

MUFITS

Reservoir Simulation Software
version 2015.B

Training Course

Author: Andrey Afanasyev

Date: 18 May, 2015

Moscow, Russia

About

This MUFITS training course is designed for researchers and students working in numerical modeling of hydrothermal fields. The course program consists of 5 days during which the sample simulation scenarios are worked through. The emphasis is done on EOS modules T2EOS1 and BINMIXT.

Note, the course is designed for MUFITS version 2015.B. Generally, you cannot run the sample simulations, given in this document, using an older version of the simulator, because full backward and forward compatibility is not supported.

Contents

1. Useful Notes	6
1.1. Installation instruction for Mac OS	6
1.2. Recommended folders hierarchy	7
1.3. Recommended ParaView settings	7
1.4. RUN-file sections	8
1.5. Keywords	10
1.6. Mnemonics	10
1.7. Data flowchart	10
2. Day 1. Fundamentals	12
2.1. Introduction	14
2.2. RUN-file	18
2.3. Scenario 1	24
2.4. Fundamentals	27
2.5. Heat conduction (Scenario 2)	39
3. Day 2. Operations on arrays, Regions & Boundary conditions	45
3.1. Operations on arrays (Scenario 3)	47
3.2. Regions	57
3.3. Boundary conditions	62
3.4. Point sources	66
3.5. Scenario 4	70

4. Day 3. Module BINMIXT, Section POST & Wells	78
4.1. BINMIXT EOS module (Scenario 5)	80
4.2. POST section	93
4.3. Wells (Scenario 6)	100
5. Day 4. Radial grids, Fluid-in-place regions, Local grid refinements & Grid decomposition	123
5.1. Radial grids (Scenario 7)	125
5.2. Fluid-in-Place regions	131
5.3. Local grid refinements (Scenario 8)	142
5.4. Scenario 9	151
5.5. Grid decomposition	159
6. Day 5. Corner-point grids, Faults, Aquifers, & Onshore/offshore	165
6.1. Corner-point grids (Scenario 10)	167
6.2. Faults	183
6.3. Modeling aquifers	187
6.4. Onshore/offshore boundary conditions	194
References	202

1 Useful Notes

1.1. Installation instruction for Mac OS

MPICH2 installation

- Make sure that Xcode is installed on your system. If Xcode is not installed then install it;
- Install MacPorts from www.macports.org;
- Install MPICH2 through MacPorts. Open the terminal and execute the command: `sudo port install mpich2`. Follow the instructions in the terminal window;
- Check that the command `mpirun` is available. Type and execute in the terminal: `which mpirun`. The system path to the file, which specifies the command, must be returned;

Note that OpenMPI can also be used instead of MPICH2.

MUFITS installation

- Create the folder `SIMULATIONS` somewhere on your system. Navigate to the folder `SIMULATIONS`;
- create the folder `BIN` in the folder `SIMULATIONS`;
- Download and copy the simulator executable `H64.EXM` (or `H32.EXM` for 32-bit system) to the folder `SIMULATIONS/BIN`;

- If necessary, download and copy the PVT-program executable I64.EXM (or I32.EXM for 32-bit system) to the folder SIMULATIONS/BIN;
- Navigate to the folder SIMULATIONS/BIN in the terminal. Enable the execution permission for H64.EXM and I64.EXM (H32.EXM and I32.EXM). For doing this, execute in the terminal: chmod +x H64.EXM, and chmod +x I64.EXM;
- Create the folder INCLUDE in the folder SIMULATIONS;
- Copy the mixture properties file CO2H2O_V3.0.EOS to the folder SIMULATIONS/INCLUDE.

1.2. Recommended folders hierarchy

See the figure 1.1.

1.3. Recommended ParaView settings

The recommendations below are developed for ParaView version 4.3.1.

1. Edit→Settings→Color Palette; Change background color to black; Change edge color to grey RGB(115,115,115).
2. In the top toolbar click on “Apply changes to parameters automatically” button.
3. Open a dataset (e.g., pvd file) and select a property (e.g., PRES); Click on “Edit color map” in the top toolbar; On the right panel click on “Chose preset”; Select “Rainbow Blended Grey” color scale; Click on “Save current color map settings values as default” button below.

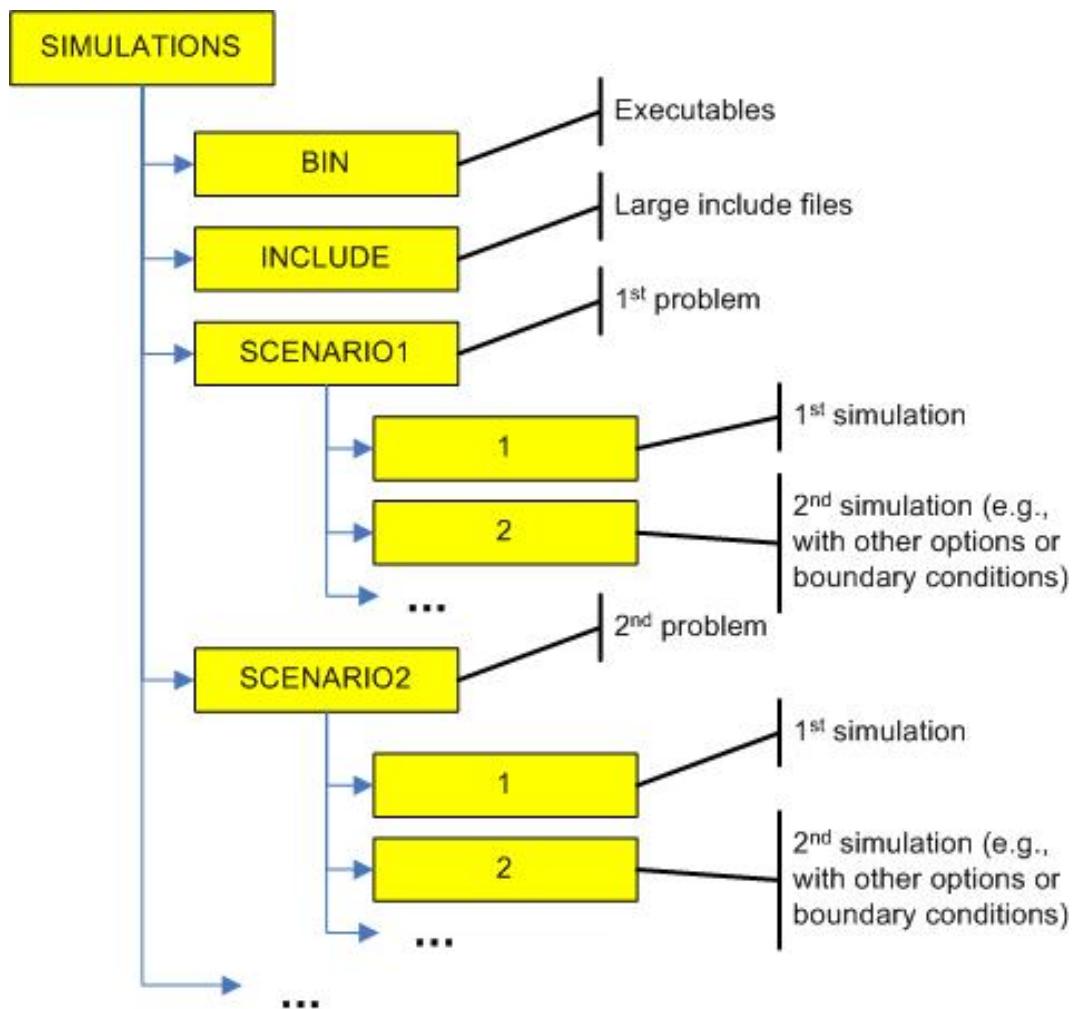


Figure 1.1: Recommended folders hierarchy

1.4. RUN-file sections

The RUN-file must consist of the following sections

RUN-file sections	
RUNSPEC	- General options.
GRID	- Grid & petrophysical data.
EDIT (optional)	- Transmissibilities.
PROPS	- Thermophysical properties of fluids and rocks
INIT	- Initial and boundary conditions
SCHEDULE	- Simulation schedule.

POST (option -al) Output postprocessing.

In the RUNSPEC section, determines the general options and parameters of the simulation.

The GRID section determines

- domain extensions;
- computational grid;
- logical boundaries of the reservoir;
- wells;
- point sources;
- porosity, absolute permeability, and rock heat conduction coefficient distribution;
- inactive/impermeable grid blocks;

The optional EDIT section allows to modify calculated transmissibilities.

The PROPS section determines

- the PVT properties of reservoir fluids;
- the PVT properties of the reservoir rock;
- the relative permeabilities and capillary pressure curves

The initial conditions must be specified for every cell in the INIT section.

The SCHEDULE section determines

- the times at which reports are required;
- well and point sources control modes (production, injection).

The optional POST section allows postprocessing of output files, e.g., the output files can be converted to the format compatible with ParaView.

1.5. Keywords

The keywords are the instructions in the RUN-file to input parameters of the simulation. 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.

See more details on the keywords syntax in the chapter ??.

1.6. Mnemonics

Mnemonic is a short name of a property corresponding to a physical, geological, geometrical, and logical data used in simulation. 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).

See more details on the mnemonics in the chapter ??.

