAT-X500.CSV 54 /
RPTPOST
  NOTHING #ELEVTN SGAS #XCOORD /        We save distribution of gas at 500 m
POSTBLK                                 distance from the axis of symmetry.
  15 1 34 /
  15 1 33 /
  15 1 32 /
  15 1 31 /
  15 1 30 /
  15 1 29 /
  15 1 28 /
  15 1 27 /
  15 1 26 /
  15 1 25 /
  15 1 24 /
  15 1 23 /
  15 1 22 /
  15 1 21 /
  15 1 20 /
  15 1 19 /
  15 1 18 /
  15 1 17 /
  15 1 16 /
  15 1 15 /
  15 1 14 /
  15 1 13 /
  15 1 12 /
  15 1 11 /
  15 1 10 /
  15 1  9 /
  15 1  8 /
  15 1  7 /
  15 1  6 /
  15 1  5 /
  15 1  4 /
  15 1  3 /
  15 1  2 /
  15 1  1 /
/

END       ###############################################################################
```

9.26. CO_2 injection into a 2D-layered brine formation (BLACKOIL)

This problem is discussed in the paper: Pruess, K. et al. 2004. Code Intercomparison Builds Confidence in Numerical Simulation Models for Geologic Disposal of CO_2. Energy, 29(9-10): 1431-1444. DOI:10.1016/j.energy.2004.03.077. See a description of the problem, input data files and visualisations under *the link*.

UTSIRA-BLACKOIL-1.RUN

```
################################################################################

               CO2 INJECTION IN A 2D-LAYERED BRINE FORMATION

    The problem is from the paper:
        Pruess, K. et al. 2004. Code Intercomparison Builds Confidence in
        Numerical Simulation Models for Geologic Disposal of CO2. Energy,
        29(9-10): 1431-1444. DOI:10.1016/j.energy.2004.03.077.

    See description at:
        http://dx.doi.org/10.1016/j.energy.2004.03.077;
        http://esd1.lbl.gov/files/research/projects/tough/
                            documentation/TOUGH2-ECO2N_V2.0_Users_Guide.pdf

    Note:
        According to the formulation of the problem, the permeability at
        the interface between every two grid blocks is upstream weighted.
        MUFITS uses a harmonic weighted permeability which is a conventional
        approximation for an industrial simulator. Therefore, we redefine
        the permeability in the vertical direction by the MULTZ- multiplier
        assuming that the flow is in the upward direction.

--------------------------------------------------------------------------------

    Last update on 15.11.2015
    Program Version: 2015.G01

    Note: In this model, OIL stands for the water component, and GAS stands
          for the CO2 component.
    Note: In this example we neglect precipitation of salt near the injection
          point. The model can predict only distribution of CO2 away of the
          injection point.

RUNSPEC   ################### RUNSPEC section begins here ######################

```

```
METRIC    ^^^^^^^^^^^^ We use METRIC units ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
                                       Note, this keywords affects the units
                                       in many keywords below.

BLACKOIL  ^^^^^^ We switch the program to the GASSTORE EOS module ^^^^^^^^^^^^^^

                                       We enable
OIL                                    - oil phase (stands for water)
GAS                                    - gas phase (stands for CO2)
DISGAS                                 - gas dissolution in oil

CAPPRES                                We enable capillary pressure.

FAST                                   We use FAST option.

START                                  The date of beginning of the CO2
  1 OCT 1996 /                         injection is 01 october, 1996.

AUXARRAY                               We enable 2 auxiliary arrays
  #ELEVTN /                            - elevation;
  #XCOORD /                            - x coordinate.
/                                      Both arrays will be defined in the GRID
                                       section.

GRAVITY                                The gravity acceleration is 9.81 m/sec^2
   9.81 /

GRID      ###################### GRID section begins here ######################

          The grid is specified within brackets MAKE-ENDMAKE
MAKE      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
-- cartesian                           We select Cartesian gridding
--    grid   nx  ny  nz                option and specify the number of
      CART   29   1  34  /             grid blocks along every axis.

XYZBOUND
  2*  0 1  858 1000 /                  The top surface of the formation is at
                                       the depth 858 meters. The number 1000
                                       is redefined using the following
                                       keyword DZV. The domain extension
                                       along axis Oy is 1 meter.

DXV
   1   1   2   4   4   8  15  20  30  40
  50  50 100 150  50 150 300  50 475 500
 500 500 500 500 500 500 500 500 1E-3 / Here, we specify the extensions of the
                                       grid blocks along axis Ox.

```

```
DZV
   3   6  12   6   3   3   3   6  12   6
   3   3   3   6  12   6   3   3   3   6
  12   6   3   3   3   6  12   6 2.5   1
 2.5   6   6   7 /                      Here, we specify the thickness of
                                        every layer.


SRCSPECG
  'INJECTOR' 1 1 30 /                   The injection well is modelled by a
/                                       point source (named INJECTOR) which is
                                        located in the grid block i=1, j=1,
                                        k=30.

ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

EQUALS                                  By default, the SATNUM number is 1 in
  SATNUM 2  4*  2*6  /                  all grid blocks. Here, we equal SATNUM
  SATNUM 2  4*  2*12 /                  to 2 in the shale layers which are
  SATNUM 2  4*  2*18 /                  k=6; k=12; k=18 and k=24.
  SATNUM 2  4*  2*24 /
/

EQUALREG
-- -------- -------- -------- --------  We equal porosity to
   PORO     0.35     SATNUM   1 /       -  0.35 in the sand layers;
   PORO     0.1025   SATNUM   2 /       -  0.1025 in the shales.
-- -------- -------- -------- --------
/

EQUALREG
-- -------- -------- -------- --------  We equal permeability to
   PERMX    3000     SATNUM   1 /       - 3000 mD in the sand layers (X-dir)
   PERMZ    3000     SATNUM   1 /       - 3000 mD in the sand layers (Z-dir)
   PERMX    10       SATNUM   2 /       - 10 mD in the shales (X-dir)
   PERMZ    3000     SATNUM   2 /       - 3000 mD in the shales (Z-dir)
   MULTZ-  0.0033333 SATNUM   2 /       - we multiply the above value by
-- -------- -------- -------- --------    0.003333.
/

MULTIPLY                                We multiply permeability by 0.9869 thus
  PERMX 0.9869 /                        converting units from mD to m^2.
  PERMZ 0.9869 /
/

EQUALS
  ACTNUM 2  2*29 /                      We fix parameters in the boundary
/                                       grid blocks i=29.
```

```
132
133 COPY
134   ZCOORD #ELEVTN /
135   XCOORD #XCOORD /                        Here, we define the values of two
136 /                                         auxiliary arrays which are used for
137 ADD                                       data output.
138   #ELEVTN 1042 /
139 /
140
141 RPTGRID                                  We report these properties from the
142   PORO PERMX PERMZ ACTNUM /               GRID section.
143
144 PROPS     ######################## PROPS section begins here ####################
145
146 EOS       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
147 /
148
149 -- the black-oil PVT data were generated by the following keyword:
150 -- GSTOBO
151 --   UTSIRA.PVT 310.15 4.54104E-4 /
152 --   1 2 5 10 30 50 70 80 90 100 110 120 130 140 160 200 /
153
154 DENSITY
155    998.610 1* 1.840 /                    Oil density at stock tank conditions
156                                          is 998.313 kg/m3, and gas density is
157                                          1.84 kg/m3.
158
159                                          Using the PVTO keyword we specify the
160                                          properties of live oil.
161
162 PVTO
163 --  gas-oil ratio   pressure          FVF         viscosity
164     5.6311246E-01  1.0000000E+00  1.0057622E+00  6.9224330E-01
165                    2.0000000E+00  1.0057251E+00  6.9224635E-01
166                    5.0000000E+00  1.0056138E+00  6.9225566E-01
167                    1.0000000E+01  1.0054283E+00  6.9227166E-01
168                    3.0000000E+01  1.0046861E+00  6.9234189E-01
169                    5.0000000E+01  1.0039439E+00  6.9242197E-01
170                    7.0000000E+01  1.0032018E+00  6.9251180E-01
171                    8.0000000E+01  1.0028307E+00  6.9256035E-01
172                    9.0000000E+01  1.0024596E+00  6.9261130E-01
173                    1.0000000E+02  1.0020885E+00  6.9266465E-01
174                    1.1000000E+02  1.0017174E+00  6.9272038E-01
175                    1.2000000E+02  1.0013464E+00  6.9277849E-01
176                    1.3000000E+02  1.0009753E+00  6.9283896E-01
177                    1.4000000E+02  1.0006042E+00  6.9290178E-01
178                    1.6000000E+02  9.9986203E-01  6.9303446E-01
179                    2.0000000E+02  9.9837769E-01  6.9332763E-01 /
```

```
   1.1580827E+00  2.0000000E+00  1.0065992E+00  6.9224635E-01
                  5.0000000E+00  1.0064879E+00  6.9225566E-01
                  1.0000000E+01  1.0063023E+00  6.9227166E-01
                  3.0000000E+01  1.0055601E+00  6.9234189E-01
                  5.0000000E+01  1.0048180E+00  6.9242197E-01
                  7.0000000E+01  1.0040758E+00  6.9251180E-01
                  8.0000000E+01  1.0037047E+00  6.9256035E-01
                  9.0000000E+01  1.0033336E+00  6.9261130E-01
                  1.0000000E+02  1.0029626E+00  6.9266465E-01
                  1.1000000E+02  1.0025915E+00  6.9272038E-01
                  1.2000000E+02  1.0022204E+00  6.9277849E-01
                  1.3000000E+02  1.0018493E+00  6.9283896E-01
                  1.4000000E+02  1.0014782E+00  6.9290178E-01
                  1.6000000E+02  1.0007361E+00  6.9303446E-01
                  2.0000000E+02  9.9925173E-01  6.9332763E-01 /
   2.9071678E+00  5.0000000E+00  1.0090573E+00  6.9225566E-01
                  1.0000000E+01  1.0088718E+00  6.9227166E-01
                  3.0000000E+01  1.0081296E+00  6.9234189E-01
                  5.0000000E+01  1.0073875E+00  6.9242197E-01
                  7.0000000E+01  1.0066453E+00  6.9251180E-01
                  8.0000000E+01  1.0062742E+00  6.9256035E-01
                  9.0000000E+01  1.0059031E+00  6.9261130E-01
                  1.0000000E+02  1.0055320E+00  6.9266465E-01
                  1.1000000E+02  1.0051610E+00  6.9272038E-01
                  1.2000000E+02  1.0047899E+00  6.9277849E-01
                  1.3000000E+02  1.0044188E+00  6.9283896E-01
                  1.4000000E+02  1.0040477E+00  6.9290178E-01
                  1.6000000E+02  1.0033055E+00  6.9303446E-01
                  2.0000000E+02  1.0018212E+00  6.9332763E-01 /
   5.7044954E+00  1.0000000E+01  1.0129812E+00  6.9227166E-01
                  3.0000000E+01  1.0122390E+00  6.9234189E-01
                  5.0000000E+01  1.0114968E+00  6.9242197E-01
                  7.0000000E+01  1.0107547E+00  6.9251180E-01
                  8.0000000E+01  1.0103836E+00  6.9256035E-01
                  9.0000000E+01  1.0100125E+00  6.9261130E-01
                  1.0000000E+02  1.0096414E+00  6.9266465E-01
                  1.1000000E+02  1.0092703E+00  6.9272038E-01
                  1.2000000E+02  1.0088993E+00  6.9277849E-01
                  1.3000000E+02  1.0085282E+00  6.9283896E-01
                  1.4000000E+02  1.0081571E+00  6.9290178E-01
                  1.6000000E+02  1.0074149E+00  6.9303446E-01
                  2.0000000E+02  1.0059306E+00  6.9332763E-01 /
   1.5449557E+01  3.0000000E+01  1.0265549E+00  6.9234189E-01
                  5.0000000E+01  1.0258127E+00  6.9242197E-01
                  7.0000000E+01  1.0250706E+00  6.9251180E-01
                  8.0000000E+01  1.0246995E+00  6.9256035E-01
                  9.0000000E+01  1.0243284E+00  6.9261130E-01
                  1.0000000E+02  1.0239573E+00  6.9266465E-01
```

```
                  1.1000000E+02  1.0235862E+00  6.9272038E-01
                  1.2000000E+02  1.0232152E+00  6.9277849E-01
                  1.3000000E+02  1.0228441E+00  6.9283896E-01
                  1.4000000E+02  1.0224730E+00  6.9290178E-01
                  1.6000000E+02  1.0217308E+00  6.9303446E-01
                  2.0000000E+02  1.0202465E+00  6.9332763E-01 /
 2.2939139E+01    5.0000000E+01  1.0368153E+00  6.9242197E-01
                  7.0000000E+01  1.0360731E+00  6.9251180E-01
                  8.0000000E+01  1.0357020E+00  6.9256035E-01
                  9.0000000E+01  1.0353309E+00  6.9261130E-01
                  1.0000000E+02  1.0349598E+00  6.9266465E-01
                  1.1000000E+02  1.0345888E+00  6.9272038E-01
                  1.2000000E+02  1.0342177E+00  6.9277849E-01
                  1.3000000E+02  1.0338466E+00  6.9283896E-01
                  1.4000000E+02  1.0334755E+00  6.9290178E-01
                  1.6000000E+02  1.0327333E+00  6.9303446E-01
                  2.0000000E+02  1.0312490E+00  6.9332763E-01 /
 2.8107453E+01    7.0000000E+01  1.0436656E+00  6.9251180E-01
                  8.0000000E+01  1.0432945E+00  6.9256035E-01
                  9.0000000E+01  1.0429234E+00  6.9261130E-01
                  1.0000000E+02  1.0425523E+00  6.9266465E-01
                  1.1000000E+02  1.0421812E+00  6.9272038E-01
                  1.2000000E+02  1.0418101E+00  6.9277849E-01
                  1.3000000E+02  1.0414391E+00  6.9283896E-01
                  1.4000000E+02  1.0410680E+00  6.9290178E-01
                  1.6000000E+02  1.0403258E+00  6.9303446E-01
                  2.0000000E+02  1.0388415E+00  6.9332763E-01 /
 2.9411141E+01    8.0000000E+01  1.0452096E+00  6.9256035E-01
                  9.0000000E+01  1.0448386E+00  6.9261130E-01
                  1.0000000E+02  1.0444675E+00  6.9266465E-01
                  1.1000000E+02  1.0440964E+00  6.9272038E-01
                  1.2000000E+02  1.0437253E+00  6.9277849E-01
                  1.3000000E+02  1.0433542E+00  6.9283896E-01
                  1.4000000E+02  1.0429831E+00  6.9290178E-01
                  1.6000000E+02  1.0422410E+00  6.9303446E-01
                  2.0000000E+02  1.0407566E+00  6.9332763E-01 /
 2.9921496E+01    9.0000000E+01  1.0455883E+00  6.9261130E-01
                  1.0000000E+02  1.0452172E+00  6.9266465E-01
                  1.1000000E+02  1.0448461E+00  6.9272038E-01
                  1.2000000E+02  1.0444750E+00  6.9277849E-01
                  1.3000000E+02  1.0441040E+00  6.9283896E-01
                  1.4000000E+02  1.0437329E+00  6.9290178E-01
                  1.6000000E+02  1.0429907E+00  6.9303446E-01
                  2.0000000E+02  1.0415064E+00  6.9332763E-01 /
 3.0936789E+01    1.0000000E+02  1.0467087E+00  6.9266465E-01
                  1.1000000E+02  1.0463376E+00  6.9272038E-01
                  1.2000000E+02  1.0459665E+00  6.9277849E-01
                  1.3000000E+02  1.0455955E+00  6.9283896E-01
```

```
                    1.4000000E+02  1.0452244E+00  6.9290178E-01
                    1.6000000E+02  1.0444822E+00  6.9303446E-01
                    2.0000000E+02  1.0429979E+00  6.9332763E-01 /
     3.1675015E+01  1.1000000E+02  1.0474221E+00  6.9272038E-01
                    1.2000000E+02  1.0470510E+00  6.9277849E-01
                    1.3000000E+02  1.0466799E+00  6.9283896E-01
                    1.4000000E+02  1.0463089E+00  6.9290178E-01
                    1.6000000E+02  1.0455667E+00  6.9303446E-01
                    2.0000000E+02  1.0440824E+00  6.9332763E-01 /
     3.2264725E+01  1.2000000E+02  1.0479173E+00  6.9277849E-01
                    1.3000000E+02  1.0475463E+00  6.9283896E-01
                    1.4000000E+02  1.0471752E+00  6.9290178E-01
                    1.6000000E+02  1.0464330E+00  6.9303446E-01
                    2.0000000E+02  1.0449487E+00  6.9332763E-01 /
     3.2760444E+01  1.3000000E+02  1.0482745E+00  6.9283896E-01
                    1.4000000E+02  1.0479034E+00  6.9290178E-01
                    1.6000000E+02  1.0471612E+00  6.9303446E-01
                    2.0000000E+02  1.0456769E+00  6.9332763E-01 /
     3.3189560E+01  1.4000000E+02  1.0485338E+00  6.9290178E-01
                    1.6000000E+02  1.0477916E+00  6.9303446E-01
                    2.0000000E+02  1.0463073E+00  6.9332763E-01 /
     3.3906750E+01  1.6000000E+02  1.0488452E+00  6.9303446E-01
                    2.0000000E+02  1.0473609E+00  6.9332763E-01 /
     3.4978504E+01  2.0000000E+02  1.0489353E+00  6.9332763E-01 /
/

PVDG
--     pressure         FVF          viscosity
     1.0000000E+00  1.0730862E+00  1.5565440E-02 /
     2.0000000E+00  5.3412277E-01  1.5580561E-02 /
     5.0000000E+00  2.1075013E-01  1.5628945E-02 /
     1.0000000E+01  1.0289902E-01  1.5720696E-02 /
     3.0000000E+01  3.0714827E-02  1.6270207E-02 /
     5.0000000E+01  1.5850420E-02  1.7327514E-02 /
     7.0000000E+01  8.7440572E-03  1.9946955E-02 /
     8.0000000E+01  5.6206055E-03  2.4789558E-02 /
     9.0000000E+01  3.0067093E-03  4.6278183E-02 /
     1.0000000E+02  2.6881677E-03  5.4658120E-02 /
     1.1000000E+02  2.5458054E-03  5.9804921E-02 /
     1.2000000E+02  2.4548815E-03  6.3731572E-02 /
     1.3000000E+02  2.3883166E-03  6.7003429E-02 /
     1.4000000E+02  2.3359290E-03  6.9859365E-02 /
     1.6000000E+02  2.2561814E-03  7.4773629E-02 /
     2.0000000E+02  2.1484836E-03  8.2781117E-02 /
/

ENDEOS    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

```

Using the PVDG keyword we specify the properties of dry gas.