1.7. Data flowchart

See the figure 1.2.

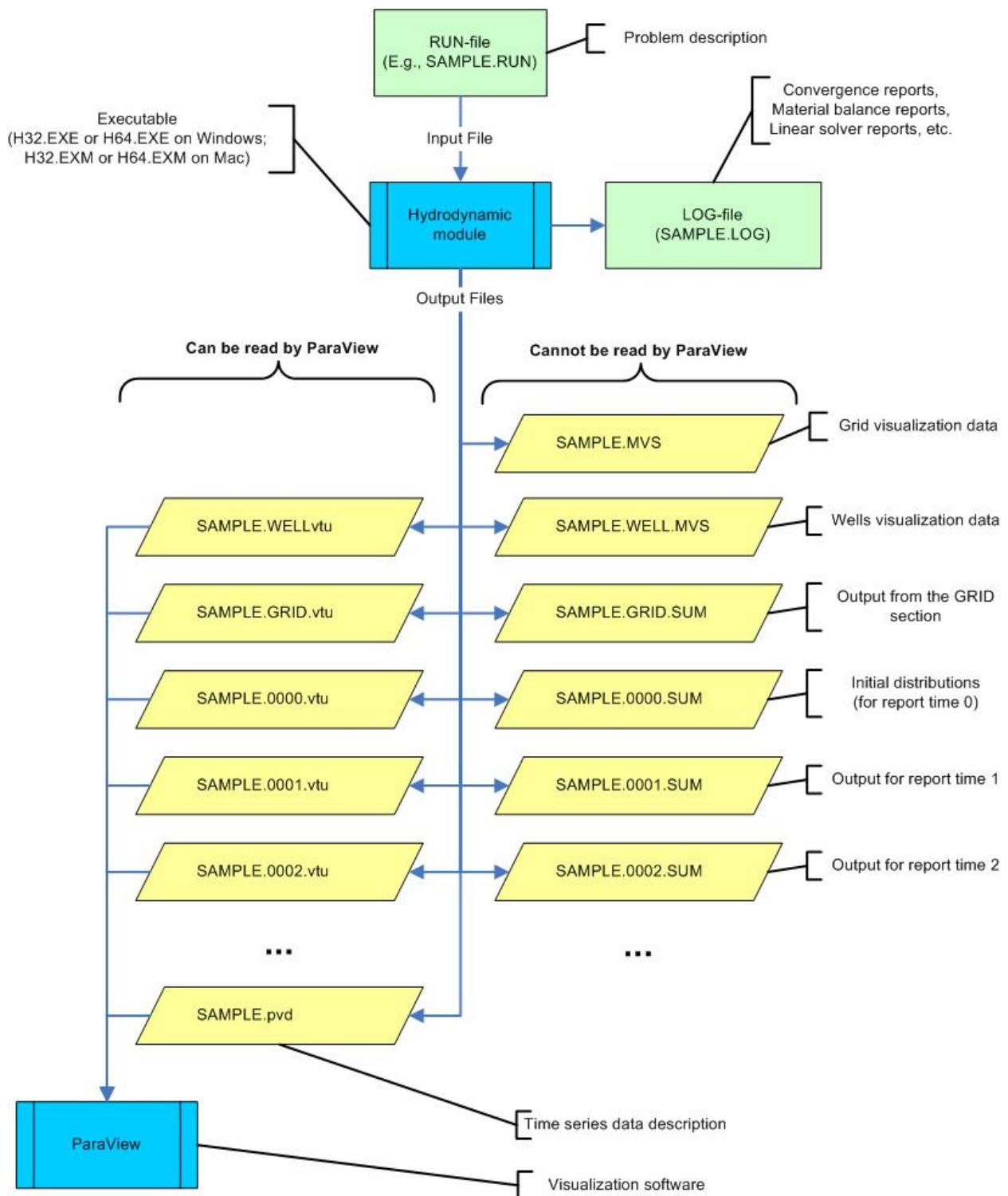


Figure 1.2: MUFITS files

2 Day 1. Fundamentals

MUFITS Training Course

Day 1
Fundamentals

Program

- Introduction (mathematical model, EOS-modules)
- RUN-file (input data preparation)
- Scenario1 (the first example)
- Fundamentals (Loading arrays, initial conditions, report times, etc.)
- Heat conduction (Scenario 2)

2.1. Introduction

Introduction

Day 1. Fundamentals

3

Mathematical model

Equations for multicomponent multiphase flows in porous medium:

$$\frac{\partial}{\partial t} \left(\phi \sum_{i=1}^p \rho_i c_{i(j)} s_i \right) + \operatorname{div} \left(\sum_{i=1}^p \rho_i c_{i(j)} \mathbf{w}_i + \boldsymbol{\Psi}_{(j)} \right) = q_{(j)}, \quad j = 1, \dots, c \quad \text{- mass balance equations}$$

$$\frac{\partial}{\partial t} \left(\phi \sum_{i=1}^p \rho_i e_i s_i + (1-\phi) \rho_r e_r \right) + \operatorname{div} \left(\sum_{i=1}^p \rho_i h_i \mathbf{w}_i + \boldsymbol{\Psi}_{(e)} \right) = q_{(e)} \quad \text{- energy balance equation}$$

$$\mathbf{w}_i = -K \frac{k_{r,i}}{\mu_i} (\operatorname{grad} P_i - \rho_i \mathbf{g}) \quad \text{- Darcy correlation}$$

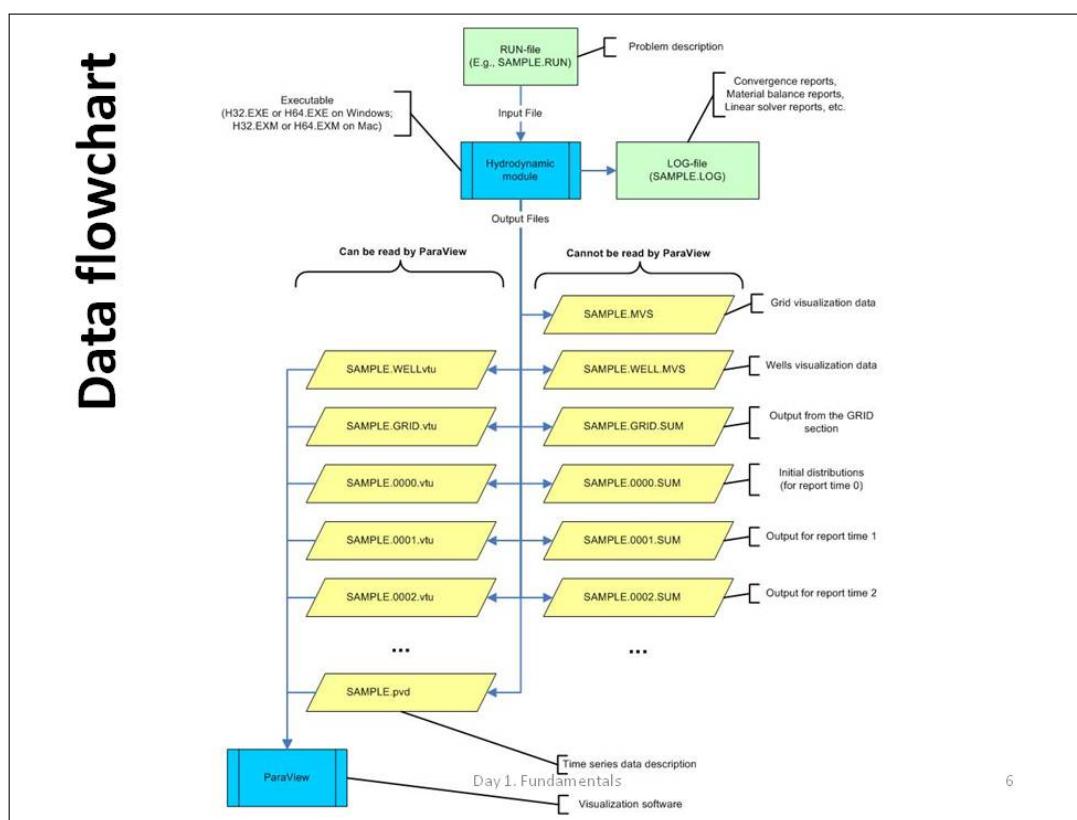
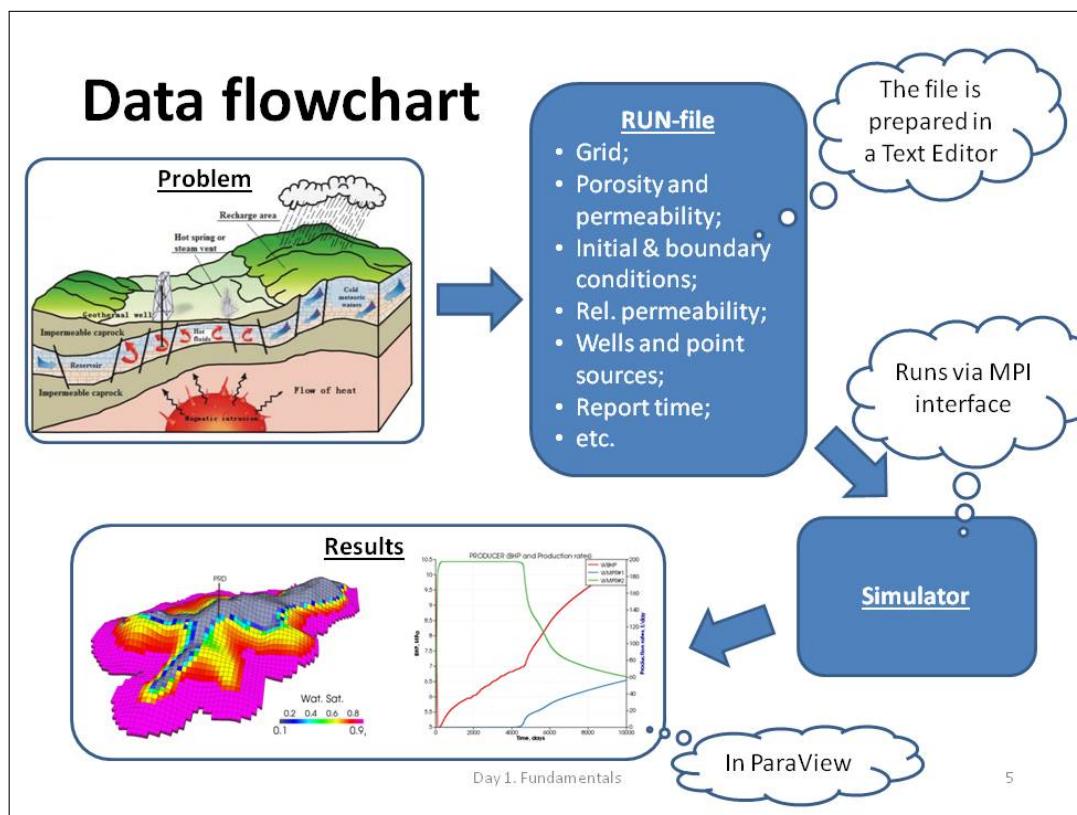
$$k_{r,i} = k_{r,i}(s), \quad P_i - P_k = P_{c,ik}(s) \quad \text{- saturation functions (rel. perm. & cap. pres.)}$$