```
          Rock properties are specified within brackets ROCK-ENDROCK
ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
/
ROCKDH                                The rock density is 2600 kg/m3, and the
   2600  0.92 /                       heat capacity is 0.92 kJ/kg/K. These
                                       numbers do not affect the simulation
                                       since it is running in the Isothermal
                                       mode.
ENDROCK   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The relative permeabilities and capillary pressure are specified
          within brackets SAT-ENDSAT
SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
/         Here, we specify the saturation functions for the region SATNUM=1,
          i.e. for the sand layers.

SGOF
--     sgas            krgas            krliq            pcgl
   0.000000000000   0.000000000000   1.000000000000   0.000000000000 /
   0.000001000000   0.000000000000   0.987479385400   0.000000000000 /
   0.000010000000   0.000000000000   0.968695014700   0.000000000000 /
   0.000100000000   0.000000000000   0.922270476300   0.000000000000 /
   0.001000000000   0.000000000000   0.810412236900   0.000000000000 /
   0.005000000000   0.000000000000   0.655485851400   0.002598035231 /
   0.010000000000   0.000000000000   0.560372644900   0.004254391500 /
   0.020000000000   0.000000000000   0.446701011000   0.006751168135 /
   0.030000000000   0.000000000000   0.372772184500   0.008820526801 /
   0.050000000000   0.000000000000   0.275111549000   0.012425808540 /
   0.051000000000   0.000000004738   0.271296445200   0.012595260470 /
   0.052000000000   0.000000037875   0.267557283600   0.012763989140 /
   0.055000000000   0.000000590617   0.256771407600   0.013266131240 /
   0.060000000000   0.000004709136   0.240105789900   0.014091145200 /
   0.075000000000   0.000072839506   0.198049174000   0.016501192060 /
   0.100000000000   0.000572839507   0.146500089300   0.020408807050 /
   0.150000000000   0.004424691356   0.082996814130   0.028285745690 /
   0.200000000000   0.014400000000   0.047583886410   0.036787448310 /
   0.250000000000   0.032869135810   0.026991883090   0.046453149880 /
   0.300000000000   0.061728395060   0.014912902340   0.057894785670 /
   0.350000000000   0.102400000000   0.007909588050   0.071951695230 /
   0.400000000000   0.155832098800   0.003962729625   0.089897683190 /
   0.450000000000   0.222498765400   0.001837449554   0.113817678700 /
   0.500000000000   0.302400000000   0.000766497053   0.147393851800 /
   0.550000000000   0.395061728400   0.000275630707   0.197750575000 /
   0.600000000000   0.499535802400   0.000079621867   0.280452234600 /
   0.650000000000   0.614400000000   0.000016203809   0.436070193400 /
   0.700000000000   0.737758024800   0.000001732079   0.805851789800 /
   0.750000000000   0.867239506100   0.000000038169   2.285560899000 /
   0.770000000000   0.920125440000   0.000000002298   4.919830283000 /
```

```
   0.780000000000   0.946704086800   0.000000000247    9.039249414000 /
   0.785000000000   0.960015840000   0.000000000051   13.917259450000 /
   0.788000000000   0.968008126500   0.000000000015   19.450202320000 /
   0.790000000000   0.973338042500   0.000000000005   25.568110170000 /
   0.792000000000   0.978669080900   0.000000000002   35.732680940000 /
   0.793000000000   0.981334951900   0.000000000001   43.657037670000 /
   0.794000000000   0.984001019900   0.000000000000   55.014231000000 /
   0.794500000000   0.985334119200   0.000000000000   62.684207110000 /
   0.795000000000   0.986667257200   0.000000000000   72.318162360000 /
   0.795300000000   0.987467157300   0.000000000000   79.351605310000 /
   0.795500000000   0.988000430700   0.000000000000   84.700076910000 /
   0.795700000000   0.988533709200   0.000000000000   90.677587800000 /
   0.795850000000   0.988933671300   0.000000000000   95.638012850000 /
   0.795971600000   0.989257909100   0.000000000000  100.0           /
   0.8              1.0              0.0              100.0           /
/
                                        By using the SGLF keyword we define
                                        the relative permeability of gas
                                        (krgas), the relative permeability of
                                        brine (krliq), the capillary pressure
                                        (pcgl) as the functions of the gas
                                        saturation (sgas).

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
  2 /     Here, we specify the saturation functions for the region SATNUM=1,
          i.e. shales.

SGOF
--      sgas             krgas            krliq            pcgl
   0.000000000000   0.000000000000   1.000000000000   0.000000000000 /
   0.000001000000   0.000000000000   0.987479385400   0.000000000000 /
   0.000010000000   0.000000000000   0.968695014700   0.000000000000 /
   0.000100000000   0.000000000000   0.922270476300   0.000000000000 /
   0.001000000000   0.000000000000   0.810412236900   0.000000000000 /
   0.005000000000   0.000000000000   0.655485851400   0.044993906240 /
   0.010000000000   0.000000000000   0.560372644900   0.073679405870 /
   0.020000000000   0.000000000000   0.446701011000   0.116919671600 /
   0.030000000000   0.000000000000   0.372772184500   0.152757726700 /
   0.050000000000   0.000000000000   0.275111549000   0.215195566900 /
   0.051000000000   0.000000004738   0.271296445200   0.218130209300 /
   0.052000000000   0.000000037875   0.267557283600   0.221052326000 /
   0.055000000000   0.000000590617   0.256771407600   0.229748641600 /
   0.060000000000   0.000004709136   0.240105789900   0.244036592800 /
   0.075000000000   0.000072839506   0.198049174000   0.285774834600 /
   0.100000000000   0.000572839507   0.146500089300   0.353448613700 /
   0.150000000000   0.004424691356   0.082996814130   0.489864869500 /
```

```
   0.200000000000   0.014400000000   0.047583886410   0.637101060100 /
   0.250000000000   0.032869135810   0.026991883090   0.804495891800 /
   0.300000000000   0.061728395060   0.014912902340   1.002647126000 /
   0.350000000000   0.102400000000   0.007909588050   1.246090811000 /
   0.400000000000   0.155832098800   0.003962729625   1.556887251000 /
   0.450000000000   0.222498765400   0.001837449554   1.971144157000 /
   0.500000000000   0.302400000000   0.000766497053   2.552630952000 /
   0.550000000000   0.395061728400   0.000275630707   3.424730628000 /
   0.600000000000   0.499535802400   0.000079621867   4.856994007000 /
   0.650000000000   0.614400000000   0.000016203809   7.552053629000 /
   0.700000000000   0.737758024800   0.000001732079  13.956092450000 /
   0.730000000000   0.814883524000   0.000000243083  23.876580870000 /
   0.740000000000   0.840983040000   0.000000104078  30.101074860000 /
   0.750000000000   0.867239506100   0.000000038169  39.582339580000 /
   0.755000000000   0.880419040000   0.000000021379  46.365704030000 /
   0.758000000000   0.888341397500   0.000000014627  51.424821340000 /
   0.760000000000   0.893628649900   0.000000011184  55.331915820000 /
   0.763000000000   0.901567542900   0.000000007284  62.199801390000 /
   0.765000000000   0.906865183100   0.000000005365  67.609094800000 /
   0.768000000000   0.914818697200   0.000000003277  77.339833350000 /
   0.769000000000   0.917471646000   0.000000002752  81.113294900000 /
   0.770000000000   0.920125440000   0.000000002298  85.203764680000 /
   0.772000000000   0.925435459500   0.000000001572  94.496320770000 /
   0.773035000000   0.928184612100   0.000000001278  100.0           /
   0.8              1.0              0.0             100.0           /
/
                                        By using the SGLF keyword we define
                                        the relative permeability of gas
                                        (krgas), the relative permeability of
                                        brine (krliq), the capillary pressure
                                        (pcgl) as the functions of the gas
                                        saturation (sgas).

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

RPTSATTA
  1 /
RPTSATTA
  2 /

INIT      ###################### INIT section begins here ####################

          The initial equilibrium is specified within brackets EQL-ENDEQL
EQL       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
/

EQUIL
-- -------- -------- -------- -------- --------
```

```
-- datum    datum                        GOC
--   depth  pressure                     depth
-- -------- -------- -------- -------- --------
   1020     110        1*       1*       500 /
-- -------- -------- -------- -------- --------

                   Using the keyword EQUILGS we specify that
                   - pressure is 110 bar at the injection point at depth
                     1020 meters;
                   - gas-water contact is above the top layer
                     (at depth 500 m).

RSVD
   800  0 /                             No gas in oil phase at the initial
   1000 0 /                             moment of time
/


ENDEQL    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

EQUALNAM
  PRES   110  INJECTOR /
  SGAS   1.0  /                         We inject dry gas.
/

RPTSUM
  PRES SOIL SGAS RS
  FLUX#1 FLUX#2 #ELEVTN #XCOORD PCOG /  We report these properties for the grid
                                        blocks.

RPTSRC                                  We report these properties for the
  SMIR#2 SMIT#2 /                       INJECTOR.

RPTFIELD                                We report these properties for the
  FGIPM FGIPGM FGIPLM /                 field (formation).

SCHEDULE   ##################### SCHEDULE section begins here ###################

REPORTS
  NOTHING /                             No reports in LOG file.

VARS
  PRES DMAX 30 /                        Maximum pressure change is 30 bars.
/

ILUTFILL                                Parameters of the linear solver.
  3 /
ILUTDROP
```

```
  1E-3 /

SRCINJE                                  We specify that the injection rate is
  'INJECTOR' MASS  1*  200 1* 13.6944 / 13.6944 ton/day=0.1585 kg/sec. The
/                                        maximum pressure in the "well" grid
                                         block (1,1,30) is 200 bars (this value
                                         is not reached during the simulation).

TUNING                                   The maximal time step is 0.5 days.
  1* 0.5 /

TIME
  0.001 0.0025 0.005 0.0075 0.01 0.015
  0.025 0.05 0.075 0.1 0.15 0.25 0.5 0.75 /  We report parameters at these
                                             times since beginning of the
                                             injection.

DATES
  2 OCT 1996 /
  3 OCT 1996 /
  4 OCT 1996 /
  6 OCT 1996 /
  8 OCT 1996 /
 15 OCT 1996 /
 23 OCT 1996 /
  1 NOV 1996 /
  8 NOV 1996 /
 15 NOV 1996 /
 23 NOV 1996 /
  1 DEC 1996 /
  8 DEC 1996 /                           We report parameters at these
 15 DEC 1996 /                           dates. The total simulation
 23 DEC 1996 /                           time is 2 years.
  1 JAN 1997 /
 15 JAN 1997 /
  1 FEB 1997 /
 15 FEB 1997 /
  1 MAR 1997 /
 15 MAR 1997 /
  1 APR 1997 /
  1 MAY 1997 /
  1 JUN 1997 /
  1 JUL 1997 /
  1 AUG 1997 /
  1 SEP 1997 /
  1 OCT 1997 /
  1 NOV 1997 /
  1 DEC 1997 /
```

```
  1 JAN 1998 /
  1 FEB 1998 /
  1 MAR 1998 /
  1 APR 1998 /
  1 MAY 1998 /
  1 JUN 1998 /
  1 JUL 1998 /
  1 AUG 1998 /
  1 SEP 1998 /
  1 OCT 1998 /
/

POST      ######################## POST section begins here ######################

CONVERT                                  We convert the output to ParaView
                                         compatible format.

RPTPOST
  TIMESEC PRES SOIL SGAS #ELEVTN / We save these parameters variation
POSTBLOC                                 with time in the grid blocks:
  1 1 30 /                               (1,1,30);
  2 1 30 /                               (2,1,30);      (i,j,k)
  1 1 29 /                               (1,1,29);
  1 1 31 /                               (1,1,31);
  1 1 19 /                               (1,1,19).
/

RPTPOST
  NOTHING TIMESEC FGIPM FGIPGM FGIPLM
  DAY MONTH YEAR /                       We save consolidated report for the
POSTFLD                                  field data.
/

RPTPOST
  NOTHING TIMESEC SMIR#2 SMIT#2 /        We save consolidated report for the
POSTSRC                                  INJECTOR.
  INJECTOR /
/

RPTPOST
  NOTHING  TIMESEC FLUX#1 FLUX#2 /
POSTBCON                                 We save reports for the fluxes
   1 1 30 2*    2 1 30 /                   from (1,1,30) into (2,1,30);
   1 1 30 2*    1 1 29 /                   from (1,1,30) into (1,1,29);
   1 1 30 2*    1 1 31 /                   from (1,1,30) into (1,1,31);
   1 1 29 2*    1 1 30 /                   from (1,1,29) into (1,1,30).
/

```

```
612 POSTSPEC
613   1* UTSIRA-BLACKOIL-1.SAT-X500.CSV 54 /
614 RPTPOST
615   NOTHING #ELEVTN SGAS #XCOORD /        We save distribution of gas at 500 m
616 POSTBLK                                 distance from the axis of symmetry.
617   15 1 34 /
618   15 1 33 /
619   15 1 32 /
620   15 1 31 /
621   15 1 30 /
622   15 1 29 /
623   15 1 28 /
624   15 1 27 /
625   15 1 26 /
626   15 1 25 /
627   15 1 24 /
628   15 1 23 /
629   15 1 22 /
630   15 1 21 /
631   15 1 20 /
632   15 1 19 /
633   15 1 18 /
634   15 1 17 /
635   15 1 16 /
636   15 1 15 /
637   15 1 14 /
638   15 1 13 /
639   15 1 12 /
640   15 1 11 /
641   15 1 10 /
642   15 1  9 /
643   15 1  8 /
644   15 1  7 /
645   15 1  6 /
646   15 1  5 /
647   15 1  4 /
648   15 1  3 /
649   15 1  2 /
650   15 1  1 /
651 /
652
653 END       ###############################################################################
```

9.27. Test for longitudinal mechanical dispersion of solutes (GASSTORE)

This problem is discussed in the paper: Afanasyev, A.A. 2018. Numerical modelling of solute flow dispersion in porous media using simulator MUFITS // IOP Conf. Series: Journal of Physics: Conf. Series. 1129: 012002 DOI: 10.1088/1742-6596/1129/1/012002. See a description of the problem, input data files and visualisations under *the link*.

MECHDISPL.RUN

```
################################################################################

               TEST FOR LONGITUDIAL MECHANICAL DISPERSION OF SOLUTES

    The problem is from the paper
    Afanasyev, A.A. 2018. Numerical modelling of solute flow dispersion in
       porous media using simulator MUFITS // IOP Conf. Series: Journal of
       Physics: Conf. Series. 1129: 012002 DOI:10.1088/1742-6596/1129/1/012002.

    See description at: http://doi.org/10.1088/1742-6596/1129/1/012002.

--------------------------------------------------------------------------------

    Last update on 17.07.2019
    Program Version: 2019.B01

################################################################################

RUNSPEC   ################### RUNSPEC section begins here ######################

GASSTORE  ^^^^^^ We switch the program to the GASSTORE EOS module ^^^^^^^^^^^^^^

                                      The fluid consists of
WATER                                 - water
SALT                                  - salt (solute)

ISOTHERM                              We switch the program to the isothermal
                                      modelling option.

METRIC   ^^^^^^^^^^^^^^^^^^^^^^ We use METRIC units ^^^^^^^^^^^^^^^^^^^^^^^^^^^

MECHDISP                              We enable the mechanical dispersion
                                      modelling option.

FAST                                  We use FAST option.
```

```
36
37 AUXARRAY                                  We create auxiliary array #XCOORD which
38   #XCOORD /                               will be used for reporting x coordinate
39 /                                         from the SCHEDULE section.
40
41 GRID      ###################### GRID section begins here ########################
42
43           The grid is specified within brackets MAKE-ENDMAKE
44 MAKE      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
45 -- cartesian                             We select Cartesian gridding
46 --    grid   nx     ny nz                option and specify the number of
47       CART   12000  1  1 /               along every axis.
48
49 XYZBOUND
50 -- xmin-xmax   ymin-ymax   zmin-zmax     We specify the domain extent in metres.
51    -20 100  0  1   0  1 /                The domain length is 120 metres.
52
53 BOUNDARY
54 -- ---- ----------- ------------ -------- --------
55    102   6*         'I+' 5*      INFTHIN  4* 1  2 /
56 -- ---- ----------- ------------ -------- --------
57 /                                         We create infinetely thin gost grid
58                                           block at the right boundary. The
59                                           parameters in this block are fixed
60                                           (ACTNUM=2), will be used as open
61                                           boundary. Further we can refer to this
62                                           block with FLUXNUM=102.
63
64 SRCSPECG
65 -- name   i j k
66    MYSRC  1 1 1 /                         We create a point source with the name
67 /                                         MYSRC and place it in the grid block
68                                           i=1, j=1 and k=1.
69
70 ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
71
72 EQUALS                                   We equate
73    PORO     0.1  /                       - porosity to 0.1;
74    PERMX    100  /                       - X permeability to 100 mD;
75    PERMY    100  /                       - Y permeability to 100 mD.
76 /
77
78 RPTGRID                                  We reporrt these properties from the
79    DEPTH PORO PERMX PERMZ /               GRID section.
80
81 PROPS     ###################### PROPS section begins here ####################
82
83           Rock properties are specified within brackets ROCK-ENDROCK
```

```
ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /
ROCKDH                                  We specify that rock density and heat
      2896.5  1.0   /                   capacity are 2896.5 kg/m3 and
                                        1 kJ/kg/K. These data are irrelevant
                                        with the ISOTHERM simulation but must
                                        be specified according to input data
                                        rules.

MDISPL                                  This keyword specifies the longitudial
-- phase   -- sat --                    dispersivity for the LIQ phase. The
  LIQ      0       1    /               dispersivity is equal to the true fluid
-- vel -------------                    velocity and it does not depend on the
   0       0       0    /               phase saturation.
   1       1       1    /
/

ENDROCK  >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The relative permeabilities are specified within brackets SAT-ENDSAT
SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

SSFN
-- ssol  krf                            We define the reduction factor for
   0.0   1.0 /                          the rock permeability (krf). These data
   1.0   0.0 /                          does not influence the simulation
/                                       results because the halite does not
                                        precipitate.

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

INIT      ####################### INIT section begins here #####################

REGALL    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
          The following OPERAREG keyword will be applied to all grid blocks.

OPERAREG
-- -------- -------- -------- -------- -------- -------- -------- --------
   PRES     1*       ROCKNUM  1        EQUALS   50                        /
   TEMPC    1*       ROCKNUM  1        EQUALS   50                        /
   XSM      1*       FLUXNUM  1        EQUALS   0.0000000                 /
   XSM      XCOORD   FLUXNUM  1        SETINT   0.0001     -3000 0        /
   XSM      1*       FLUXNUM  102      EQUALS   0.0000000                 /
-- -------- -------- -------- -------- -------- -------- -------- --------
/
        With the above keyword we specfy that at t=0:
        - pressure = 50 bar in the whole domain
```

```
        - temperature = 50 C in the whole domain
        - salt concentration is equal to 0 in all internal grid blocks
        - salt concentration is equal to 0.0001 at x<0
        - salt concentration is equal to 0 at the right boundary.

EQUALNAM                    The fluid injected throuh the point source is
   PRES   50   MYSRC /      as follows: T=50 C at P=50bar, its salinity
   TEMPC  50         /      is 0.0001.
   XSM    0.0001     /
/

COPY                        We copy x coordinate to the user created array
  XCOORD #XCOORD /          #XCOORD.
/

RPTSUM
   PRES TEMPC XSM  PHST DENT SSOL #XCOORD /  We report these parameters at
                                             every report time.

SCHEDULE   ###################### SCHEDULE section begins here ###################

REPORTS
   NOTHING /                              No reports to LOG-file.

                                          Linear solver setup:
ILUTFILL                                  - Ilut-fill parameter is 5;
  5 /
ILUTDROP                                  - Ilut-drop parameter is 1e-4.
  1E-4 /

TUNING                                    The maximum time step is 7.5 days.
   1*  7.5 /

SRCINJE                                   Through the point source we inject
   MYSRC  RATEIN   1*  100   1E-4 /        fluid at a rate of 1e-4 m3/day.
/

TSTEP                                     We advance simulation to 50000 days
  100*500  /                              reporting results every 500 days.

POST      ######################### POST section begins here ####################

CONVERT                                   We convert the output to ParaView
                                          compatible format.

MNEMPARA                                  The change the output format of the XSM
  XSM  1PE12.5 /                          mnemonic.
/
```

```
180
181 RPTPOST                                The report list consits of two columns,
182   #XCOORD XSM /                        x xoordinate and salinity.
183
184 POSTSPEC                                With these kewords we report the
185    1* TIME25.CSV  25 /                 salinity distribution at the 25th report
186 POSTBLK                                time, i.e. at t=12500 days.
187    1750 1 1 /
188    2000 1 1 /
189    2250 1 1 /
190    2500 1 1 /
191    2750 1 1 /
192    3000 1 1 /
193    3250 1 1 /
194    3500 1 1 /
195    3750 1 1 /
196    4000 1 1 /
197    4250 1 1 /
198    4500 1 1 /
199    4750 1 1 /
200    5000 1 1 /
201    5250 1 1 /
202    5500 1 1 /
203    5750 1 1 /
204    6000 1 1 /
205    6250 1 1 /
206 /
207
208 POSTSPEC                                With these kewords we report the
209    1* TIME50.CSV  50 /                 salinity distribution at the 50th report
210 POSTBLK                                time, i.e. at t=25000 days.
211    2750 1 1 /
212    3000 1 1 /
213    3250 1 1 /
214    3500 1 1 /
215    3750 1 1 /
216    4000 1 1 /
217    4250 1 1 /
218    4500 1 1 /
219    4750 1 1 /
220    5000 1 1 /
221    5250 1 1 /
222    5500 1 1 /
223    5750 1 1 /
224    6000 1 1 /
225    6250 1 1 /
226    6500 1 1 /
227    6750 1 1 /
```

```
   7000 1 1 /
   7250 1 1 /
/

POSTSPEC                                  With these kewords we report the
   1* TIME75.CSV  75 /                    salinity distribution at the 75th report
POSTBLK                                   time, i.e. at t=37500 days.
   3750 1 1 /
   4000 1 1 /
   4250 1 1 /
   4500 1 1 /
   4750 1 1 /
   5000 1 1 /
   5250 1 1 /
   5500 1 1 /
   5750 1 1 /
   6000 1 1 /
   6250 1 1 /
   6500 1 1 /
   6750 1 1 /
   7000 1 1 /
   7250 1 1 /
   7500 1 1 /
   7750 1 1 /
   8000 1 1 /
   8250 1 1 /
/

POSTSPEC                                  With these kewords we report the
   1* TIME100.CSV  100 /                  salinity distribution at the 100th
POSTBLK                                   report time, i.e. at t=50000 days.
   4750 1 1 /
   5000 1 1 /
   5250 1 1 /
   5500 1 1 /
   5750 1 1 /
   6000 1 1 /
   6250 1 1 /
   6500 1 1 /
   6750 1 1 /
   7000 1 1 /
   7250 1 1 /
   7500 1 1 /
   7750 1 1 /
   8000 1 1 /
   8250 1 1 /
   8500 1 1 /
   8750 1 1 /
```

```
276    9000 1 1 /
277    9250 1 1 /
278 /
279
280 END       ###############################################################################
```

9.28. Test for transverse mechanical dispersion of solutes (GASSTORE)

This problem is discussed in the paper: Afanasyev, A.A. 2018. Numerical modelling of solute flow dispersion in porous media using simulator MUFITS // IOP Conf. Series: Journal of Physics: Conf. Series. 1129: 012002 DOI: 10.1088/1742-6596/1129/1/012002. See a description of the problem, input data files and visualisations under *the link*.

MECHDISPT.RUN

```
################################################################################

               TEST FOR TRANSVERSAL MECHANICAL DISPERSION OF SOLUTES

    The problem is from the paper
    Afanasyev, A.A. 2018. Numerical modelling of solute flow dispersion in
       porous media using simulator MUFITS // IOP Conf. Series: Journal of
       Physics: Conf. Series. 1129: 012002 DOI:10.1088/1742-6596/1129/1/012002.

    See description at: http://doi.org/10.1088/1742-6596/1129/1/012002.

--------------------------------------------------------------------------------

    Last update on 17.07.2019
    Program Version: 2019.B01

################################################################################

RUNSPEC   ################### RUNSPEC section begins here ######################

GASSTORE  ^^^^^^ We switch the program to the GASSTORE EOS module ^^^^^^^^^^^^^^

                                      The fluid consists of
WATER                                 - water
SALT                                  - salt (solute)

ISOTHERM                              We switch the program to the isothermal
                                      modelling option.

METRIC   ^^^^^^^^^^^^^^^^^^^^^^ We use METRIC units ^^^^^^^^^^^^^^^^^^^^^^^^^^^

MECHDISP                              We enable the mechanical dispersion
                                      modelling option.

FAST                                  We use FAST option.
```

```
36
37 AUXARRAY                                  We create auxiliary array #YCOORD which
38   #YCOORD /                               will be used for reporting y coordinate
39 /                                         from the SCHEDULE section.
40
41 GRID      ###################### GRID section begins here ########################
42
43           The grid is specified within brackets MAKE-ENDMAKE
44 MAKE      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
45 -- cartesian                             We select Cartesian gridding
46 --    grid   nx   ny  nz                 option and specify the number of
47      CART   200  120   1 /               along every axis.
48 XYZBOUND
49 -- xmin-xmax   ymin-ymax   zmin-zmax     We specify the domain extent in metres.
50     0   100     -15  15     0    1 /
51
52 BOUNDARY
53 -- ---- ----------- ------------ -------- ------
54    101   6*          'I-' 5*      INFTHIN  4* 1  2 /
55    102   6*          'I+' 5*      INFTHIN  4* 1  2 /
56 -- ---- ----------- ------------ -------- ------
57 /                                         We create infinetely thin gost grid
58                                           blocks at the left and right boundaries.
59                                           The parameters in these blocks are fixed
60                                           (ACTNUM=2). Further we can refer to
61                                           these blocks with FLUXNUM=101 and
62                                           FLUXNUM=102 respectively.
63
64 ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
65
66 EQUALS                                   We equate
67    PORO     0.2  /                       - porosity to 0.2;
68    PERMX    100  /                       - X permeability to 100 mD;
69    PERMY    100  /                       - Y permeability to 100 mD.
70 /
71
72 RPTGRID                                  We reporrt these properties from the
73    DEPTH PORO PERMX PERMZ /               GRID section.
74
75 PROPS     ###################### PROPS section begins here ####################
76
77           Rock properties are specified within brackets ROCK-ENDROCK
78 ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
79    /
80 ROCKDH                                   We specify that rock density and heat
81       2896.5  1.0   /                    capacity are 2896.5 kg/m3 and
82                                           1 kJ/kg/K. These data are irrelevant
83                                           with the ISOTHERM simulation but must
```

```
                                        be specified according to input data
                                        rules.

MDISPT                                  This keyword specifies the transversal
-- phase   -- sat --                    dispersivity for the LIQ phase. The
   LIQ     0.0   1.0   /                dispersivity is 10 times less than the
-- vel ---------------                  true fluid velocity and it does not
   0.0     0.0   0.0   /                depend on the phase saturation.
   1.0     0.1   0.1   /
/

ENDROCK  >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

         The relative permeabilities are specified within brackets SAT-ENDSAT
SAT      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

SSFN
-- ssol  krf                            We define the reduction factor for
   0.0   1.0 /                          the rock permeability (krf). These data
   1.0   0.0 /                          does not influence the simulation
/                                       results because the halite does not
                                        precipitate.

ENDSAT   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

INIT     ##########################################################################

FUNCNAME                                With these keywords we specify a user
  2 /                                   defined funciton with name 2. It is
FUNCSPEC                                a linear function which passes through
  00000  55  /                          the points (0,55) and (12000,55)
  12000  50  /
/

REGALL   ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
         The following OPERAREG keyword will be applied to all grid blocks.

OPERAREG
-- -------- -------- -------- -------- -------- -------- -------- --------
   PRESSURE XCOORD   ROCKNUM  1        USERFUNC 2                        /
   TEMPC    1*       ROCKNUM  1        EQUALS   50                       /
   XSM      1*       FLUXNUM  1        EQUALS   0.0001                   /
   XSM      YCOORD   FLUXNUM  1        SETINT   0.0000  0      100       /
   XSM      1*       FLUXNUM  101      EQUALS   0.0001                   /
   XSM      YCOORD   FLUXNUM  101      SETINT   0.0000  0      100       /
   XSM      1*       FLUXNUM  102      EQUALS   0.0001                   /
-- -------- -------- -------- -------- -------- -------- -------- --------
```

```
/

        With the above keyword we specfy that at t=0:
        - the pressure has the linear distribution of the x coordinate
        - salinity is equal to 0.0001 at y<0
        - salinity is equal to 0 at y>=0

COPY                        We copy y coordinate to the user created array
  YCOORD #YCOORD /          #YCOORD.
/

RPTSUM
   PRES TEMPC XSM  PHST DENT SSOL #YCOORD /  We report these parameters at
                                             every report time.

SCHEDULE   ###################### SCHEDULE section begins here ###################

REPORTS
   NOTHING /                                No reports to LOG-file.

                                            Linear solver setup:
ILUTFILL                                    - Ilut-fill parameter is 5;
  5 /
ILUTDROP                                    - Ilut-drop parameter is 1e-4.
  1E-4 /

TUNING                                      The maximum time step is 500 days.
   1*  500 /

TSTEP                                       We run the simulation until
  100*1000  /                               t=100000 days reporting results every
                                            1000 days.

POST      ######################### POST section begins here ###################

CONVERT                                     We convert the output to ParaView
                                            compatible format.

MNEMPARA                                    The change the output format of the XSM
  XSM  1PE12.5 /                            mnemonic.
/

RPTPOST                                     The report list consits of two columns,
  #YCOORD XSM /                             y xoordinate and salinity.

POSTSPEC                                    We report salinity distribution in the
   1* TIME100.CSV  100 /                    i=180 column of grid blocks at the
POSTBLK                                     final simulation time.
```

```
   180   1 1 /
   180   5 1 /
   180  10 1 /
   180  15 1 /
   180  20 1 /
   180  25 1 /
   180  30 1 /
   180  35 1 /
   180  40 1 /
   180  45 1 /
   180  50 1 /
   180  55 1 /
   180  60 1 /
   180  65 1 /
   180  70 1 /
   180  75 1 /
   180  80 1 /
   180  85 1 /
   180  90 1 /
   180  95 1 /
   180 100 1 /
   180 105 1 /
   180 110 1 /
   180 115 1 /
   180 120 1 /
/

END       ###########################################################################
```

9.29. Test for longitudinal mechanical dispersion of temperature (GASSTORE)

See a description of the problem, input data files and visualisations under *the link*.

THRMDISPL.RUN

```
################################################################################

            TEST FOR LONGITUDIAL MECHANICAL DISPERSION OF TEMPERATURE

    The problem is not described in any paper. See a similar problem for
    solute transport in the following paper:
    Afanasyev, A.A. 2018. Numerical modelling of solute flow dispersion in
       porous media using simulator MUFITS // IOP Conf. Series: Journal of
       Physics: Conf. Series. 1129: 012002 DOI:10.1088/1742-6596/1129/1/012002.