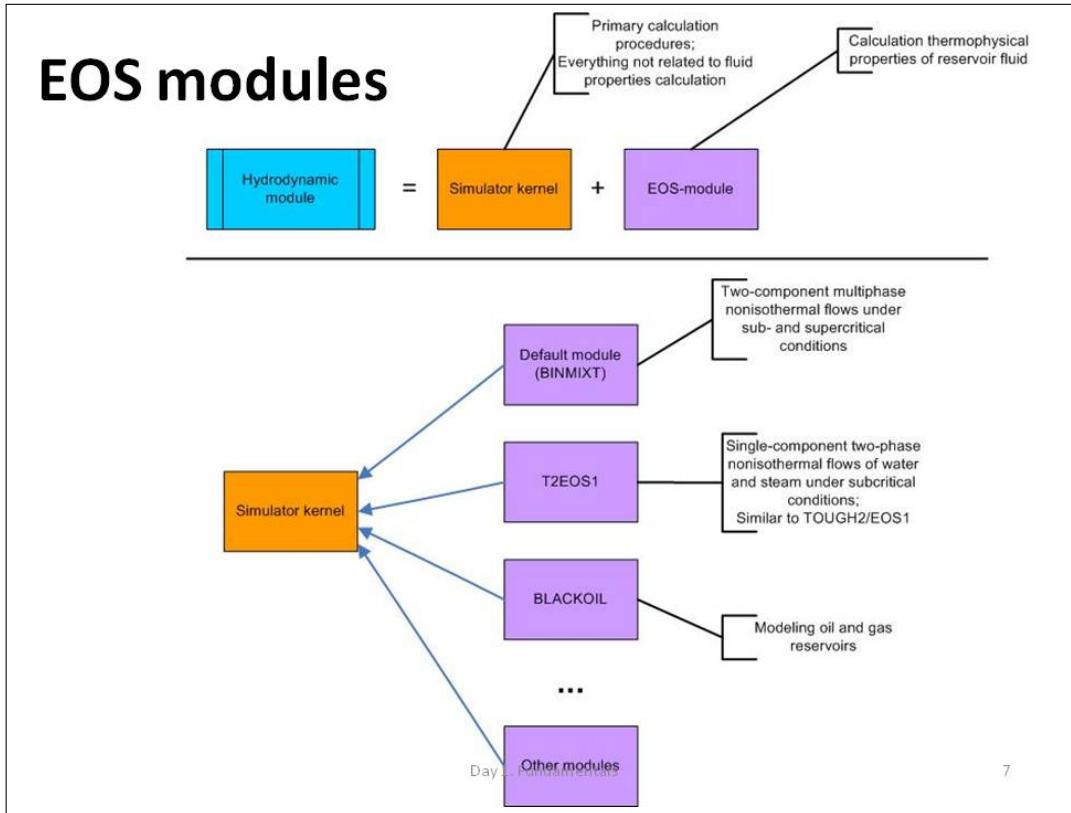
$$\sum_{i=1}^p s_i = 1, \quad \sum_{j=1}^c c_{i(j)} = 1 \quad \text{- consistency relations}$$

+ Equations of state (for fluid and rocks)

Day 1. Fundamentals

4





EOS module T2EOS1

Single-component (H_2O) single-phase and two-phase flows of after and vapor under subcritical conditions. This module is similar to TOUGH2/EOS1.

$$\frac{\partial}{\partial t}(\phi(\rho_w s_w + \rho_v s_v) + \text{div}(\rho_w w_w + \rho_v w_v)) = q$$

$$\frac{\partial}{\partial t}(\phi(\rho_w e_w s_w + \rho_v e_v s_v) + (1-\phi)\rho_r e_r) + \text{div}(\rho_w h_w \mathbf{w}_w + \rho_v h_v \mathbf{w}_v - \bar{\lambda} \mathbf{grad}T) = q_{(e)}$$

$$\mathbf{w}_i = -K \frac{k_{r,i}}{\mu_i} (\mathbf{grad}P - \rho_i \mathbf{g}), \quad i = w, v$$

$$s_w + s_v = 1, \quad \bar{\lambda} = \phi(s_w \lambda_w + s_v \lambda_v) + (1-\phi)\lambda_r, \quad k_{r,j} = k_{r,i}(s_w)$$

$$\rho_i = \rho_i(P, T), \quad e_i = e_i(P, T), \quad h_i = h_i(P, T), \quad \mu_i = \mu_i(P, T), \quad \lambda_i = \lambda_i(P, T), \quad i = w, v$$

$$\text{if}(T < T_{eq}(P)) \quad s_w = 1$$

$$\text{if}(T > T_{eq}(P)) \quad s_v = 1$$

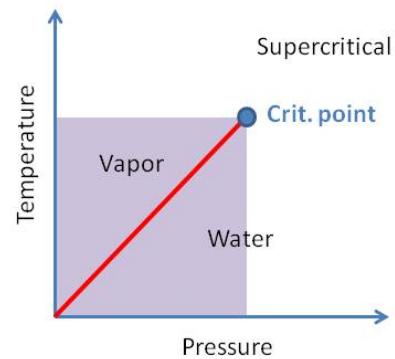
$$\text{if}(0 \leq s_w \leq 1)$$

EOS module T2EOS1

Single-component (H_2O) single-phase and two-phase flows of after and vapor under subcritical conditions. This module is similar to TOUGH2/EOS1.

Closing relations:

$\text{if}(T < T_{eq}(P)) \quad s_w = 1 \quad - \text{single-phase water}$
 $\text{if}(T > T_{eq}(P)) \quad s_v = 1 \quad - \text{single-phase vapor}$
 $\text{if}(0 \leq s_w \leq 1) \quad T = T_{eq}(P) \quad - \text{two-phase}$



2.2. RUN-file

**RUN-file
(or data-file)**

Day 1. Fundamentals

10

RUN-file (scenario 1)

Open RUN-file

Day 1. Fundamentals

11


```

96
97 PRES We equate pressure to 10 MPa
98   100*10 / in the whole domain (in all 100 blocks)
99
100 SWAT We equate water saturation to 0.6
101   100*0.6 /
102
103 RPTSUM We specify the properties which are
104   PRES SWAT SVAP TEMPC / saved at every report time (in the files
105
106 SCENARIO1.####.SUM (.vtu)). We specify
107   the output of pressure (PRES),
108   water saturation (SWAT),
109   vapor saturation (SVAP),
110   and temperature (TEMPC).
111 SCHEDULE ##### SCHEDULE section begins here #####
112
113 TSTEP We specify the output at the report times
114   6*10 2*20 / 10, 20, 30, 40, 50, 60, 80, 100 days.
115
116 POST ##### POST section begins here #####
117
118 CONVERT We enable the MUFITS output files
119   conversion to ParaView compatible
120   formats (the .vtu and .pvf files output
121   is initiated by this keyword).
122 END #####

```

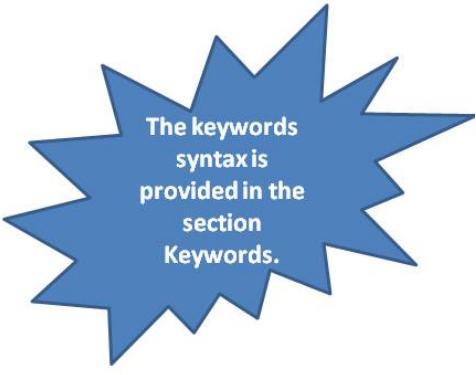
Keywords

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

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

Example of keywords

```
MAKE
    CART   20  1   5   /
XYZBOUND
    0    200   -5    5   1000 1050   /
ENDMAKE
PORO
    100x0.25 /
```



The keywords syntax is provided in the section **Keywords**.

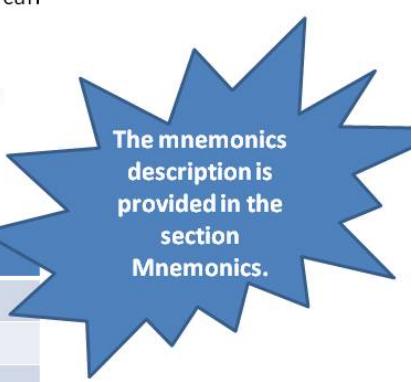
Day 1. Fundamentals

12

Mnemonics

Mnemonics are a short character references to physical, geological, geometrical, logical data, etc. Using mnemonics you can

- load data in simulation;
- perform operations on arrays;
- specify regions for thermophysical properties;
- specify initial or boundary conditions;
- specify output data.



The mnemonics description is provided in the section **Mnemonics**.

Examples of Mnemonics

Mnemonic	Parameter
PRES	Pressure (MPa)
TEMP	Temperature (K)
TEMPC	Temperature (C)
SWAT	Water saturation
SVAP	Vapor saturation

Day 1. Fundamentals

13

Comments in RUN-file

Commented out lines begin with: '--', '!', ' ', lowercase letters.

Day 1. Fundamentals

14

Repeated counts

A data item can be repeated in a keyword a number of times by using asterisk symbol.

```

1      Example of repeated co
2 BOX
3 -- imin-imax jmin-jmax kmin-kmax
4     1    4    1    3    1    1   /
5
6 -- This keyword
7
8 PORO
9   0.25  0.25  0.25  0.25
10  0.25  0.25  0.25  0.25
11  0.15  0.15  0.15  0.15 /
12
13 -- is equivalent to any of the following keywords
14
15 PORO
16   4*0.25
17   4*0.25
18   4*0.15 /
19
20 PORO
21   8*0.25 4*0.15 /
22
23 PORO
24   7*0.25 0.25 2*0.15 2*0.15 /

```

Day 1: Fundamentals

15

Default values

There are two options for defaulted parameters: by using asterisk or by the slash symbol.

```
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
```

Day 1. Fundamentals

16

2.3. Scenario 1

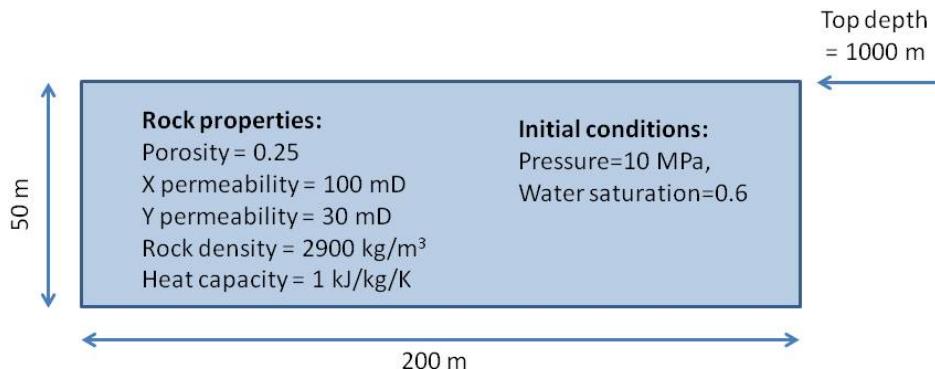
Scenario 1

Day 1. Fundamentals

17

Formulation of the problem

A simple gravity segregation 2D problem:

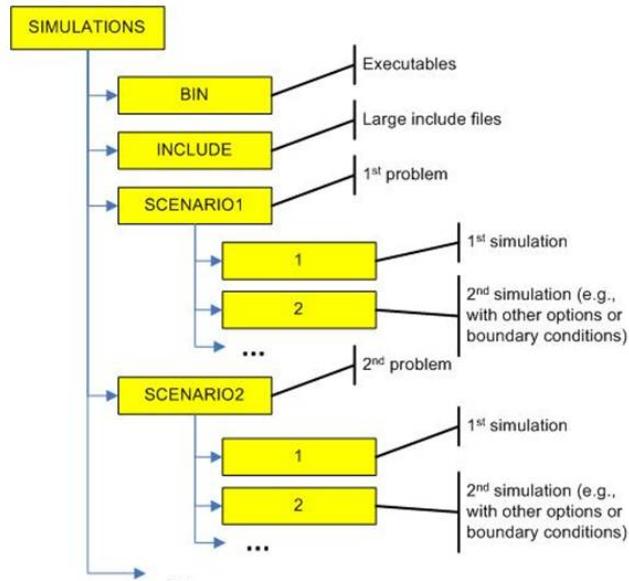


All boundaries of the domain are impermeable and insulated

RUN-file (scenario 1)

[Open RUN-file](#)

Folders hierarchy



Day 1. Fundamentals

20

Running the problem

In the folder SIMULATIONS/SCENARIO1/0/ execute

Mac:

```
mpirun -n 1 ../../BIN/H64.EXM SCENARIO1.RUN > SCENARIO1.LOG
```

Linux:

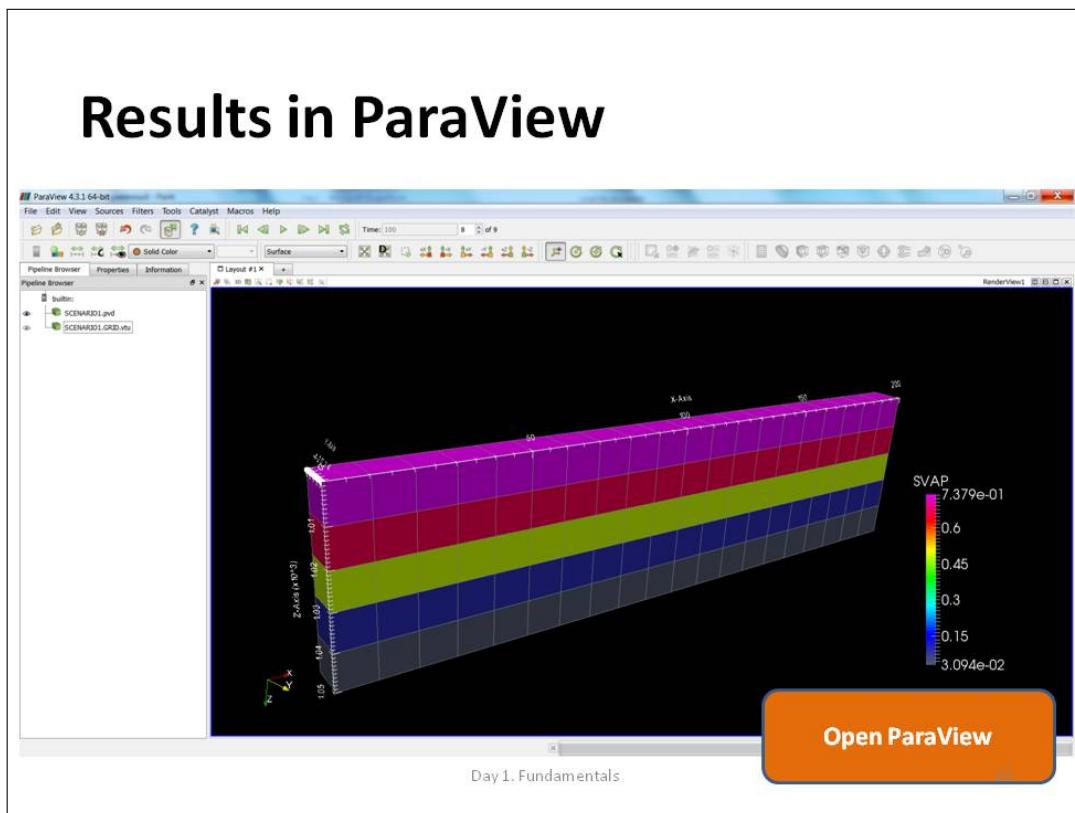
```
mpirun -n 1 ../../BIN/H64.EXL SCENARIO1.RUN > SCENARIO1.LOG
```

Windows:

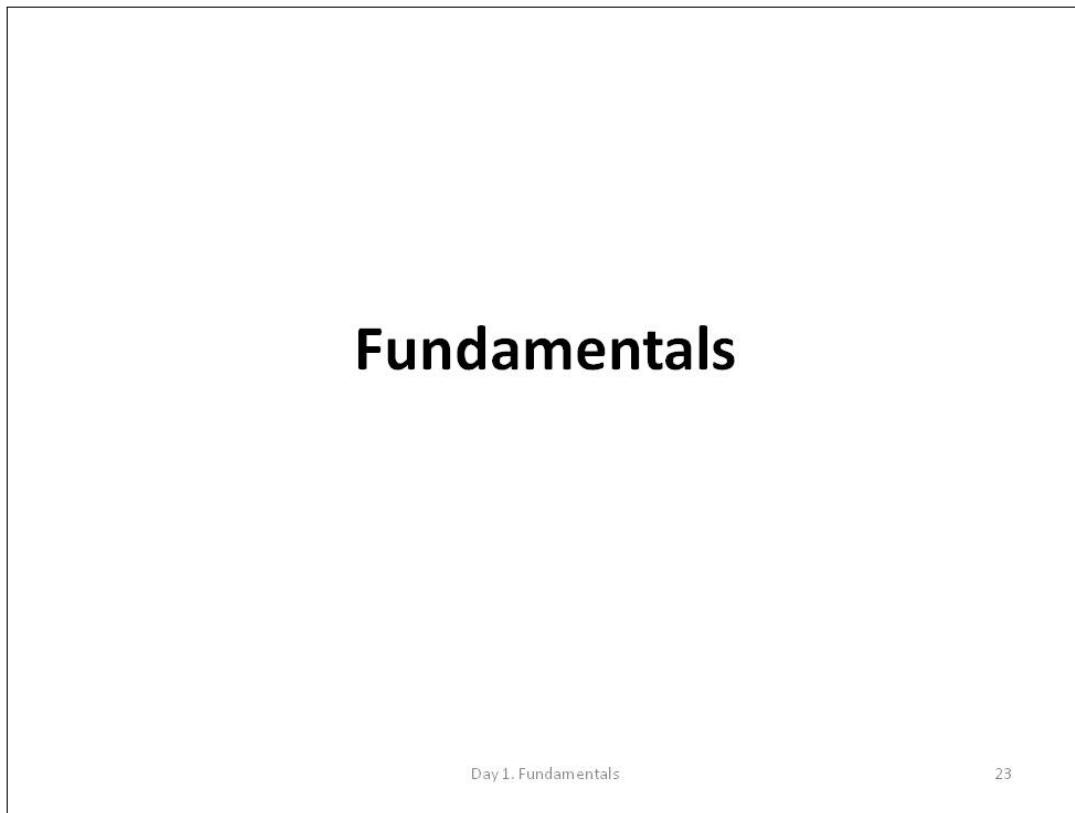
```
"...mpiexec.exe" -n 1 ../../BIN/H64.EXE SCENARIO1.RUN > SCENARIO1.LOG
```

Day 1. Fundamentals

21



2.4. Fundamentals



Control on output data

The output in the file SCENARIO1.GRID.SUM is controlled by the RPTGRID keyword

```
RPTGRID syntax
1 -- in GRID section
2
3 RPTGRID
4 mnemonic1 mnemonic2 mnemonic3 ... /
5
6 =====
7
8 mnemonic# - is the mnemonic of a property saved in the file *.GRID.SUM.
9 If one of the mnemonics is ASCII then the formatted file
10 is saved.
```

Exercise: Add output of grid blocks coordinates

The output in the file SCENARIO1.#####.SUM is controlled by the RPTSUM keyword

```
RPTSUM syntax
1 -- in INIT or SCHEDULE section
2
3 RPTSUM
4 mnemonic1 mnemonic2 mnemonic3 ... /
5
6 =====
7
8 mnemonic# - is the mnemonic of a property saved in the files *.0000.SUM,
9 *.0001.SUM, *.0000.SUM, etc. If one of the mnemonics is ASCII
10 then the formatted file is saved.
```

Exercise : Add output of water and vapor densities

24

Cartesian grids

The number of grid blocks along every axis is defined by the keyword **MAKE**

```
MAKE-ENDMAKE syntax
1 -- in GRID section
2
3 MAKE
4 gridtype ni nj nk /
5
6 -- other keywords
7
8 ENDMAKE
9
10 =====
11
12 gridtype = CART - Cartesian Grid
13 = RADIAL - Radial Grid
14 = CORNER - Corner-Point grid
15
16 ni - number of grid blocks along i-indexation axis
17 nj - number of grid blocks along j-indexation axis
18 nk - number of grid blocks along k-indexation axis
```

Exercise : Re-simulate the Scenario 1 by using grid 25*10 blocks

Day 1. Fundamentals

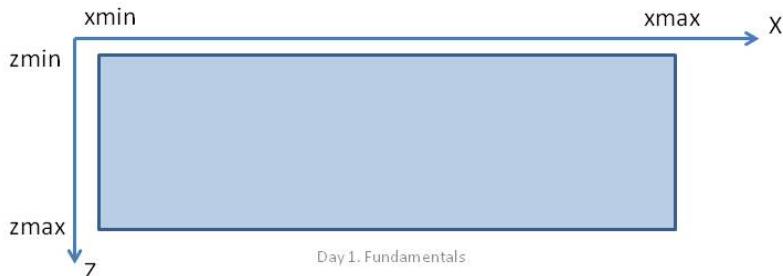
25

Cartesian grids

The domain boundaries are defined by the keyword **XYZBOUND**

```
XYZBOUND syntax
1 -- within MAKE/ENDMAKE brackets.
2
3 XYZBOUND
4   xmin xmax ymin ymax zmin zmax xincr yincr zincr /
5 ====
6
7   xmin/xmax - the domain boundaries along axis x (xmin<xmax)
8   ymin/ymax - the domain boundaries along axis y (ymin<ymax)
9   zmin/zmax - the domain boundaries along axis z (zmin<zmax)
```

Exercise : Locate the top boundary of the reservoir at depth 900 meters, whereas the bottom boundary at depth 950 meters.



Day 1. Fundamentals

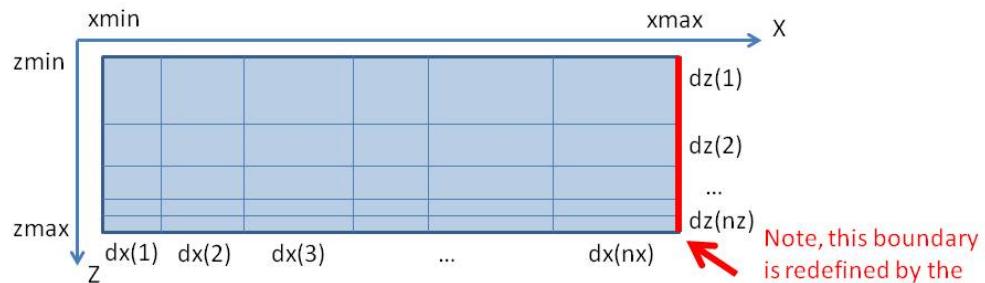
26

Cartesian grids

The grid block extensions can be redefined using the **DXV**, **DYV**, **DZV** keywords

```
DXV syntax
1 -- within MAKE-ENDMAKE brackets
2
3 DXV
4   dx(1) dx(2) dx(3) ... dx(nx) /
5 ====
6
7   dx(#)- grid blocks extensions along axis X.
8   nx - number of grid block along axis X. nx is the 2nd argument of the
9     keyword MAKE.
```

Exercise : Introduce a refined grid at the bottom boundary and re-simulate the Scenario 1



Day 1. Fundamentals

Note, this boundary is redefined by the keyword DXV²⁷

Arrays loading

```

1 -- in GRID section          PORE syntax
2
3 PORE
4   value1  value2  value3 ... value(nbox) /
5
6 =====
7
8   value(#) - value assigned to the corresponding grid block in the input box.
9     The grid blocks are ordered with i index cycling fastest,
10    followed by j and k indexes.
11   nbox   - number of grid blocks in the current input box.
12   nbox=(imax-imin+1)*(jmax-jmin+1)*(kmax-kmin+1), where
13     imin, imax, jmin ,jmax, kmin, kmax are defined by the keyword
14   BOX.

```

An example for the grid 4*1*3:

i-index

	value 1	value 2	value 3	value 4
k-index	value 5	value 6	value 7	value 8
	value 9	value 10	value 11	value 12

Day 1. Fundamentals

Note:

i-index – X axis
j-index – Y axis
k-index – Z axis

28

Arrays loading

An example for the grid 4*1*3:

i-index

	value 1	value 2	value 3	value 4
k-index	value 5	value 6	value 7	value 8
	value 9	value 10	value 11	value 12

Note:

i-index – X axis
j-index – Y axis
k-index – Z axis

Exercise : Increase fivefold the permeability of the layers 3 and 4 and re-simulate scenario 1

Day 1. Fundamentals

29

Arrays loading

An example for the grid 4*1*3:

i-index			
k-index	value 1	value 2	value 3
	value 5	value 6	value 7
	value 9	value 10	value 11

Note:

i-index – X axis
j-index – Y axis
k-index – Z axis

Exercise : Increase fivefold the permeability in the columns i=8-13 and re-simulate the scenario 1

Arrays loading

An example for the grid 4*1*3:

i-index			
k-index	value 1	value 2	value 3
	value 5	value 6	value 7
	value 9	value 10	value 11

Note:

i-index – X axis
j-index – Y axis
k-index – Z axis

Exercise : In the left half of the domain the initial pressure is 11 MPa and the water saturation is 0.35. In the right half of the domain the initial pressure is 10 MPa and the water saturation is 0.7. Impose these initial conditions and re-simulate scenario 1.