--------------------------------------------------------------------------------

    Last update on 17.07.2019
    Program Version: 2019.B01

################################################################################

RUNSPEC   ################### RUNSPEC section begins here ######################

GASSTORE  ^^^^^^ We switch the program to the GASSTORE EOS module ^^^^^^^^^^^^^^

                                    The fluid consists of
WATER                               - water

METRIC   ^^^^^^^^^^^^^^^^^^^^^ We use METRIC units ^^^^^^^^^^^^^^^^^^^^^^^^^^^^

MECHDISP                            We enable the mechanical dispersion
                                    modelling option.

FAST                                We use FAST option.

AUXARRAY                            We create auxiliary array #XCOORD which
  #XCOORD /                         will be used for reporting x coordinate
/                                   fro

GRID      ##################### GRID section begins here #######################

          The grid is specified within brackets MAKE-ENDMAKE
MAKE      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
-- cartesian                        We select Cartesian gridding
```

```
--    grid   nx     ny nz              option and specify the number of
      CART   12000  1  1 /             along every axis.

XYZBOUND
-- xmin-xmax   ymin-ymax   zmin-zmax    We specify the domain extent in metres.
   -20 100  0  1   0  1 /              The domain length is 120 metres.

BOUNDARY
-- ---- ----------- ------------ -------- --------
   102   6*         'I+' 5*      INFTHIN  4* 1  2 /
-- ---- ----------- ------------ -------- --------
/                                       We create infinetely thin gost grid
                                        block at the right boundary. The
                                        parameters in this block are fixed
                                        (ACTNUM=2), will be used as open
                                        boundary. Further we can refer to this
                                        block with FLUXNUM=102.

SRCSPECG
-- name   i j k
   MYSRC  1 1 1 /                       We create a point source with the name
/                                       MYSRC and place it in the grid block
                                        i=1, j=1 and k=1.

ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

EQUALS                                  We equate
   PORO     0.1  /                     - porosity to 0.1;
   PERMX    100  /                     - X permeability to 100 mD;
   PERMY    100  /                     - Y permeability to 100 mD.
/

RPTGRID                                 We reporrt these properties from the
   DEPTH PORO PERMX PERMZ /             GRID section.

PROPS     ######################## PROPS section begins here ####################

          Rock properties are specified within brackets ROCK-ENDROCK
ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /
ROCKDH                                  We specify that rock density and heat
      2896.5  1.0   /                   capacity are 2896.5 kg/m3 and
                                        1 kJ/kg/K.

MDISPL                                  This keyword specifies the longitudial
-- phase   -- sat --                    dispersivity for the LIQ phase. The
  LIQ      0       1    /               dispersivity is equal to the true fluid
-- vel -------------                    velocity and it does not depend on the
```

```
   0       0       0    /               phase saturation.
   1       1       1    /
/

ENDROCK  >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The relative permeabilities are specified within brackets SAT-ENDSAT
SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

INIT      ####################### INIT section begins here #####################

REGALL    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
          The following OPERAREG keyword will be applied to all grid blocks.

OPERAREG
-- -------- -------- -------- -------- -------- -------- -------- --------
   PRES     1*       ROCKNUM  1        EQUALS   50                         /
   TEMPC    1*       ROCKNUM  1        EQUALS   30                         /
   TEMPC    XCOORD   FLUXNUM  1        SETINT   31     -3000      0        /
   TEMPC    1*       FLUXNUM  101      EQUALS   31                         /
   TEMPC    1*       FLUXNUM  102      EQUALS   30                         /
-- -------- -------- -------- -------- -------- -------- -------- --------
/
        With the above keyword we specfy that at t=0:
        - pressure = 50 bar in the whole domain;
        - temperature is equal to 31 C at x<0;
        - temperature is equal to 30 C at x>=0.

EQUALNAM                   The temperature of the fluid injected throuh
   PRES   50   MYSRC /     the point source is 31 C at 50 bar.
   TEMPC  31  /
/

COPY                       We copy x coordinate to the user created array
  XCOORD #XCOORD /         #XCOORD.
/

RPTSUM
   PRES TEMPC PHST DENT DEN$1 ETM$1 #XCOORD /  We report these parameters at
                                               every report time.

SCHEDULE   #####################################################################

REPORTS
   NOTHING /                            No reports to LOG-file.
```

```
137
138                                     Linear solver setup:
139 ILUTFILL                            - Ilut-fill parameter is 5;
140   5 /
141 ILUTDROP                            - Ilut-drop parameter is 1e-4.
142   1E-4 /
143
144 TUNING                              The maximum time step is 50 days.
145    1*  50 /
146
147 SRCINJE                             Through the point source we inject
148    MYSRC  RATEIN   1*  100   1E-4 /  fluid at a rate of 1e-4 m3/day.
149 /
150
151 TSTEP                               We advance simulation to 365699.4489
152   100*3656.994489  /                days reporting results every
153                                     3656.994489 days.
154
155 POST      ######################## POST section begins here ####################
156
157 CONVERT                             We convert the output to ParaView
158                                     compatible format.
159
160 MNEMPARA                            The change the output format of the
161   TEMPC 1PE12.5 /                   TEMPC mnemonic.
162 /
163
164 RPTPOST                             The report list consits of two columns,
165   #XCOORD TEMPC /                   x coordinate and temperature.
166
167 POSTSPEC                            With these kewords we report the
168    1* TIME25.CSV  25 /              temperature distribution at the 25th
169 POSTBLK                            report time.
170    1750 1 1 /
171    2000 1 1 /
172    2250 1 1 /
173    2500 1 1 /
174    2750 1 1 /
175    3000 1 1 /
176    3250 1 1 /
177    3500 1 1 /
178    3750 1 1 /
179    4000 1 1 /
180    4250 1 1 /
181    4500 1 1 /
182    4750 1 1 /
183    5000 1 1 /
184    5250 1 1 /
```

```
   5500 1 1 /
   5750 1 1 /
   6000 1 1 /
   6250 1 1 /
/

POSTSPEC                                  With these kewords we report the
   1* TIME50.CSV  50 /                    temperature distribution at the 50th
POSTBLK                                   report time.
   2750 1 1 /
   3000 1 1 /
   3250 1 1 /
   3500 1 1 /
   3750 1 1 /
   4000 1 1 /
   4250 1 1 /
   4500 1 1 /
   4750 1 1 /
   5000 1 1 /
   5250 1 1 /
   5500 1 1 /
   5750 1 1 /
   6000 1 1 /
   6250 1 1 /
   6500 1 1 /
   6750 1 1 /
   7000 1 1 /
   7250 1 1 /
/

POSTSPEC                                  With these kewords we report the
   1* TIME75.CSV  75 /                    temperature distribution at the 75th
POSTBLK                                   report time.
   3750 1 1 /
   4000 1 1 /
   4250 1 1 /
   4500 1 1 /
   4750 1 1 /
   5000 1 1 /
   5250 1 1 /
   5500 1 1 /
   5750 1 1 /
   6000 1 1 /
   6250 1 1 /
   6500 1 1 /
   6750 1 1 /
   7000 1 1 /
   7250 1 1 /
```

```
   7500 1 1 /
   7750 1 1 /
   8000 1 1 /
   8250 1 1 /
/

POSTSPEC                                  With these kewords we report the
   1* TIME100.CSV  100 /                  temperature distribution at the 100th
POSTBLK                                   report time.
   4750 1 1 /
   5000 1 1 /
   5250 1 1 /
   5500 1 1 /
   5750 1 1 /
   6000 1 1 /
   6250 1 1 /
   6500 1 1 /
   6750 1 1 /
   7000 1 1 /
   7250 1 1 /
   7500 1 1 /
   7750 1 1 /
   8000 1 1 /
   8250 1 1 /
   8500 1 1 /
   8750 1 1 /
   9000 1 1 /
   9250 1 1 /
/

END       ###############################################################################
```

9.30. Test for transverse mechanical dispersion of temperature (GASSTORE)

See a description of the problem, input data files and visualisations under *the link*.

THRMDISPT.RUN

```
################################################################################

            TEST FOR TRANSVERSAL MECHANICAL DISPERSION OF TEMPERATURE

    The problem is not described in any paper. See a similar problem for
    solute transport in the following paper:
    Afanasyev, A.A. 2018. Numerical modelling of solute flow dispersion in
       porous media using simulator MUFITS // IOP Conf. Series: Journal of
       Physics: Conf. Series. 1129: 012002 DOI:10.1088/1742-6596/1129/1/012002.

--------------------------------------------------------------------------------

    Last update on 17.07.2019
    Program Version: 2019.B01

################################################################################

RUNSPEC   ################### RUNSPEC section begins here ######################

GASSTORE  ^^^^^^ We switch the program to the GASSTORE EOS module ^^^^^^^^^^^^^^

                                    The fluid consists of
WATER                               - water

METRIC   ^^^^^^^^^^^^^^^^^^^^^ We use METRIC units ^^^^^^^^^^^^^^^^^^^^^^^^^^^^

MECHDISP                            We enable the mechanical dispersion
                                    modelling option.

FAST                                We use FAST option.

AUXARRAY                            We create auxiliary array #YCOORD which
  #YCOORD /                         will be used for reporting y coordinate
/                                   from the SCHEDULE section.

GRID      ##################### GRID section begins here #######################

          The grid is specified within brackets MAKE-ENDMAKE
MAKE      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
-- cartesian                        We select Cartesian gridding
```

```
--    grid   nx   ny  nz                option and specify the number of
     CART   200  120   1 /              along every axis.
XYZBOUND
-- xmin-xmax   ymin-ymax   zmin-zmax    We specify the domain extent in metres.
    0   100     -15  15     0    1 /

BOUNDARY
-- ---- ----------- ------------ -------- ------
   101   6*         'I-' 5*      INFTHIN  4* 1  2 /
   102   6*         'I+' 5*      INFTHIN  4* 1  2 /
-- ---- ----------- ------------ -------- ------
/                                       We create infinetely thin gost grid
                                        blocks at the left and right boundaries.
                                        The parameters in these blocks are fixed
                                        (ACTNUM=2). Further we can refer to
                                        these blocks with FLUXNUM=101 and
                                        FLUXNUM=102 respectively.

ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

EQUALS                                  We equate
   PORO     0.2  /                      - porosity to 0.2;
   PERMX    100  /                      - X permeability to 100 mD;
   PERMY    100  /                      - Y permeability to 100 mD.
/

RPTGRID                                 We reporrt these properties from the
   DEPTH PORO PERMX PERMZ /             GRID section.

PROPS     ######################## PROPS section begins here ####################

          Rock properties are specified within brackets ROCK-ENDROCK
ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /
ROCKDH                                  We specify that rock density and heat
      2896.5  1.0   /                   capacity are 2896.5 kg/m3 and
                                        1 kJ/kg/K.

MDISPT                                  This keyword specifies the transversal
-- phase   -- sat --                    dispersivity for the LIQ phase. The
   LIQ     0.0   1.0   /                dispersivity is 10 times less than the
-- vel ---------------                   true fluid velocity and it does not
   0.0     0.0   0.0   /                depend on the phase saturation.
   1.0     0.1   0.1   /
/

ENDROCK  >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

```
          The relative permeabilities are specified within brackets SAT-ENDSAT
SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

INIT      ######################## INIT section begins here #####################

FUNCNAME                                 With these keywords we specify a user
  2 /                                    defined funciton with name 2. It is
FUNCSPEC                                 a linear function which passes through
  00000  55  /                           the points (0,55) and (12000,55)
  12000  50  /
/

REGALL    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
          The following OPERAREG keyword will be applied to all grid blocks.

OPERAREG
-- -------- -------- -------- -------- -------- -------- -------- --------
   PRESSURE XCOORD   ROCKNUM  1        USERFUNC 2                        /
   TEMPC    1*       FLUXNUM  1        EQUALS   50                       /
   TEMPC    YCOORD   FLUXNUM  1        SETINT   50.1     -100     0       /
   TEMPC    1*       FLUXNUM  101      EQUALS   50                       /
   TEMPC    YCOORD   FLUXNUM  101      SETINT   50.1     -100     0       /
   TEMPC    1*       FLUXNUM  102      EQUALS   50                       /
-- -------- -------- -------- -------- -------- -------- -------- --------
/

        With the above keyword we specfy that at t=0:
        - the pressure has the linear distribution of the x coordinate
        - temperature is equal to 50.1 C at y<0
        - temperature is equal to 50 C at y>=0

COPY                       We copy y coordinate to the user created array
  YCOORD #YCOORD /         #YCOORD.
/

RPTSUM
   PRES TEMPC PHST DENT #YCOORD /           We report these parameters at
                                            every report time.
SCHEDULE   ###################### SCHEDULE section begins here ##################

REPORTS
   NOTHING /                             No reports to LOG-file.

                                         Linear solver setup:
ILUTFILL                                 - Ilut-fill parameter is 5;
```

```
  5 /
ILUTDROP                                - Ilut-drop parameter is 1e-4.
  1E-4 /

TUNING                                  The maximum time step is 3500 days.
   1*  3500 /

TSTEP                                   We run the simulation until
  100*7313.988978  /                    t=731398.8978 days reporting results
                                        every 7313.988978 days.

POST      ######################## POST section begins here ####################

CONVERT                                 We convert the output to ParaView
                                        compatible format.

MNEMPARA                                The change the output format of the
  TEMPC  1PE12.5 /                      TEMPC mnemonic.
/

RPTPOST                                 The report list consits of two columns,
  #YCOORD TEMPC /                       y coordinate and temperature.

POSTSPEC                                We report the temperature distribution
   1* TIME100.CSV  100 /                in the i=180 column of grid blocks at
POSTBLK                                 the final simulation time.
   180   1 1 /
   180   5 1 /
   180  10 1 /
   180  15 1 /
   180  20 1 /
   180  25 1 /
   180  30 1 /
   180  35 1 /
   180  40 1 /
   180  45 1 /
   180  50 1 /
   180  55 1 /
   180  60 1 /
   180  65 1 /
   180  70 1 /
   180  75 1 /
   180  80 1 /
   180  85 1 /
   180  90 1 /
   180  95 1 /
   180 100 1 /
   180 105 1 /
```

```
   180 110 1 /
   180 115 1 /
   180 120 1 /
/

END       ################################################################################
```

9.31. Test for modelling the brittle-ductile transition. Problem 2c (BINMIXT)

This problem is discussed in the paper: Afanasyev, A. Numerical Modelling of Brittle-Ductile Transition with the MUFITS Simulator // 2020. Comput. Geosci. 24, 1651–1662. DOI: 10.1007/s10596-020-09973-2.. See a description of the problem, input data files and visualisations under *the link*.

BDT-PROBLEM-2C.RUN

```
################################################################################

              MODELLING THE BRITTLE-DUCTILE TRANSITION. PROBLEM 2C

--------------------------------------------------------------------------------

    Last update on 22.02.2020
    Program Version: 2019.B01

    We simulate the problem using the BINMIXT module with CO2-H2O equation of
    state. The concentration of CO2 is set to 0. Thus it does not influence the
    simulation results. A small CO2 concentration can be specified in order to
    track the fluid.

################################################################################

RUNSPEC   ################### RUNSPEC section begins here ######################
          In this section, the general options and parameters of the simulation
             are specified.
                                       We enable
HCROCK                                 - heat conduction
DUCTILE                                - ductile behavior of rocks
DYNFRACK                               - hydrofracturing

METRIC    ^^^^^^^^^^^^^^^^^^^^^^ We use METRIC units ^^^^^^^^^^^^^^^^^^^^^^^^^^

FAST                                   We use FAST option.

AUXARRAY                               We create auxiliary array with name
  #XCOORD /                            #XCOORD in that the x coordinate will
/                                      be saved.

GRID      ###################### GRID section begins here ######################
          In this section, we must define the geometrical and logical structures
             of the reservoir (computational grid, boundaries, faults, wells,
             point sources, etc.).
```

```
37
38          The grid is specified within brackets MAKE-ENDMAKE
39 MAKE      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
40 -- cartesian                          We select Cartesian gridding
41 --    grid   nx   ny  nz               option and specify the number of
42       CART   75    1  25   /          grid blocks along every axis.
43                                       The grid is 75*1*25 blocks. If better
44                                       resolution is needed then increase the
45                                       number of grid blocks to e.g.
46                                       150*1*50.
47
48 XYZBOUND
49 -- xmin-xmax   ymin-ymax   zmin-zmax   We specify the domain extent in meters.
50    -18000 18000   -0.5 0.5  0  12000 / The domain width is 36000 meters and
51                                       its depth is 12000 meters.
52
53 BOUNDARY                              We create the auxiliary blocks for
54    2  6* 'K-' 5*  INFTHIN  4* 1  2 /   - the top boundary (with FLUXNUM=2);
55    3  6* 'K+' 5*  INFTHIN  4* 1  2 /   - the bottom boundary (with FLUXNUM=3).
56 /                                     Parameters in all these blocks are
57                                       constant over time (ACTNUM=2).
58
59 ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
60
61 REGALL    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
62          The following operations are applied to all grid blocks including
63          the auxiliary blocks created with the BOUNDARY keyword.
64
65 OPERAREG
66 -- -------- -------- -------- -------- -------- -------- -------- --------
67    PORO     DEPTH     SATNUM   1       CRUSTPOR  -0.65    -0.1     0.0019  /
68    PERMX    DEPTH     2*               CRUSTPER   1.0      3.2     1000    /
69    HCONDCFX 1*        2*               EQUALS     2.8                      /
70    PRESFAIL DEPTH     2*               MULTA      2        0.11            /
71    PRESLITH DEPTH     2*               MULTA      2        0.27            /
72 -- -------- -------- -------- -------- -------- -------- -------- --------
73 /
74          We specify:
75          1. PORO=10^(-0.65-0.1*DEPTH/1000+0.0019*(DEPTH/1000)^2)
76             (see DOI: 10.1134/S1069351314040181)
77          2. lg(PERMX)=min(1.0-3.2*lg(DEPTH/1000),1000)
78             (see DOI: 10.1029/97JB00552)
79          3. We equate heat conductivity (HCONDCFX) to 2.8 W/m/K
80          4. PRESFAIL=2+0.11*DEPTH (failure pressure)
81          5. PRESLITH=2+0.27*DEPTH (lithostatic pressure)
82
83 COPYREG
84 -- -------- -------- -------- --------    We copy in all grid blocks
```

```
  PERMX     PERMZ     SATNUM   1       /  - PERMX to PERMZ
  HCONDCFX  HCONDCFZ                   /  - HCONDCFX to HCONDCFZ
-- -------- -------- -------- --------
/

OPERAREG
-- -------- -------- -------- -------- -------- -------- -------- --------
   PERMXFR  PERMX    TYPENUM   1       COPY                              /
   PERMXFR  1*       2*                MULTIPLY  5                       /
   PERMZFR  PERMXFR  2*                COPY                              /
-- -------- -------- -------- -------- -------- -------- -------- --------
/
         We specify in all permeable grid blocks (TYPENUM=1):
         1. PERMXFR=5*PERMX
         2. PERMZFR=PERMXFR
         i.e. maximum fractures permeability is 5 times the matrix permeability.

RPTGRID
  DEPTH PORO PERMX PERMZ HCONDCFX HCONDCFZ PRESFAIL PRESLITH PERMXFR PERMZFR  /
                                       We report these parameters from the
                                       GRID section. The parameters are saved
                                       in BDT-PROBLEM-2C.GRID.SUM and
                                       BDT-PROBLEM-2C.GRID.vtu.

PROPS     ###################### PROPS section begins here ######################
          In this section, we must define the thermophysical properties of
             the fluid, host rock as well as the relative permeability
             curves.

          Rock properties are specified within brackets ROCK-ENDROCK
ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /
ROCKDH                                 We specify that
      2700  1.0   /                    - rock density is 2700 kg/m3
                                       - rock heat capacity is 1.0 kJ/kg/K.

DUCTTAB
-- ----------  ----------  ----------  ----------
--   T units     Temp_b                 Temp_d
-- ----------  ----------  ----------  ----------
       C          360         430         500       / <<< temperature T
-- ----------  ----------  ----------  ----------               (degree C)
                  0.0         0.5         1.0       / <<< eta(T) function
-- ----------  ----------  ----------  ----------
--               -log(D)     -log(D)     -log(D)
--   lambda     at 360 C    at 430 C    at 500 C
-- ----------  ----------  ----------  ----------
      1.0         0.0000      0.0000      0.0000     /
```

```
      0.9        0.0000      0.8622      1.7245     /
      0.8        0.0000      1.5918      3.1837     /
      0.7        0.0000      2.1888      4.3776     /
      0.6        0.0000      2.6531      5.3061     /
      0.5        0.0000      2.9847      5.9694     / <<< log(D) as function of
      0.4        0.0000      3.1837      6.3673     / <<< lambda and T.
      0.3        0.0000      3.2500      6.5000     /
      0.2        0.0000      3.2500      6.5000     /
      0.1        0.0000      3.2500      6.5000     /
      0.0        0.0000      3.2500      6.5000     /
-- ----------  ----------  ----------  ----------
/                  With the DUCTTAB keyword we specify parameters of the rocks
                   creep, i.e. functions eta(T) and D(lambda,T). The transition
                   from brittle to ductile  behavior begins at TempB=360 C.
                   eta(T)=0 if T<360 C and eta(T)=1 if T>500 C.

DYNFRTAB
--   pore-fluid           fractures
-- factor (lambda)     productivity (F)
      0.5                -0.00025          /
      0.6                -0.00016          /
      0.7                -0.00009          /
      0.8                -0.00004          /
      0.9                -0.00001          /
      0.99               -0.0000001        /
      1.0                 0.00000          /
      1.01                0.0000001        /
      1.1                 0.00001          /
      1.2                 0.00004          /
      1.3                 0.00009          /
      1.4                 0.00016          /
      1.5                 0.00025          /
/                  With the DYNFRTAB keyword we specify parameters of the
                   hydrofracturing, i.e. the function F(lambda).

ENDROCK  >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

LOADEOS
  '../../INCLUDE/CO2H2O_V3.0.EOS' /     We load the CO2-H2O mixture properties
                                        file.

          The relative permeabilities are specified within brackets SAT-ENDSAT
SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

SATTAB
--  swat    krwat     krvap
     0.0     0.0       1.0  /          The relative permeability for water and
```

```
     1.0     1.0       0.0  /          vapor are specified here. They do not
/                                       influence the simulation results because
                                        phase partitioning does not occur.

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

INIT      ######################## INIT section begins here ####################
          In this section, we must define the initial and boundary conditions.
             We must define the required number of parameters for every cell
             associated with the simulation.

REGALL    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
          The following operations are applied to all grid blocks including
          the auxiliary blocks created with the BOUNDARY keyword.

OPERAREG      -- This is applied to all grid blocks.
-- -------- -------- -------- -------- -------- -------- -------- --------
   PRESSURE DEPTH    SATNUM   1        MULTA    1        0.27             /
   TEMPC    DEPTH    SATNUM   1        MULTA    20       0.055            /
   COMP1T   1*       2*                EQUALS   0                         /
-- -------- -------- -------- -------- -------- -------- -------- --------
/

          We specify:
          1. PRESSURE=1+0.27*DEPTH
             (Initial pressure is 1 bar less than the lithostatic pressure.
              The pressure of 1 bar is maintained at the top boundary.)
          2. TEMPC=20+0.055*TEMPC
             (We impose linear distribution of temperature vs depth.
              Initial temperature gradient is 5.5 C per 100 m and
              temperature of 20C is maintained at the top boundary.)
          3. COMP1T=0
             (CO2 concentration is 0.)

OPERAREG
-- -------- -------- -------- -------- -------- -------- -------- --------
    PRES    1*       FLUXNUM   3        ADD      -75                      /
    PRES    XCOORD   FLUXNUM   3        EXP2     5000     0        100    /
    COMP1T  1*       2*                EQUALS   1E-6                      /
-- -------- -------- -------- -------- -------- -------- -------- --------
/                                       We specify inhomogeneous distribution
                                        of pressure at the bottom bondary.
                                        To track the fluid that enters the
                                        domain from below we equals its CO2
                                        concentration to 1e-6. This small value
                                        does not influence the fluid properties.

COPY
```

```
-- -------- --------
   XCOORD   #XCOORD   /                We copy XCOORD to #XCOORD for futher
-- -------- --------                  usage in the POST section. We do this
/                                     because XCOORD is deleted by the
                                      simulator after the SCHEDULE section.
RPTSUM
   DEPTH #XCOORD DENT PRES TEMPC
   DCTTMULT PFLDFACT FLUXI#T FLUXK#T
   PHST COMP1T PRESLITH PRESFAIL
   PRESFDYN TRANMULT TRANFRMT  /      We specify the properties which are
                                      saved at every report time (in the files
                                      BDT-PROBLEM-2C.####.SUM (.vtu)).

SCHEDULE   ##################### SCHEDULE section begins here ####################

REPORTS
  NOTHING /                           No reports to the LOG-file.

<<<<<<<<<<<<<<<<<<<<<<<< These are numerical algorithm setups <<<<<<<<<<<<<<<<<<<

VARS
-- -------- -------- --------         We specify that in every simulation cell
    PRES     MAXV    3500     /       - maximum pressure is 3500 bar;
    PRES     DMAX    500      /       - maximum pressure change is 500 bar;
    TEMP     MAXV    1373.15  /       - maximum temperature is 1373.15 K;
    TEMP     DMAX    100      /       - maximum temperature change is 100 K.
-- -------- -------- --------         (The change is in one iteration of the
/                                     Newton method).

ILUTFILL                              Here, we change parameters of the linear
  7 /                                 solver:
ILUTDROP                              - ilut-fill=7
  1E-5 /                              - ilut-drop=1e-5

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

TUNING                                We specify that the maximum time step
  1* 10000  2*0.0001 /                is 10000 days.
TSTEP                                 We advance simulation to t=10000 years=
 3652500 /                            =3652500 days.

TUNING                                We increase the maximum time step to
  1* 25000 /                          25000 days.
TSTEP                                 We advance simulation to t=500000 years
  49*3652500 /                        reporting simulation data every 10000
                                      years.

POST      ###################### POST section begins here ####################
```

```
277
278 CONVERT                                  We convert the output to ParaView
279                                          compatible format.
280
281 RPTPOST
282    NOTHING #XCOORD FLUXK#T /             We report these properties in all
283 POSTSPEC                                 grid blocks K=25 at the final
284    1*  BDT-PROBLEM-2C.FLUX.CSV /         moment of time. The data are written
285 POSTBLK2                                 to the file BDT-PROBLEM-2C.RESULT.CSV.
286    1 75  2*1   2*25 /
287 /
288
289 END       ###############################################################################
```

9.32. Modelling flue gas injection into a saline aquifer

This problem is discussed in the paper: Afanasyev, A., Vedeneeva, E. Compositional modeling of multicomponent gas injection into saline aquifers with the MUFITS simulator // 2021 J. Nat. Gas Sci. Eng., 94. 103988. DOI: 10.1016/j.jngse.2021.103988. See a description of the problem, input data files and visualisations under *the link*.

EXAMPLE-H3-FLUE-GAS.RUN

```
################################################################################

                                  FLUE GAS INJECTION

    This is a modified problem 3.1 from
    Class H. et al. 2009 A benchmark study on problems related to CO2 storage
       in geological formations. Comput. Geosci. 13(4):409-434.
        DOI:10.1007/s10596-009-9146-x
    Instead of modelling the injection of pure CO2, we simulate an impure gas
    injection. The injection gas consists of 90% of CO2 and 10% of N2. It
    should be regarded as a flue gas.

    Note, that oil is treated as the brine phase.

    See details in Afanasyev A., Vedeneeva E. Compositional modeling
    of multicomponent gas injection into saline aquifers with the
    MUFITS simulator // 2021 J. Nat. Gas Sci. Eng. 94. 103988.
    DOI: 10.1016/j.jngse.2021.103988.

--------------------------------------------------------------------------------

    Last update on 21.02.2021
    Program Version: 2021.B01

################################################################################

RUNSPEC   ################### RUNSPEC section begins here ######################

FAST                                  We use the FAST option.

COMPS     ^^^^^^ We switch the program to the COMPS EOS module ^^^^^^^^^^^^^^^^^
   3 /                                The fluid is characterised by 3
                                      components.

CAPPRES                               We enable capillary pressure.

ISOTHERM                              We use the isothermal modelling option.
```

```
38
39 METRIC    ^^^^^^^^^^^^ We use METRIC units ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
40
41 GRID      ###################### GRID section begins here ######################
42
43          The grid is specified within brackets MAKE-ENDMAKE
44 MAKE      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
45 -- corner point                          We select Corner-point gridding
46 --    grid   nx  ny  nz                  option and specify the number of
47      CORNER  78  78  9   /               grid blocks along every axis.
48
49 INCLUDE                                  We load the grid in corner-point format
50    'EXAMPLE-H3.GRDECL' /                 from include file.
51
52 INCLUDE                                  We load ACTNUM distribution to cut-off
53    'EXAMPLE-H3-LARGE.ACTNUM' /           a smaller sector of the reservoir.
54
55 FAULTS
56 -- --name--  ------box------  -direction tag-    We specify fault plane
57    FAULT1    40 78 39 39 1 9   J+             /  named FAULT1.
58 -- --------  ---------------  ---------------
59 /
60
61 MULTFLT
62 -- --name--  -multiplier-                 We multiply transmissibility across
63    FAULT1     0.0         /               the fault plane by factor 0.0.
64 -- --------  ------------
65 /
66
67 BOUNDARY
68 -- --name-- ------box------ --direction tags-- --type-- --mode--    -actnum-
69     123     6*                I- I+ J- J+ 2*   SAMESIZE ACTBASED  4*   2    /
70     223     40 78 39 39 1 9   J+ 5*            SAMESIZE ALL       4*   2    /
71     223     40 78 40 40 1 9   J- 5*            SAMESIZE ALL       4*   2    /
72 -- -------- --------------- ------------------ -------- --------    --------
73 /
74               Using the keyword BOUNDARY
75               - we encompass the reservoir with auxiliary grid blocks
76                 marked as FLUXNUM=123. Further, we use these blocks for
77                 modelling fluid influx from the aquifer;
78               - we create auxiliary grid blocks on either side of the fault
79                 plane marked as FLUXNUM=223. Further, we use these grid blocks
80                 to fix pressure at the fault plane.
81
82 WELSPECS
83 -- -------- -- ---- ---- --------
84 -- well        i-   j-   datum
85 --     name     ind  ind   depth
```

```
-- -------- -- ---- ---- --------       We specify well INJECTOR in the column
   INJECTOR 1*  30   48  /              i=30, j=48.
-- -------- -- ---- ---- --------
/

COMPDAT
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
   INJECTOR   1*  1*   6    9   OPEN  1*        1*        0.15    /
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
/                                     We specify that the well is completed
                                      in layer k=6-9. The wellbore diameter
                                      is 0.15 meters.

DZWELL                                We define that the well is visible in
  250 25 /                            ParaView up to 250 meters above the top
                                      layer and up to 25 meters below the
                                      bottom layer.

ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

PORO                                  We load porosity from the include file.
   ASCII 'EXAMPLE-H3.PORO' /

PERMX                                 We load X-axis permeability from
   ASCII 'EXAMPLE-H3.PERM' /          the include file.

COPY
-- --from-- ---to--- ------box------  We copy X-axis permeability into
   PERMX    PERMY    /                Y-axis and Z-axis permeability,
   PERMX    PERMZ    /
-- -------- -------- ---------------
/

RPTGRID
   DEPTH XCOORD YCOORD PORO
   PERMX PERMY PERMZ
   I-IJKRES J-IJKRES K-IJKRES /       We report these properties from the
                                      GRID section.

PROPS     ######################## PROPS section begins here ####################

SATOPT                                We change the method of interpolation
  2* 1 /                              of saturation functions beyond the
                                      the critical values.

          Rock properties are specified within brackets ROCK-ENDROCK
ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /
```

```

ROCKDH                                  We specify that
  2500  1.0 /                           rock density is 2500 kg/m3,
                                        rock heat capacity is 1.0 kJ/kg/K
                                        These quantities are required to be
                                        entered, but do not influence the
                                        injection process, because it is
                                        considered isothermal.

ENDROCK   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          Fluid is characterised within brackets EOS-ENDEOS
EOS       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
/

EOSTATE                                 We chose SW section of the built-in
  SW /                                  library of EoS coefficients. The SW
                                        section contains correlations of the
                                        Soreide and Whitson (1992) model.

LIBCOMPS                                The fluid consists of H2O (i.e. brine),
   H2O   CO2   N2  /                    CO2, and N2. Note, that H2O must be
                                        the 1st component of the fluid.

ZI                                      The default (i.e. initial) fluid
   1.0   0.0   0.0 /                    composition is 100% of H2O, 0%
                                        of CO2, and 0% of N2.

RXSM                                    The mass fraction of NaCl in brine is
  0.1 /                                 0.1 (i.e. 1.9 mol/kg).

ENDEOS    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The relative permeabilities are specified within brackets SAT-ENDSAT
SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

SGOF
--  sgas       krg      krl         Pcgl
  0.000000  0.000000  1.000000   1.00000E-01 /    We specify the relative
  0.050000  0.000000  1.000000   1.03280E-01 /    permeability, krg and krl, and
  0.100000  0.000573  0.758835   1.06904E-01 /    capillary pressure, Pcgl, as
  0.150000  0.004425  0.564168   1.10940E-01 /    functions of gas saturation,
  0.200000  0.014400  0.409600   1.15470E-01 /    sgas.
  0.250000  0.032869  0.289205   1.20605E-01 /
  0.300000  0.061728  0.197531   1.26491E-01 /
  0.350000  0.102400  0.129600   1.33333E-01 /
  0.400000  0.155832  0.080909   1.41421E-01 /
```

```
  0.450000  0.222499  0.047427   1.51186E-01 /
  0.500000  0.302400  0.025600   1.63299E-01 /
  0.550000  0.395062  0.012346   1.78885E-01 /
  0.600000  0.499536  0.005057   2.00000E-01 /
  0.650000  0.614400  0.001600   2.30940E-01 /
  0.700000  0.737758  0.000316   2.82843E-01 /
  0.750000  0.867240  0.000020   4.00000E-01 /
  0.800000  1.000000  0.000000   1.00000E+01 /
/

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

INIT      ####################### INIT section begins here #####################

REGALL    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

          The initial fluid equilirium is specified within EQL-ENDEQL brackets.
EQL       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
/

EQUIL
-- -------- -------- -------- -------- --------
-- datum    datum                      GOC
--   depth  pressure                   depth
-- -------- -------- -------- -------- --------
   2500     253.1     1*       1*       2000 /
-- -------- -------- -------- -------- --------

                    Using the keyword EQUIL, we specify that
                    - pressure is 253.1 bar at depth 2500 m;
                    - gas-water contact is above the top layer
                      (at depth 2000 m).

TEMPCVD
   2500  85  /      We specify that the temperature is 85C at 2500 m depth and
   3000  100 /      100 C at 3000 m depth.
/

ENDEQL    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

EQUALNAM                                   We specify that the injected gas
  PRES    250. INJECTOR /                  temperature is 80 C at 25 MPa.
  TEMPC   80            /
  ZMF$1   0.0           /                  The injection gas composition is 90% of
  ZMF$2   0.90          /                  CO2 and 10% of N2.
  ZMF$3   0.10         /
/