Keyword BOX

```

1 -- in all sections except RUNSPEC and POST           BOX syntax
2
3 BOX
4   imin imax  jmin jmax  kmin kmax /
5
6 =====
7
8   imin/imax - the boundaries of the input box along i-indexation axis.
9     By default imin=1 and imax=ni, where ni is the 2nd argument
10    of the keyword MAKE.
11   jmin/jmax - the boundaries of the input box along j-indexation axis.
12     By default jmin=1 and jmax=nj, where nj is the 3rd argument
13    of the keyword MAKE.
14   kmin/kmax - the boundaries of the input box along k-indexation axis.
15     By default kmin=1 and kmax=nk, where nk is the 4th argument
16    of the keyword MAKE.

```

The **BOX** keyword allows to select a region of the reservoir for arrays input. The arrays in the selected region are loaded on a block by block basis, as this region is the whole domain.

The **ENDBOX** keyword resets the input box so that it encompasses the whole domain.

Note that at the beginning of every section the input box is reset by the program.

Day 1. Fundamentals

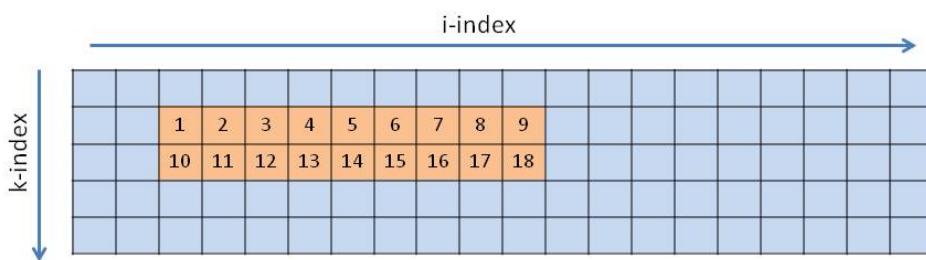
32

Keyword BOX

For example, the keyword

BOX
3 11 2* 2 3 /

selects the following region for arrays input

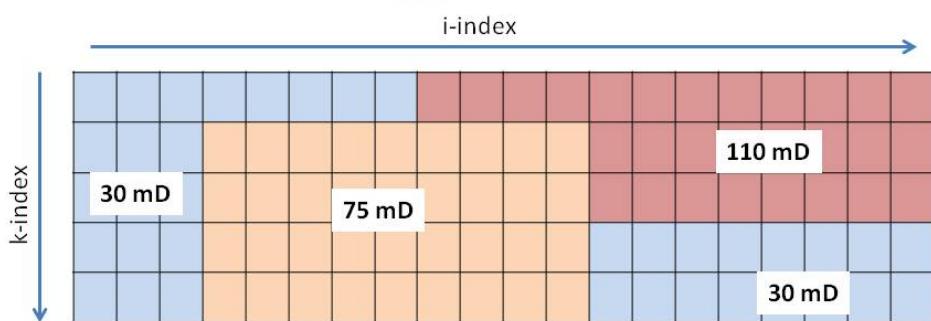


Day 1. Fundamentals

33

Keyword BOX (Task)

Exercise : Create the following initial distribution of Z direction permeability and re-simulate the Scenario 1.



Day 1. Fundamentals

34

Rock properties

The rock thermophysical properties must be specified within brackets **ROCK-ENDROCK**.
The rock density and heat capacity can be defined using the **ROCKDH** keyword:

```

1 -- within ROCK-ENDROCK brackets          ROCKDH syntax
2
3 ROCKDH
4   dens  heatcap /
5
6 =====
7   dens    - rock density
8   heatcap - rock heat capacity
9

```

Day 1. Fundamentals

35

Relative permeabilities

The saturation functions (e.g., the relative permeabilities and capillary pressure) must be specified within brackets **SAT-ENDSAT**. They can be defined using the **SATTAB** keyword.

```

1 -- within SAT-ENDSAT brackets
2
3 SATTAB
4   sliq1  krliq1  krgas1  pcap1 /
5   sliq2  krliq2  krgas2  pcap2 /
6   sliq3  krliq3  krgas3  pcap3 /
7 ...
8 /
9 =====
10
11
12   sliq# - liquid phase saturation
13   krliq# - liquid phase relative permeability
14   krgas# - gas phase relative permeability
15   pcap# - capillary pressure

```

Day 1. Fundamentals

36

Relative permeabilities (example)

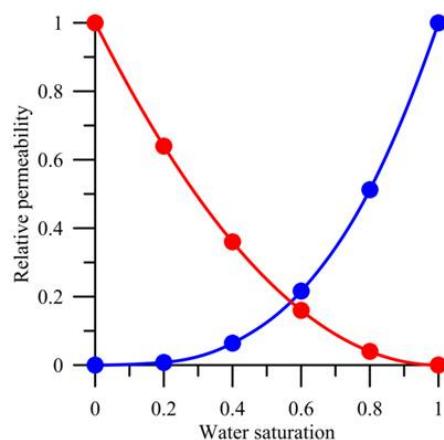
```

SATTAB
-- sliq    krliq    krgas
  0.0    0.0     1.0  /
  0.2    0.008   0.64 /
  0.4    0.064   0.36 /
  0.6    0.216   0.16 /
  0.8    0.512   0.04 /
  1.0    1.0     0.0  /

```

$$k_{r,wat}(s_{wat}) = s_{wat}^3$$

$$k_{r,vap}(s_{wat}) = (1-s_{wat})^2$$



Day 1. Fundamentals

37

Relative permeabilities in ParaView

The relative permeabilities can be loaded in ParaView by the **RPTSATTA** keyword. This keyword invokes the output of CSV file with a table for relative permeabilities (and capillary pressure).

```

1 -- in PROPS section
2
3 RPTSATTA
4   satnum filename n smin smax /
5
6 =====
7
8   satnum - SATNUM region ID for which the output is requested;
9   filename - the file name in which saturation functions are saved. This
10    file should have extension .csv;
11   n       - number of output points in the interval [smin,smax];
12   smin-smax - the boundaries of the output interval.

```

Exercise : Load and create a plot for the relative permeability curves in ParaView

Day 1. Fundamentals

38

Relative permeabilities (exercise)

Exercise : Re-simulate scenario 1 by using Corey relative permeabilities
 $S_{l,min}=0.2$
 $S_{g,min}=0.05$
 $n_l=2.5$
 $n_g=1.5$.
Build the relative permeability plots in ParaView

Corey curves:

$$k_{r,liq} = \left(\frac{s_{liq} - s_{liq,min}}{1 - s_{liq,min} - s_{gas,min}} \right)^{n_{liq}}$$

$$k_{r,gas} = \left(\frac{1 - s_{liq} - s_{gas,min}}{1 - s_{liq,min} - s_{gas,min}} \right)^{n_{gas}}$$

Day 1. Fundamentals

39

Relative permeabilities (exercise)

The answer is:

SATTAB	0.2	0.0	1.0	/
	0.4	0.0367	0.6279	/
	0.6	0.2077	0.3187	/
	0.8	0.5724	0.0894	/
/	0.95	1.0	0.0	/

Note, that the critical saturations must be defined.

Initial conditions

The initial conditions for every cell must be specified in the **INIT** section

There are 5 options to impose the initial conditions when using T2EOS1 module:

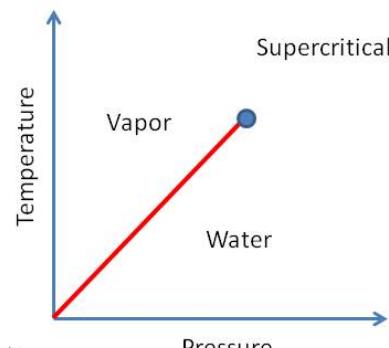
- 1) Specify pressure (PRES) and water saturation (SWAT) [priority 1]
- 2) Specify pressure (PRES) and vapor saturation (SVAP) [priority 2]
- 3) Specify temperature (TEMP) and water saturation (SWAT) [priority 3]
- 4) Specify temperature (TEMP) and vapor saturation (SVAP) [priority 4]
- 5) Specify pressure (PRES) and temperature (TEMP) [priority 5]

2-phase

1-phase

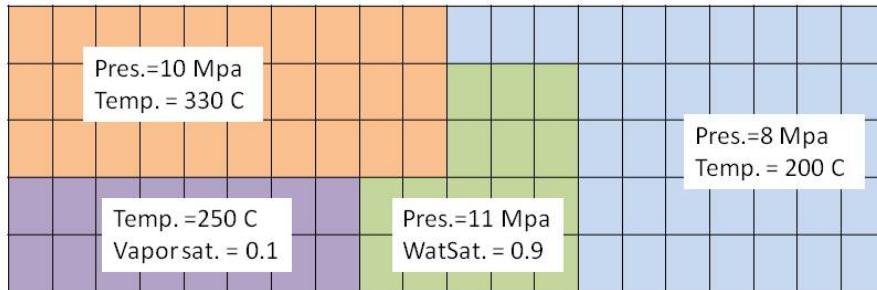
Note, that the TEMPC mnemonic can be used to specify the temperature in degrees of Celsius instead of the TEMP mnemonic which is for degrees of Kelvin.

Note, that if parameters in a cell remain unspecified the simulation will be terminated after the INIT section.



Initial conditions (Exercise)

Exercise : Specify the following initial conditions and re-simulate Scenario 1



Day 1. Fundamentals

42

Report times

Note that time dimension is Days.

The report times are the times at which the output is saved. These can be specified in **SCHEDULE** section by the **TSTEP** keyword. When the simulator encounters this keyword it proceeds the simulation further in time and produces the required outputs.

```

-- 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.

```

Exercise : Re-simulate Scenario 1 reporting parameters distributions at the times: 10, 15, 30, 70, 100, 150, 500 days

Day 1. Fundamentals

43

Report times

```
1 -- in SCHEDULE section
2
3 TIMES
4   time1 time2 time3 ... /
5
6 =====
7
8   time# - report times (in days). Every time# initiates the
9     program to advance the simulation to this time and save a new
10    *.###_SUM file.
```

Exercise Task: Re-simulate Scenario 1 reporting parameters distributions at the times: 10, 15, 30, 70, 100, 150, 500 days and using the TIMES keyword

Exercise : Re-simulate Scenario 1 reporting parameters distributions at the times: 10, 15, 30, 70, 100, 150, 500 days and using both TIMES and TSTEP keywords in the same simulation.

2.5. Heat conduction (Scenario 2)

Heat conduction

Day 1, Fundamentals

45

Heat conduction option

Heat conduction coefficient:

$$\lambda = \phi(s_w \lambda_w + s_v \lambda_v) + (1 - \phi) \lambda_r$$

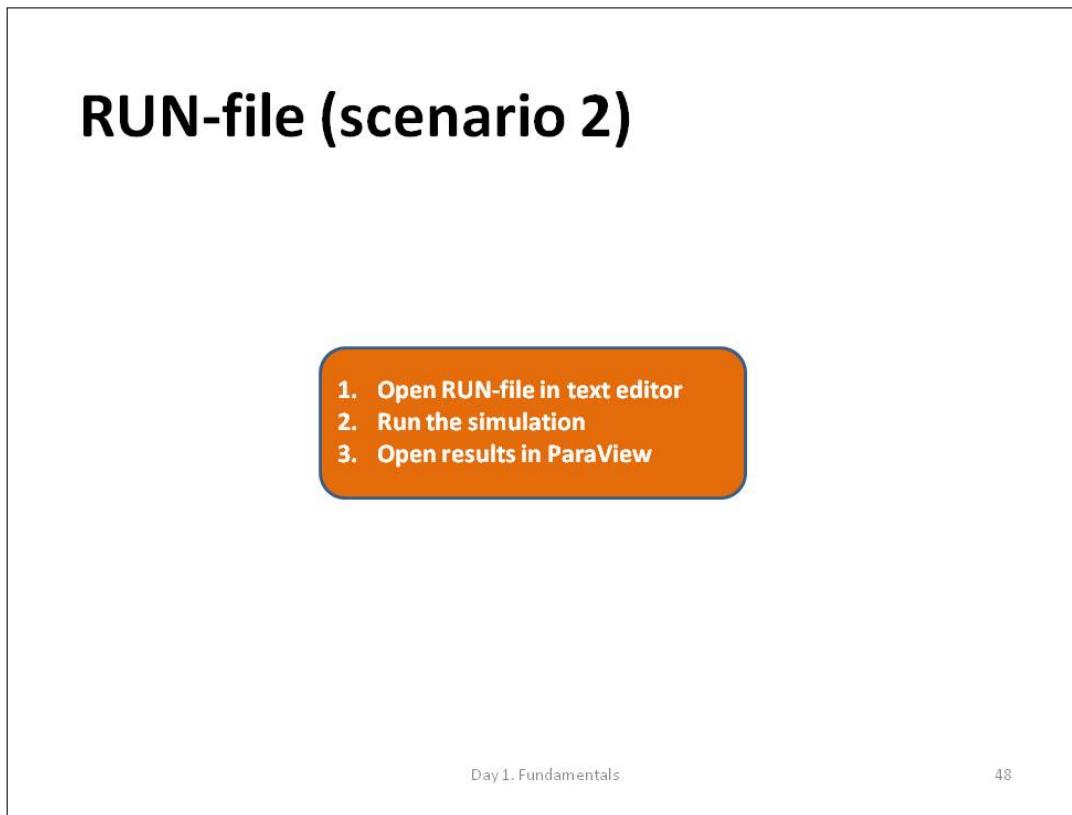
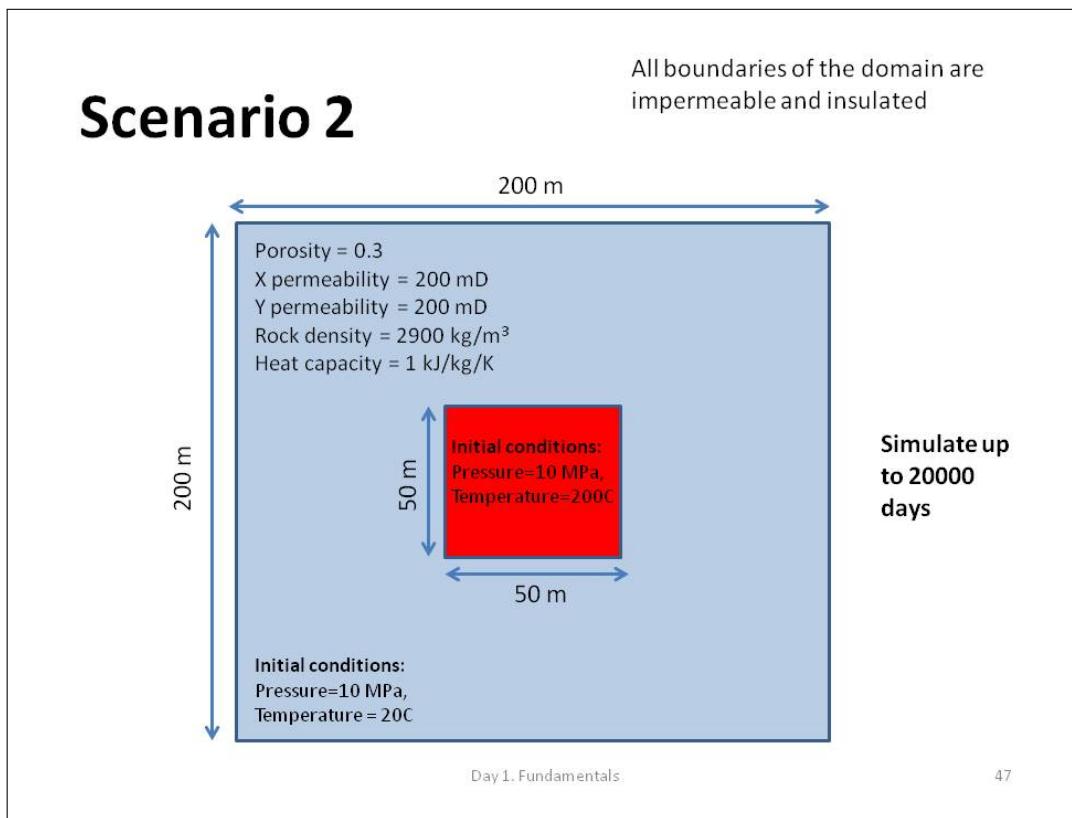

HCFLUID HCROCK

The HCFLUID and HCROCK enable heat conduction modelling. The keywords must be specified in the RUNSPEC section.

The rock heat conduction coefficient distribution must be loaded in the GRID section using HCONDCAF, HCONDCAFY and HCONDCAFZ keywords.

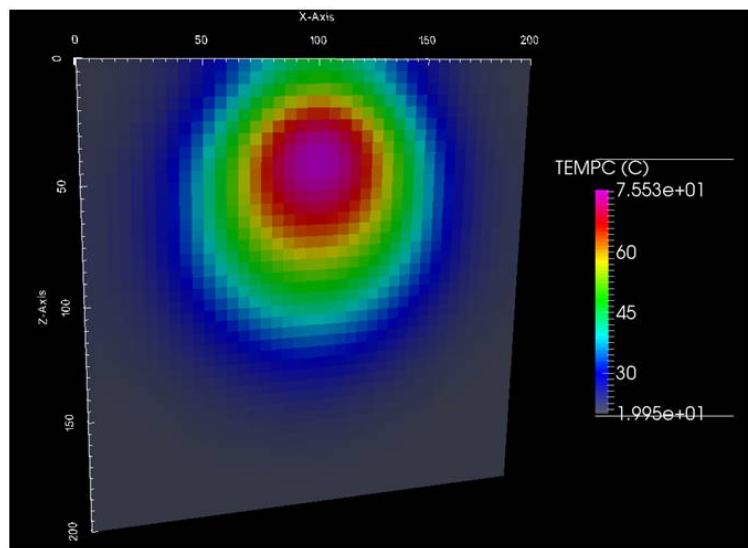
Day 1, Fundamentals

46



143 END #####

Result (Scenario 2, t=10000 days)