```

```
230
231
232 RPTSUM
233   PRES TEMPC SGAS SOIL ZMF$1 ZMF$2 ZMF$3
234   DGAS DOIL XMF$1 XMF$2 XMF$3
235   YMF$1 YMF$2 YMF$3  VISGAS VISOIL /    We save the ditributions of these
236                                         properties at every report time.
237                                         blocks.
238
239 RPTFIELD
240   FXIPM$1 FXIPM$2 FXIPM$3
241   FYIPM$1 FYIPM$2 FYIPM$3
242   FZIPM$1 FZIPM$2 FZIPM$3 /           We save these fluid-in-place quantities
243                                       at every report time.
244
245 SCHEDULE   ###################### SCHEDULE section begins here ####################
246
247 WEEKTOL                               We enable WEEKTOL modelling option
248                                       (week convergence tolerance).
249
250 ILUTFILL                              ILUT fill factor is 5.0.
251    5 /
252 ILUTDROP                              ILUT drop factor is 0.001.
253   1E-4 /
254
255 REPORTS                              No reports to the LOG-file.
256   NOTHING  /
257
258           ================================================================================
259
260 WELLINJE
261 -- -------- ---- ---- -------- -------- -------- -------- -------- --------
262 --   name   mode ctrl volume   mass     bhp      max pump tanktype vol rate
263 --                        rate     rate            rate              dimens
264 -- -------- ---- ---- -------- -------- -------- -------- -------- --------
265    INJECTOR OPEN MASS     0     1119.5      500     1E5 /
266 -- -------- ---- ---- -------- -------- -------- -------- -------- --------
267 /
268                                       Using the keyword WELLINJE, we start
269                                       CO2 injection. Injection rate is
270                                       1119.5 t/day. Maximal bottom hole
271                                       pressure is 500 bar.
272
273 TUNING                                The next time step is 1 day. The
274     1. 92.  1. /                      maximal time step is 92 days.
275
276 TSTEP                                 We advance simulation to 25 years
277    25*365.25 /                        reporting every year.
```

```
278
279          ========================================================================
280
281 WELLSTOP                                  Now, we stop the gas injection.
282    INJECTOR /
283 /
284
285 TUNING                                   The next time step is 1 day. The
286     1.  92  1. /                         maximal time step is 92 days.
287
288 TSTEP                                    We advance simulation for 25 years
289    25*365.25 /                           reporting every year.
290
291 POST      ######################## POST section begins here ######################
292
293 CONVERT                                  We convert the output to ParaView
294                                          compatible format.
295
296 RPTPOST
297    NOTHING TIMEY
298    FXIPM$1 FXIPM$2 FXIPM$3
299    FYIPM$1 FYIPM$2 FYIPM$3
300    FZIPM$1 FZIPM$2 FZIPM$3 /             We report these quantities changes
301 POSTFLD                                  with time in a separate table.
302 /
303
304 END        ############################################################################
```

9.33. Modelling cushion gas substitution with CO_2

This problem is discussed in the paper: Afanasyev, A., Vedeneeva, E. Compositional modeling of multicomponent gas injection into saline aquifers with the MUFITS simulator // 2021 J. Nat. Gas Sci. Eng., 94. 103988. DOI: 10.1016/j.jngse.2021.103988. See a description of the problem, input data files and visualisations under *the link*.

GAS-STORAGE.RUN

```
################################################################################

                      MODELLING CUSHION GAS SUBSTITUTION WITH CO2

    This problem concerns modelling CO2 injection in a natural gas storage
    reservoir aiming at both cushion (base) gas substitution and CO2 storage.
    The storage reservoir is a sytetic model of an anticline structure
    saturated with brine. We use methane (CH4) as a proxy for natural gas.

    Note, that oil is treated as the brine phase.

    See details in Afanasyev A., Vedeneeva E. Compositional modeling
    of multicomponent gas injection into saline aquifers with the
    MUFITS simulator // 2021 J. Nat. Gas Sci. Eng. 94. 103988.
    DOI: 10.1016/j.jngse.2021.103988.

--------------------------------------------------------------------------------

    Last update on 23.02.2021
    Program Version: 2021.B01

################################################################################

RUNSPEC   ################### RUNSPEC section begins here ######################

COMPS     ^^^^^^ We switch the program to the COMPS EOS module ^^^^^^^^^^^^^^^^
   3 /                                  The fluid is characterised by 3
                                        components.

CAPPRES                                 We enable capillary pressure.

MECHDISP                                We enable the option for modelling
                                        mechanical dispersion.

ISOTHERM                                We use the isothermal modelling option.

METRIC    ^^^^^^^^^^^^ We use METRIC units ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
```

```
38
39 FAST                                    We use the FAST option.
40
41 GRID      ###################### GRID section begins here ######################
42
43           The grid is specified within brackets MAKE-ENDMAKE
44 MAKE      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
45 -- corner point                        We select Corner-point gridding
46 --    grid   nx  ny  nz                 option and specify the number of
47      CORNER  60  60  8   /             grid blocks along every axis.
48
49 INCLUDE                                 We load the grid in corner-point format
50    'GAS-STORAGE.GRDECL' /               from the include file.
51
52 BOUNDARY
53 -- --name--  ---box---  --direction tags--  --type--  --mode--     -actnum-
54       123       6*       I- J- I+ J+ 2*     SAMESIZE  ACTBASED  4*    2     /
55 -- --------  ---------  ------------------  --------  --------     --------
56 /
57               Using the keyword BOUNDARY,
58               we encompass the reservoir with auxiliary grid blocks
59               marked as FLUXNUM=123. Further, we use these blocks for
60               modelling fluid influx from the aquifer.
61
62 WELSPECS
63 -- -------- -- ---- ---- --------
64 -- well        i-   j-   datum
65 --     name     ind  ind   depth
66 -- -------- -- ---- ---- --------
67    NGWELL1  1*  29   30     0.0    /    We specify 8 wells NGWELLi,
68    NGWELL2  1*  29   31     0.0    /    where i=1,...,8, that are used for
69    NGWELL3  1*  30   32     0.0    /    CH4 injection and withdrawal.
70    NGWELL4  1*  31   32     0.0    /
71    NGWELL5  1*  32   31     0.0    /
72    NGWELL6  1*  32   30     0.0    /
73    NGWELL7  1*  31   29     0.0    /
74    NGWELL8  1*  30   29     0.0    /
75    CO2WELL1 1*  17   25     0.0    /    We specify 8 wells CO2WELLi,
76    CO2WELL2 1*  25   17     0.0    /    where i=1,...,8, that are used for
77    CO2WELL3 1*  17   36     0.0    /    CO2 injection
78    CO2WELL4 1*  36   17     0.0    /
79    CO2WELL5 1*  44   36     0.0    /
80    CO2WELL6 1*  36   44     0.0    /    The datum depth for every well
81    CO2WELL7 1*  25   44     0.0    /    is set to 0 m. This means that
82    CO2WELL8 1*  44   25     0.0    /    the reported bottom-hole pressure
83 -- -------- -- ---- ---- --------       (WBHP) should be regarded as the
84 /                                        well head pressure. The pressure
85                                         distribution within each wellbore
```

```
                                        is assumed hydrostatic.

COMPDAT
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
-- well     i-   j-   k-   k-   mode satnum   explicit   well
--     name  ind  ind  min  max                 tran   diameter
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
   NGWELL1    1*  1*   1    4   OPEN  1*         1*        0.3048  /
   NGWELL2    1*  1*   1    4   OPEN  1*         1*        0.3048  /
   NGWELL3    1*  1*   1    4   OPEN  1*         1*        0.3048  /
   NGWELL4    1*  1*   1    4   OPEN  1*         1*        0.3048  /
   NGWELL5    1*  1*   1    4   OPEN  1*         1*        0.3048  /
   NGWELL6    1*  1*   1    4   OPEN  1*         1*        0.3048  /
   NGWELL7    1*  1*   1    4   OPEN  1*         1*        0.3048  /
   NGWELL8    1*  1*   1    4   OPEN  1*         1*        0.3048  /
   CO2WELL1   1*  1*   5    8   OPEN  1*         1*        0.3048  /
   CO2WELL2   1*  1*   5    8   OPEN  1*         1*        0.3048  /
   CO2WELL3   1*  1*   5    8   OPEN  1*         1*        0.3048  /
   CO2WELL4   1*  1*   5    8   OPEN  1*         1*        0.3048  /
   CO2WELL5   1*  1*   5    8   OPEN  1*         1*        0.3048  /
   CO2WELL6   1*  1*   5    8   OPEN  1*         1*        0.3048  /
   CO2WELL7   1*  1*   5    8   OPEN  1*         1*        0.3048  /
   CO2WELL8   1*  1*   5    8   OPEN  1*         1*        0.3048  /
-- -------- ---- ---- ---- ---- ---- -------- -------- --------
/                                       We specify that the wells NGWELLi
                                        are completed in the top 4 layers
                                        of the reservoir models. The wells
                                        CO2WELLi are completed in the bottom
                                        4 layers of the model. The wellbore
                                        diamter is 0.3048 m.

DZWELL                                  We define that the well is visible in
  100 25 /                              ParaView up to 250 meters above the top
                                        layer and up to 25 meters below the
                                        bottom layer.

ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

EQUALS                                  We specify that
  PORO    0.2   /                       - porosity is 0.2,
  PERMX   1000  /                       - horizontal permeability is 1000 mD,
  PERMY   1000  /                       - vertical permeabiloity is 500 mD.
  PERMZ   500   /
/

RPTGRID
   DEPTH XCOORD YCOORD PORO
   PERMX PERMY PERMZ
```

```
   I-IJKRES J-IJKRES K-IJKRES /         We report these properties from the
                                        GRID section.

PROPS     ###################### PROPS section begins here ####################

          Rock properties are specified within brackets ROCK-ENDROCK
ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

ROCKDH                                  We specify that
  2500  1.0 /                          rock density is 2500 kg/m3,
                                        rock heat capacity is 1.0 kJ/kg/K
                                        These quantities are required to be
                                        entered, but do not influence the
                                        injection process, because it is
                                        considered isothermal.

ROCKECL
   100  1E-5 /                         We specify that rock compressibility
                                        is 1e-5 bar^-1 at 100 bar.
MDISPL
-- phase   -- sat --
   OIL   0.0    1.0   /                Using this keyword, we specify that
-- vel ---------------                  the longitudinal dispersivity for oil
   0.0   0.0    0.0   /                is 30 m^2/day at the intinsic oil
   1.0   30     30    /                velocity of 1 m/day. This means
/                                       that the longitudinal dispersion
                                        length is 30 m. The dispersivity does
                                        not depend on the oil saturation.

MDISPT
-- phase   -- sat --
   OIL   0.0    1.0   /                Using this keyword, we specify that
-- vel ---------------                  the transversal dispersivity for oil
   0.0   0.0    0.0   /                is 15 m^2/day at the intinsic oil
   1.0   15     15    /                velocity of 1 m/day. This means
/                                       that the transversal dispersion
                                        length is 15 m. The dispersivity does
                                        not depend on the oil saturation.

MDISPL
-- phase   -- sat --
   GAS   0.0    1.0   /                Using this keyword, we specify that
-- vel ---------------                  the longitudinal dispersivity for gas
   0.0   0.0    0.0   /                is 30 m^2/day at the intinsic gas
   1.0   30     30    /                velocity of 1 m/day. This means
/                                       that the longitudinal dispersion
                                        length is 30 m. The dispersivity does
```

```
                                        not depend on the gas saturation.

MDISPT
-- phase   -- sat --
   GAS   0.0    1.0   /                 Using this keyword, we specify that
-- vel ---------------                  the transversal dispersivity for gas
   0.0   0.0    0.0   /                 is 15 m^2/day at the intinsic gas
   1.0   15     15    /                 velocity of 1 m/day. This means
/                                       that the transversal dispersion
                                        length is 15 m. The dispersivity does
                                        not depend on the gas saturation.

ENDROCK   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>


          Fluid is characterised within brackets EOS-ENDEOS
EOS       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
/

EOSTATE                                 We chose SW section of the built-in
  SW /                                  library of EoS coefficients. The SW
                                        section contains correlations of the
                                        Soreide and Whitson (1992) model.

LIBCOMPS                                The fluid consists of H2O (i.e. brine),
   H2O   CO2   C1  /                    CO2, and C1. Note, that H2O must be
                                        the 1st component of the fluid.

ZI                                      The default (i.e. initial) fluid
   1.0   0.0   0.0 /                    composition is 100% of H2O, 0%
                                        of CO2, and 0% of N2.

RTEMP                                   The reservoir temperature (i.e. the
  35  /                                 default temperature is 35 degree
                                        Celcius.

RSMOL                                   The NaCl molality of the reservoir
   0.5 /                                brine is 0.5 mol/kg.

ENDEOS    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

SAVEPVO                                 Using this keyword, we can export
  1 'GAS-STORAGE.PVO' /                 the parameters of the fluid model.

          The relative permeabilities are specified within brackets SAT-ENDSAT
SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

```

```
SGOF
--  sgas       krg       krl         Pcgl
  0.000000  0.000000  1.000000   0.00000E+00 /      We specify the relative
  0.025000  0.000000  0.406574   1.11990E-02 /      permeability, krg and krl, and
  0.050000  0.000000  0.275112   1.75764E-02 /      capillary pressure, Pcgl, as
  0.075000  0.000073  0.198049   2.32465E-02 /      functions of gas saturation,
  0.100000  0.000573  0.146500   2.86937E-02 /      sgas.
  0.125000  0.001900  0.109884   3.41306E-02 /
  0.150000  0.004425  0.082997   3.96891E-02 /
  0.175000  0.008488  0.062855   4.54710E-02 /
  0.200000  0.014400  0.047584   5.15677E-02 /
  0.225000  0.022443  0.035924   5.80709E-02 /
  0.250000  0.032869  0.026992   6.50794E-02 /
  0.275000  0.045900  0.020147   7.27049E-02 /
  0.300000  0.061728  0.014913   8.10786E-02 /
  0.325000  0.080517  0.010927   9.03585E-02 /
  0.350000  0.102400  0.007910   1.00739E-01 /
  0.375000  0.127480  0.005645   1.12463E-01 /
  0.400000  0.155832  0.003963   1.25842E-01 /
  0.425000  0.187500  0.002729   1.41278E-01 /
  0.450000  0.222499  0.001837   1.59305E-01 /
  0.475000  0.260814  0.001205   1.80642E-01 /
  0.500000  0.302400  0.000766   2.06280E-01 /
  0.525000  0.347184  0.000470   2.37628E-01 /
  0.550000  0.395062  0.000276   2.76737E-01 /
  0.575000  0.445900  0.000153   3.26711E-01 /
  0.600000  0.499536  0.000080   3.92452E-01 /
  0.625000  0.555777  0.000038   4.82114E-01 /
  0.650000  0.614400  0.000016   6.10198E-01 /
  0.675000  0.675154  0.000006   8.04846E-01 /
  0.700000  0.737758  0.000002   1.12762E+00 /
  0.725000  0.801900  0.000000   1.73905E+00 /
  0.750000  0.867240  0.000000   3.19812E+00 /
  0.775000  0.933406  0.000000   9.05003E+00 /
  0.800000  1.000000  0.000000   1.00000E+02 /
/

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

INIT      ######################## INIT section begins here #####################

REGALL    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

EQUALREG                                Using this keyword, we specofy that
   TEMPC  35  SATNUM 1 /                the temperature in every wellbore is
   PRES   100 /                         35C. This influences the hydraulic
   ZMF$1  1.0 /                         head.
   ZMF$2  0.0 /
```

```
   ZMF$3  0.0 /
/

          The initial fluid equilirium is specified within EQL-ENDEQL brackets.
EQL       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
/

EQUIL
-- -------- -------- -------- -------- --------
-- datum    datum                       GOC
--   depth  pressure                    depth
-- -------- -------- -------- -------- --------
    1000     100      1*       1*       800 /
-- -------- -------- -------- -------- --------

                   Using the keyword EQUIL, we specify that
                   - pressure is 100 bar at 1000 m depth;
                   - gas-water contact is above the top layer
                     (at the depth of 800 m).

                   Note, that the reservoir temperature and the salinity of
                   brine are specified within EOS-ENDEOS brackets.

ENDEQL    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

EQUALNAM                                 We specify that the injected CH4
  PRES    1.  'NGWELL*' /                temperature is 15.6 C at 1 bar.
  TEMPC   15.6          /                The simulator requires entering the
  ZMF$1   0.0           /                temperature, although it does not
  ZMF$2   0.0           /                influence the flow.
  ZMF$3   1.0           /                Methane is injected through these
/                                        wells.

EQUALNAM                                 We specify that the injected CO2
  PRES    1.  'CO2WELL*' /               temperature is 15.6 C at 1 bar.
  TEMPC   15.6          /                The simulator requires entering the
  ZMF$1   0.0           /                temperature, although it does not
  ZMF$2   1.0           /                influence the flow.
  ZMF$3   0.0           /                CO2 is injected through these wells.
/

RPTSUM
  PRES TEMPC SGAS SOIL
  ZMF$1 ZMF$2 ZMF$3 DGAS DOIL
  XMF$1 XMF$2 XMF$3 YMF$1 YMF$2 YMF$3
  VISGAS VISOIL /                      We save the distributions of these
                                       parameters at every report time.

```

```
RPTWELL
   WBHP WBHTC WMIR#1 WMIR#2 WMIR#3 WMPR#1 WMPR#2 WMPR#3 /
                                        We save these quantities for every well.

RPTFIELD
  FMIR#1 FMIR#2 FMIR#3 FMIT#1 FMIT#2 FMIT#3
  FMPR#1 FMPR#2 FMPR#3 FMPT#1 FMPT#2 FMPT#3
  FXIPM$1 FXIPM$2 FXIPM$3 FYIPM$1 FYIPM$2
  FYIPM$3 FZIPM$1 FZIPM$2 FZIPM$3 /    We save these fluid-in-place quantities
                                       at every report time.

SCHEDULE   ##################### SCHEDULE section begins here ####################

ILUTFILL                               ILUT fill factor is 5.0.
   5 /
ILUTDROP                               ILUT drop factor is 0.001.
  1E-4 /

VARS                                   Maximum pressure change at a Newton
  PRES DMAX  25 /                      iteration is 25 bar.
/

TUNING                                The next time step is 1 days. Maximal
    1  100  10 /                      time step is limited by 100 days.

WELLINJE
   'NGWELL*'    OPEN   MASS  1*  250    120   500000 /
   'CO2WELL*'   OPEN   MASS  1*  500    120   500000 /
/
                                      We inject 250 ton/day of CH4 through
                                      every NGWELL* well abd 500 ton/day of
                                      CO2 through each CO2WELL*. The maximum
                                      wellhead pressure for every well is 120
                                      bar.

TSTEP
  0.1 4.9 59*5 /                      We advance the simulation to 300 days.

WELLSHUT                              The well used for CO2 injection
   'CO2WELL*' /                       are stopped at t=300 days.
/

          Further, we simulate 3 years of the natural gas injection and
          withdrawal. We use identical schedules for every such year. This is
          do by the LOOP-ENDLOOP brackets. The keywords encompassed within these
          brackets are repeated 3 times.
LOOP      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   3   /  We repeat 3 times the following shcedule until ENDLOOP
```

```

TUNING                                        The next time step is 10 days. Maximal
    1  100  10 /                              time step is limited by 100 days.

WELLINJE
   'NGWELL*'    OPEN   MASS  1*  250    120 /
/                                       We inject 250 ton/day of CH4 through
                                        every NGWELL*.
TSTEP
  0.1 4.9 27*5 /                        We advance the simulation for further
                                        140 days.

WELLSTOP                                The wells NGWELLi are stopped.
  'NGWELL*' /
/

TUNING                                  The next time step is 1 day.
  1 /

TSTEP
 0.25 6*5 /                             We advance the simulation for further
                                        30.25 days.

WELLPROD
   'NGWELL*'    OPEN   MASS  1*  250    70 /
/                                       The NGWELL* well are producing 250 ton
                                        per day. the minimum wellhead pressure
                                        is 70 bar.

TUNING                                  The next time step is 1 day.
  1 /

TSTEP
 0.1 4.9 24*5 /                         We advance the simulation for further
                                        125 days

WELLSTOP                                The wells NGWELLi are stopped.
  'NGWELL*' /
/

TUNING                                  The next time step is 1 day.
  1 /

TSTEP
 0.1 4.9 13*5 /                         We advance the simulation for further
                                        80 days
ENDLOOP   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

```

```
422 POST      ######################## POST section begins here ######################
423
424
425 CONVERT                                  We convert the output to ParaView
426                                          compatible format.
427
428 RPTPOST
429    NOTHING TIMEY
430   FMIR#1 FMIR#2 FMIR#3 FMIT#1 FMIT#2 FMIT#3
431   FMPR#1 FMPR#2 FMPR#3 FMPT#1 FMPT#2 FMPT#3
432   FXIPM$1 FXIPM$2 FXIPM$3
433   FYIPM$1 FYIPM$2 FYIPM$3
434   FZIPM$1 FZIPM$2 FZIPM$3 /            We report these quantities changes
435 POSTFLD                                  with time in a separate table.
436 /
437
438 RPTPOST
439    NOTHING TIMEY WBHP WBHTC WMIR#1 WMIR#2 WMIR#3 WMPR#1 WMPR#2 WMPR#3 /
440 POSTWELL
441    '*' /                                 We save separate reports for every well.
442 /
443
444 END        ###########################################################################
```

9.34. Modelling the porosity waves. 1-D study

See a description of the problem, input data files and visualisations under *the link*.

POROWAVE-1D.RUN

```
################################################################################

                    1-D MODELLING OF A SOLITARY POROSITY WAVE

    We consider a 1-D steady propagation of the solitary porosity wave
    originating from a local porosity anomaly. We consider a vertical domain.
    At t = 0, the scaled porosity f (i.e. the porosity multiplier is defined
    as a Gaussian profile.

    This is the 1-D study in the appendix to
       Utkin I., Afanasyev A. Decompaction weakening as a mechanism of fluid
       focusing in hydrothermal systems// 2021 JGR. Solid Earth. Submitted.

--------------------------------------------------------------------------------

    Last update on 06.08.2021
    Program Version: 2021.C03

################################################################################

RUNSPEC   ################### RUNSPEC section begins here ######################
          In this section, the general options and parameters of the simulation
             are specified.
                                       We enable
POROWAVE                               - the porosity waves option
ISOTHERM                               - isothermal modelling
SIMPLMOD                               - the SIMPLMOD EoS-module

METRIC    ^^^^^^^^^^^^^^^^^^^^^ We use METRIC units ^^^^^^^^^^^^^^^^^^^^^^^^^^

FAST                                   We use FAST option.

GRID      ##################### GRID section begins here ######################
          In this section, we must define the geometrical and logical structures
             of the reservoir (computational grid, boundaries, faults, wells,
             point sources, etc.).

          The grid is specified within the MAKE-ENDMAKE brackets
MAKE      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
-- cartesian                           We select Cartesian gridding
--    grid   nx   ny  nz                option and specify the number of
      CART   1    1    1000 /          grid blocks along every axis.
```

```
                                        The grid is 1*1*1000 blocks.

XYZBOUND
-- xmin-xmax   ymin-ymax   zmin-zmax    We specify the domain extent in meters.
   0   100     0  100      0    10000 / The domain length along X and Y axes is
                                        100 meters and its depth is 10000 meters.

BOUNDARY
   101  6*  K- 5*  INFTHIN   1* 3* 1* 2 /  We create the boundary grid blocks
   101  6*  K+ 5*  INFTHIN   1* 3* 1* 2 /  (at the upper and lower boundaries
/                                          of the domain) and fix parameters
                                           in such blocks.

ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

POROMIN                                 We specify that the minimum allowed
   1e-6 /                               porosity is 1e-6 (default value is
                                        1e-3).

EQUALS                                  We specify that
   PORO      0.0001  /                  - porosity is 0.0001
   PERMZ     0.05    /                  - vertical permeability is 0.05 mD
/

REGALL    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
          The following operation is applied to all grid blocks including
          the auxiliary blocks created with the BOUNDARY keyword.

OPERAREG
  PRESLITH  DEPTH  SATNUM 1 MULTA  1  0.27 /
/                                       We specify the linear distribution of
                                        the lithostatic pressure:
                                        PRESLITH=1+0.27*DEPTH

RPTGRID                                 We report these properties from the
  DEPTH PORO PERMZ PRESLITH /           grid section.

PROPS     ###################### PROPS section begins here ######################
          In this section we specify the parameters of the rock and fluid.

          Rock properties are specified within the brackets ROCK-ENDROCK
ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /

PWVROCK                                  Using this keyword, we specify that
--  temp [K]   etas [cP]   G [GPa]       - the shear viscosity of the solid
      400       5E18         1*    /       phase is 5E18 cP (centipoise);
/                                        - the elastic shear modulus is
```

```
                                          infinity (the default value).
                                        Both parameters do not depend on
                                        temperature because the table contains
                                        only 1 row.

PWVPEMLT                                Using this keyword, we specify that
--   Pe [bar]     M     Mgr             - the quantity M=1, i.e. we do not
      100        1.0    1e-3  /           account for the decompaction
/                                         weakening.
                                        - the quantity Mgr=1e-3, i.e. the
                                          growth rate of Omega is not
                                          constrained. Changing to Mgr=1e-4
                                          will not influence the simulation
                                          results.

PV2TRMLT                                Using this table we relate the
-- porvmult  tranmult                  permeability multiplier (tranmult)
   0.0       0.000   /                  with the porosity multiplier (porvmult).
   0.1       0.001   /                  We use the Carman-Kozeny relationship:
   0.2       0.008   /
   0.3       0.027   /                  tranmult=porvmult^3
   0.4       0.064   /
   0.5       0.125   /
   0.6       0.216   /
   0.7       0.343   /
   0.8       0.512   /
   0.9       0.729   /
   1.0       1.000   /
   1.1       1.331   /
   1.2       1.728   /
   1.3       2.197   /
   1.4       2.744   /
   1.5       3.375   /
   1.6       4.096   /
   1.7       4.913   /
   1.8       5.832   /
   2.0       8.000   /
   2.5      15.625   /
   3.0      27.000   /
   3.5      42.875   /
   4.0      64.000   /
   4.5      91.125   /
   5.0     125.000   /
/

ROCKDENS                                We specify that the rock density is
   2700 /                               2700 kg/m^3.

```

```
ENDROCK   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          Fluid properties are specified within the brackets EOS-ENDEOS
EOS       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
/

DENTAB                                  We specify that the fluid density
     600 700  /                         is 999 kg/m3 at 1 bar and 1000 kg/m3 at
 1    2*999   /                         100 bar. The fluid density does not
 100  2*1000  /                         depend on temperature.
/

VISTAB                                  We specify that the fluid viscosity
    600 700 /                           is 1e3 cP. It does not depend on
 1     2*1e3 /                          pressure and temperature.
 100   2*1e3 /
/

ENTHTAB                                 We specify that the fluid enthalpy is
       600  700  /                      100 kJ/kg at T=600 K and 1100 kJ/kg at
 1    100   1100 /                      T=700 K. The enthalpy does not depend on
 100  100   1100 /                      pressure.
/

ENDEOS    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

          The relative permeabilities are specified within brackets SAT-ENDSAT
SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   /
          The SIMPLMOD module assumes that the fluid is single-phase. So
          the saturation functions are not required.

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

INIT      ####################### INIT section begins here #####################
          In this section we specify the initial conditions.

REGALL    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
          The following operation is applied to all grid blocks including
          the auxiliary blocks created with the BOUNDARY keyword.

OPERAREG
  PRES  DEPTH  SATNUM 1 MULTA  1  0.27  /
  TEMP  DEPTH  SATNUM 1 MULTA  400 0.0  /
/
                                        We specify that at t=0
                                        - PRES=1+0.27*depth
                                        - TEMP=400 K
```

```
                                        Since we are using isothermal modelling
                                        the temperature will remain constant in
                                        every grid block.

OPERATE
   PORVMULT 1*       EQUALS     1 /
   PORVMULT DEPTH    EXP2       400  8500   500 /
/                                       We impose the initial porosity
                                        perturbation:
                                        PORVMULT=1+500*exp(-(DEPTH-8500)^2/400^2)


RPTSUM
  DEPTH PRES TEMP PORVMULT POROSITY
  GBVMULT TRANMULT PRESLITH PRESEFF /   We report these quantities at every
                                        report time.

                   PORVMULT is the pore volume multiplier (f)
                   POROSITY is the porosity (phi)
                   GBVMULT  is the volume multiplier (Omega)
                   TRANMULT is the permeability multiplier (k/k_0)
                   PRESEFF  is the effective pressure.


SCHEDULE   ###################### SCHEDULE section begins here ####################
           We simulate the flow within this section.

REPORTS                                 No reports to the LOG-file.
  NOTHING /

VARS
-- -------- -------- --------           We specify that in every simulation cell
    PRES     MAXV    3500     /         - maximum pressure is 3500 bar;
    PRES     DMAX    500      /         - maximum pressure change is 500 bar;
-- -------- -------- --------           (The change is in one iteration of the
/                                       Newton method).

TUNING                                 The maximum time step is 25 days.
   1*  25 /

TSTEP                                  We simulate until t=250000 days reporting
   100*2500 /                          every 2500 days.

POST      ######################## POST section begins here ####################
          We save some useful plots from this section.

MNEMPARA
   PORVMULT  F9.4 /                     We change the output format for the
```

```
/                                       pore volume multiplier.

RPTPOST
   DEPTH PORVMULT GBVMULT PRESEFF  /    We specify the report list.

POSTSPEC                                We save a CSV file with the initial
   1* MYFILE-0.CSV 0 /                  distributions (at t=0).
POSTBLK2
   1 1 1 1 1 1000 /
/

POSTSPEC                                We save into MYFILE-100.CSV the
   1* MYFILE-100.CSV 100 /              parameters at the 100s frame, i.e.
POSTBLK2                                at t=250000 days.
   1 1 1 1 1 1000 /
/

CONVERT                                 We convert the output to ParaView
                                        format.

END       ###############################################################################
```

9.35. Modelling a CCS scenario (BLACKOIL)

See a description of the problem, input data files and visualisations under *the link*.

CCS-3D-EXAMPLE.RUN

```
1
2         This is a demonstrational example aiming at demonstrating some
3         options of the software for modeling CO2 disposal in saline
4         aquifers. Particularly the example demonstrates the following options:
5         -- using BLACKOIL for modeling CCS
6         -- complicated injection controls with balancing within a group of wells
7         -- using aquifers
8         -- using relative permeability hysteresis
9
10 RUNSPEC   ################### RUNSPEC section begins here ######################
11
12 BLACKOIL                We use black-oil model and enable
13 OIL                     -- oil that is used as a proxy for brine
14 GAS                     -- the gas phase
15 DISGAS                  -- gas dissolution in brine
16 VAPOIL                  -- water vapor in the gas phase
17
18 ROCKCOMP                We take into account the pore volume compressibility
19 CAPPRES                 We take into account the capillary pressure
20 HYSTER                  We take into account the relative permeability
21                            hysteresis.
22
23 GRDDISP                          We enable the calculation of the ground
24 --    mode   scale                displacement caused by the gas injection.
25       3D      1  /
26
27 ENDSCALE                We use the end-point scaling option.
28 /
29
30 SRCPOTEN               We set the software to calculate the injection
31                         potentials to balance the target injection between the
32                         groups of wells.
33
34 METRIC    ^^^^^^^^^^^^ We use METRIC units ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
35
36 START
37   1 'JAN' 2024 /       The time t=0 corresponds to 01 Jan 2024
38
39 RESTART                We request saving the additional data required for
40 /                          restarting the simulation.
41
42 DVARCTRL                The automated timestep selection will be used
```

```
  ON /

OPTIONS                This switch influences the visualization of wells
   85* 0 /             in Paraview.

GRID      ###################### GRID section begins here ######################

          The grid is specified within brackets MAKE-ENDMAKE
MAKE      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
--    grid
--    type   nx  ny  nz                We use a Cartesian grid
      CART   80  100 10  /

XYZBOUND
    -1000  3000 -500 4500  1100 1120  3*  2* 0.5 /
                                The top of the reservoir is at 1100 m. The
                                first grid block thickness is 0.5 meters.

CARFIN
--          --i-- --j-- --k--  nx ny nz
   LGR-WA3  27 38 44 53  1 10  24 20 10 /     We implement these local grid
CARFIN                                        refinements to better resolve
   LGR-WB2  21 31 66 71  1 10  33 18 20 /     the flow near specific wells.
ENDFIN


WELSPECC
-- name group x-head y-head bhp-depth
   WA1  PAD-A  1000  -1500   1100 /           We introduce 6 wells merged into
   WA2  PAD-A  1000  -1500   1100 /           2 groupd PAD-A and PAD-B. The
   WA3  PAD-A  1000  -1500   1100 /           wells are defined by the
   WB1  PAD-B  1000  -2500   1100 /           coordinates of the well head
   WB2  PAD-B  1000  -2500   1100 /           and the bottom-hole depth.
   WB3  PAD-B  1000  -2500   1100 /
/

INCLUDE
   WELTRACK.WA1.INC /                   For every well we load its
INCLUDE                                 trajectory and completion intervals.
   WELTRACK.WA2.INC /                   The intervals are specified as the
INCLUDE                                 measured depth intervals.
   WELTRACK.WA3.INC /
INCLUDE
   WELTRACK.WB1.INC /
INCLUDE
   WELTRACK.WB2.INC /
INCLUDE
   WELTRACK.WB3.INC /
```

```
91
92  ARITHCND
93    'ACTNUM=0' 'SQRT( ((XCOORDG-1000)/2000)**2 + ((YCOORDG-2000)/2500)**2 )>1' /
94  /                                       We leave active only an elliptically
95                                          shaped region of whole domain.
96                                          Here, XCOORDG and YCOORDG are the
97                                          x and y coordinates in the GRID frame
98                                          of reference.
99
100 AQUNUMER
101 -- name  length area  poro  perm  depth
102     2     200  3.2e5  0.2   150    1100 /     We simulate an aquifer as a 1-D
103     2     400  3.4e5  0.2   150    1100 /     sequence of 6 grid blocks of
104     2     800  4.0e5  0.2   150    1100 /     increasing size. The parameters
105     2    1600  5.0e5  0.2   150    1100 /     in the last grid block in that
106     2    3200  7.0e5  0.2   150    1100 /     sequence are fixed.
107     2     100  8.0e5  0.2    50    1100   2*  2 /
108
109 /
110
111 AQUCONN
112    2  6*  X- X+ Y- Y+ /                 We specify that the aquifer is connected
113 /                                       to any boundary grid block in the
114                                         lateral direstion.
115
116 ENDMAKE   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
117
118 OBSPOINT
119 -- name     x   y     z                 We introduce 3 observation points
120     PAD-A  1000 -1500 0 /                were the simulator will calculate the
121     MID    1000 -2000 0 /                ground displacement. Two points are
122     PAD-B  1000 -2500 0 /                at the well pads, whereas the third
123 /                                        is exactly between them.
124
125 EQUALS                                  We set
126    PORO  0.2 /                          - porosity equal 0.2
127    PERMX 150 /                          - horizontal permeability equal 150 mD
128    PERMY 150 /                          - vertical permeability equal 15 mD
129    PERMZ 15  /
130 /
131
132
133 RPTGRID
134    DEPTH PORO NTG
135    PERMX PERMY PERMZ
136    FIPNUM EQLNUM ACTNUM MPINUM
137    IRES JRES KRES /                  We report these properties from the
138                                          GRID section.
```

```
139
140 PROPS     ######################## PROPS section begins here ####################
141
142           Rock properties are specified within brackets ROCK-ENDROCK
143 ROCK      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
144    /
145
146 ROCKECL
147   100     4.5E-005 /                We specify the pore volume compressibility.
148
149 GRDPARAM
150 --  shear      drained       bulk modulus    thermal exp.
151 -- modulus   bulk modulus    of the solid    coefficient
152 --  (GPa)      (GPa)            (GPa)          (1/K)
153      2.0        5.0              30.0          1E-5      /
154                                        By using this keyword we specify the
155                                        parameters of rock required for the
156                                        ground displacement modeling.
157
158 ENDROCK   >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
159
160           The relative permeabilities are specified within brackets SAT-ENDSAT
161 SAT       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
162    /
163
164 SGOF
165 --   sgas         krg          krog       pcog
166   0.00000000  0.00000000  1.00000000   1.00000E-01 /   Here, we introduce the
167   0.05000000  0.01118034  0.86844019   1.02598E-01 /   relference saturation
168   0.10000000  0.03162278  0.74845708   1.05409E-01 /   functions that will be
169   0.15000000  0.05809475  0.63959056   1.08465E-01 /   later scaled by changing
170   0.20000000  0.08944272  0.54137409   1.11803E-01 /   the end-points.
171   0.25000000  0.12500000  0.45333419   1.15470E-01 /
172   0.30000000  0.16431677  0.37498993   1.19523E-01 /
173   0.35000000  0.20706279  0.30585229   1.24035E-01 /
174   0.40000000  0.25298221  0.24542338   1.29099E-01 /
175   0.45000000  0.30186918  0.19319567   1.34840E-01 /
176   0.50000000  0.35355339  0.14865089   1.41421E-01 /
177   0.55000000  0.40789092  0.11125881   1.49071E-01 /
178   0.60000000  0.46475800  0.08047574   1.58114E-01 /
179   0.65000000  0.52404675  0.05574259   1.69031E-01 /
180   0.70000000  0.58566202  0.03648240   1.82574E-01 /
181   0.75000000  0.64951905  0.02209709   2.00000E-01 /
182   0.80000000  0.71554175  0.01196279   2.23607E-01 /
183   0.85000000  0.78366128  0.00542314   2.58199E-01 /
184   0.90000000  0.85381497  0.00177828   3.16228E-01 /
185   0.95000000  0.92594546  0.00026434   4.47214E-01 /
186   1.00000000  1.00000000  0.00000000   1.00000E+00 /
```

```
/

ENDSAT    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

           The fluid properties are specified within EOS-ENDEOS brackets.
EOS       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   1  /

INCLUDE
  'CO2-42C-XSM=0.03.PVO' /             We load PVT tables from an include file.
                                       The tabels corresponds to the reservoir
                                       temperature of 42C and the brine
                                       salinity of 0.03 kg/kg.

ENDEOS    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

REGALL                                Here, we specify the end points for the
ARITHREG                              drainage curve. The critical saturation
   'KRO=1.0'  ROCKNUM 1 /             of brine is 0.6. The end-point of krg
   'KRG=0.5'     /                    is 0.5. The critical gas saturation is
   'SGCR=0.05'   /                    0.05.
   'SOGCR=0.6'   /
   'SGU=1-SOGCR' /
/

ARITHREG                              Here, we specify the end-points of the
   'IKRO=1.0'  ROCKNUM 1 /            imbibitoin curve. The critical gas
   'IKRG=0.5'     /                   saturation is 0.25.
   'ISGCR=0.25'   /
   'ISOGCR=0.6'   /
   'ISGU=1-ISOGCR' /
/

INIT      ####################### INIT section begins here #####################

EQL       <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
   1 /

EQUIL
-- -------- -------- -------- -------- --------
-- datum    datum                       GOC
--   depth  pressure                    depth
-- -------- -------- -------- -------- --------
   1100     110       1*       1*       900 /
-- -------- -------- -------- -------- --------

                    Using the keyword EQUIL we specify that
                    - pressure is 110 bar at depth 1100 m;
```

```
                    - gas-oil contact is above the top layer
                      (at depth 900 m).

RSVD
  1100 0.0  /                          We specify that at the initial moment
/                                      of time there is no dissolved gas in
                                       oil (i.e. in brine).

RVVD
  1100 0.0  /                          We specify that at the initial moment
/                                      of time there is no water vapor in the
                                       gas phase.

ENDEQL    >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

EQUALNAM                             We specify that the injected CO2
  PRES    1.01325 'W*' /             is at 1 bar. Here, we specify the fluid
  SGAS    1.0   /                    that is injected.
  RV      0.0   /
/

REGALL                               We two FLUXNUM regions: 1 - ordinary
ARITHREG                             domain cells, 2 - aquifer cells. By
   'FIPNUM=FLUXNUM'   ROCKNUM 1 /    copying FLUXNUM to FIPNUM and using
/                                    RPTFIP, we will request the calculation
                                     of the flow rate between the simulated
                                     sector and the numerical aquifer.

RPTSUM
  PRES SGAS DGAS DOIL VISGAS VISOIL RS  /   We report these properties at every
                                            time step.
RPTWELL
   WBHP WTHP WGIRM WGITM WGIR WGIT WGPI WGPIM /  We report these properties for
                                                 every well.

RPTGRUP                              We report these properties for every
  GGIRM GGIR GGPIM GGPI /            group of wells.

RPTOBS                               We report these properties for every
   UX UY UZ  /                       observation point.

RPTFIELD
  FGIPM FGIPLM FGIPGM FGIPMGM FGIPRGM
  FGIR FGIRM FGIT FGITM /            We report these properties for the field
                                     (i.e. entire simulation).

RPTFIP                               We report fluxes between different
  FLUX#1 FLUX#2 /                    FIPNUM regions.
```

```
283
284 SCHEDULE   ##################### SCHEDULE section begins here ####################
285
286 INCLUDE                                  We load the VFP table from an include
287   'VFP-1040.INC' /                      file to relate the bottom-hole pressures
288                                          with the tubing-head pressures. The VFP
289                                          table ID is 1.
290 NUPCOL
291   3 /                                    This is a simulation setup.
292
293 WELLINJE
294    'WA*' OPEN GRUP 1* 4000  270 1E7 2*  100 1 /   We specify that every well
295    'WB*' OPEN GRUP 1* 4000  270 1E7 2*  100 1 /   in each group will be
296 /                                                 operated under group control.
297                               The maximum BHP is 270 bar and the maximum
298                               injection rate is 4000 ton of CO2 per day.
299
300      There are 4 periods in the simulation schedule:
301
302 ---------------------------------- Period 1 ------------------------------------
303
304 GRUPINJE
305    PAD-A GRAT 1e6  2000   1*  THP /   Both groups are balanced by the constant
306    PAD-B GRAT 1e6  2000   1*  THP /   WTHP. The target injection rate for each
307 /                                      group is 1e6 sm3 of CO2 per day.
308 TUNING
309    0.1 2* 1e-5 /                       We limit the next time step.
310 TSTEP
311   20*10 /                              We inject over 200 days.
312
313 ---------------------------------- Period 2 ------------------------------------
314
315 GRUPINJE
316    PAD-A GRAT 5E5     1000   1*  THP /  Now the target rates are reduced. The
317    PAD-B GRAT 6.25E5  1250   1*  THP /  targer for groups PAD-A and PAD-B are
318 /                                       5e5 and 6.25e5 sm3/day.
319 TUNING
320    0.1 /                               We limit the next time step.
321 TSTEP
322   10*10 /                              We inject over 100 days.
323
324 ---------------------------------- Period 3 ------------------------------------
325
326 GRUPINJE
327    PAD-A GRAT 1.5e6  3000   1*  THP /  The target rates are increased to
328    PAD-B GRAT 1.0e6  2000   1*  THP /  1.5e6 and 1e6 sm2/day, respectively.
329 /
330 TUNING
```

```
   0.1 2* 1e-5 /                          We limit the next time step.
TSTEP
  20*10 /                                 We inject over 100 days.

---------------------------------- Period 4 ------------------------------------

WELLINJE                                                 We actually stop all wells
   'W*' OPEN GRAT 0.0 4000  270 1E7 2*  100 1 /          and so, the entire injection.
/
TUNING
   0.1 2* 1e-5 /                          We limit the next time step.
TSTEP
  10*10  10*30 /                          We simulate the rest period of 400 days.

POST      ######################## POST section begins here ####################

CONVERT                                   We convert the output to ParaView
                                          compatible format.

RPTPOST
   TIME  WBHP WTHP WGIRM WGITM WGIR WGIT WGPI WGPIM  /
POSTWELL
   '*' /                                  We save the CSV reports for every well.
/

POSTSPEC
  1* 'REPORT.PAD-A.CSV' 1* 1 /            We save a CSV report for PAD-A.
SAVEREP
   WELL 'WA1'      1* 3* 2* TIME  /
   WELL 'WA*'      1* 3* 2* WTHP  /
   WELL 'WA*'      1* 3* 2* WBHP  /
   WELL 'WA*'      1* 3* 2* WGIRM /
   WELL 'WA*'      1* 3* 2* WGIR /
   GRUP 'PAD-A'    1* 3* 2* GGIRM /
   GRUP 'PAD-A'    1* 3* 2* GGIR /
/

POSTSPEC
  1* 'REPORT.PAD-B.CSV' 1* 1 /            We save a CSV report for PAD-B.
SAVEREP
   WELL 'WB1'      1* 3* 2* TIME  /
   WELL 'WB*'      1* 3* 2* WTHP  /
   WELL 'WB*'      1* 3* 2* WBHP  /
   WELL 'WB*'      1* 3* 2* WGIRM /
   WELL 'WB*'      1* 3* 2* WGIR /
   GRUP 'PAD-B'    1* 3* 2* GGIRM /
   GRUP 'PAD-B'    1* 3* 2* GGIR /
/
```

```
379
380 RPTPOST
381   NOTHING  TIME  FGIPM FGIPLM FGIPGM FGIPMGM FGIPRGM
382   FGIR FGIRM FGIT FGITM /             We save a CSV report for the field.
383 POSTFLD
384 /
385
386 RPTPOST
387  NOTHING TIME  UX UY UZ /              We save a CSV report for every
388 POSTOBS                               observation point.
389     PAD-A  /
390     MID    /
391     PAD-B  /
392 /
393
394 RPTPOST
395     NOTHING TIME  FLUX#1 FLUX#2 /      We save a CSV report for the fluid
396 POSTFPCO                              flow from the simulated sector to the
397     1 2 /                              aquifer.
398 /
399
400 END        ###########################################################################
```

References

1. MUFITS reservoir simulator website. http://mufits.org/.

2. ParaView website. http://www.paraview.org.