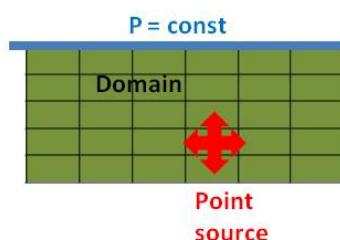
49

Next day

- Operations on arrays
- Regions
- Boundary conditions
- Point sources

Arithmetic operations

$$\begin{aligned} \text{PORO} &= 0.2 \\ \text{PERMZ} &= 0.1 * \text{PERMX} \\ \text{PRES} &= 8.5 + 0.01 * \text{DEPTH} \end{aligned}$$



Day 1. Fundamentals

50

3 Day 2. Operations on arrays, Regions & Boundary conditions

MUFITS Training Course

Day 2

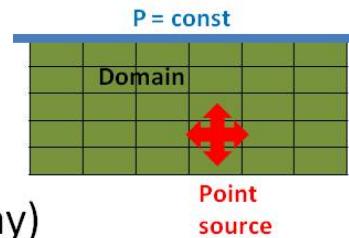
**Operations on arrays, Regions,
Boundary conditions & Point sources**

Program

Arithmetic operations

$$\begin{aligned} \text{PORO} &= 0.2 \\ \text{PERMZ} &= 0.1 * \text{PERMX} \\ \text{PRES} &= 8.5 + 0.01 * \text{DEPTH} \end{aligned}$$

- Operations on arrays (Scenario 3)
- Regions
- Boundary conditions
- Point sources
- Scenario 4 (summary of the day)



Day 2, Operations on arrays, Regions,
Boundary conditions & Point sources

2

3.1. Operations on arrays (Scenario 3)

Operations on arrays

Day 2. Operations on arrays, Regions,
Boundary conditions & Point sources

3

Keywords for operations on arrays in a box of grid blocks

Keyword	Result
ADD	Add
COPY	Copy from one array into another array
EQUALS	Equate to
MAXVALUE	Apply maximum limit
MINVALUE	Apply minimum limit
MULTIPLY	Multiply by
OPERATE	Apply a complicated arithmetic operation

Day 2. Operations on arrays, Regions,
Boundary conditions & Point sources

4

Keywords EQUALS

```

EQUALS syntax
-- in all sections except RUNSPEC and POST

EQUALS
mnemonic1 value1 imini1 imax1 jmini1 jmax1 kmini1 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.

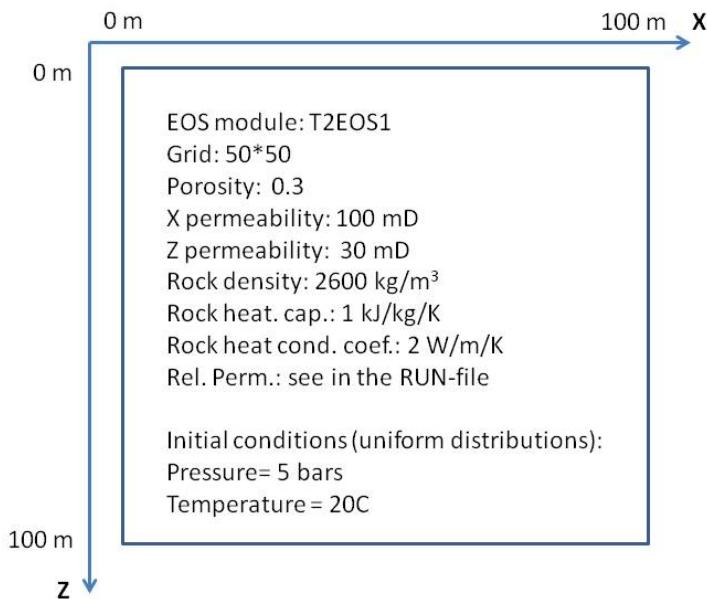
```

Day 2. Operations on arrays, Regions,
Boundary conditions & Point sources

5

Scenario3

All boundaries are
impermeable and insulated



Day 2. Operations on arrays, Regions,
Boundary conditions & Point sources

6

RUN-file (Scenario 3)

1. Open RUN-file in text editor
2. Run the simulation
3. Open results in ParaView

Day 2. Operations on arrays, Regions,
Boundary conditions & Point sources

7

SIMULATIONS/SCENARIO3/0/SCENARIO3.RUN

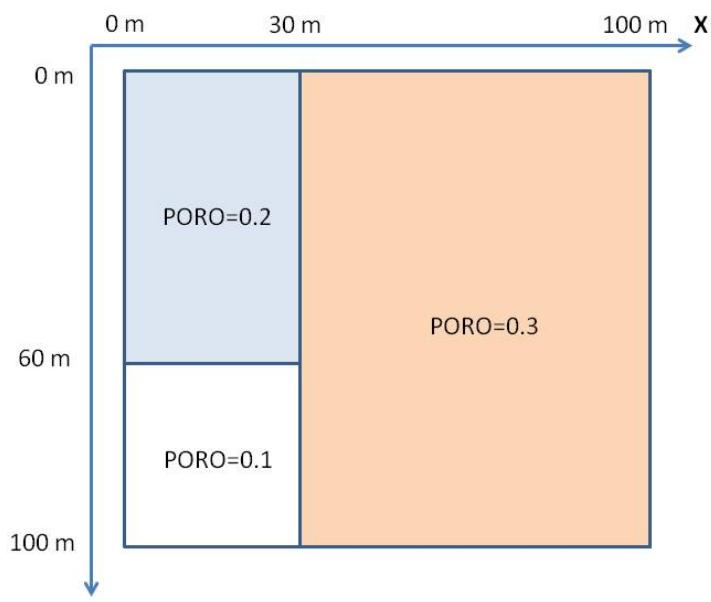
```
1 -----
2
3 Note:
4 We type all comments in lowercase letters while all keywords and mnemonics
5 must be in uppercase letters.
6 -----
7 -----
8
9 Note:
10 -- any line beginning with '!' or '--' is a comment line;
11 any line not beginning with an uppercase letter outside keyword
12 ! instruction is also a comment line.
13 -----
14 -----
15
16 Note:
17 Any data line must be terminated by the slash '/' sign.
18 -----
19 -----
20
21 Note:
22 We denote by the repeated symbols the following structural elements of this
23 RUN-file:
24 ##### - delimits the sections of the RUN-file.
```



```
121  
122 POST ##### POST section begins here #####  
123  
124 CONVERT We enable the MUFITS output files  
125 conversion to ParaView compatible  
126 formats (the .vtu and .pvf files output  
127 is initiated by this keyword).  
128 END #####
```

Using EQUALS keyword

Exercise: Specify the following porosity distribution using EQUALS and re-simulate scenario 3.



Answer

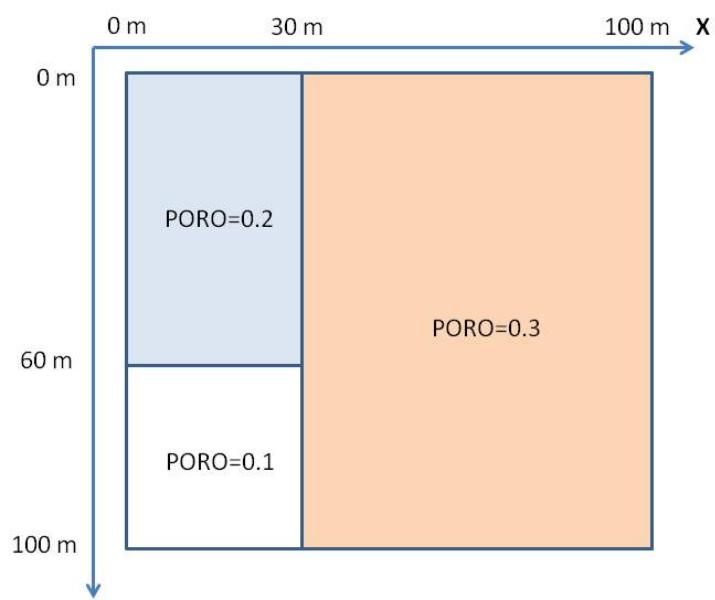
```
Day 2. Task 1
1 -- answer 1:
2
3 EQUALS
4   PORO 0.3 /
5   PORO 0.2 1 15 2* 1 30 /
6   PORO 0.1 4* 31 50 /
7 /
8
9 -- answer 2:
10
11 EQUALS
12   PORO 0.2 /
13   PORO 0.1 4* 31 50 /
14   PORO 0.3 16 50 2* 1 50 /
15 /
16 ...
17 ...
```

Day 2. Operations on arrays, Regions,
Boundary conditions & Point sources

9

Operations on arrays (exercise)

Exercise : Specify
the following
porosity
distribution using
each of EQUALS,
ADD and
MULTIPLY
keywords, and
re-simulate
scenario 3.



Day 2. Operations on arrays, Regions,
Boundary conditions & Point sources

10

Keyword OPERATE

```

OPERATE syntax
1 -- in all sections except RUNSPEC and POST
2
3 OPERATE
4   mnem1 imini1 imax1 jmini1 jmax1 kmini1 kmax1 oper1 mdep1 pari_1 par2_1 par3_1 /
5   mnem2 imini2 imax2 jmini2 jmax2 kmini2 kmax2 oper2 mdep2 pari_2 par2_2 par3_2 /
6   mnem3 imini3 imax3 jmini3 jmax3 kmini3 kmax3 oper3 mdep3 pari_3 par2_3 par3_3 /
7   ...
8   /
9 =====
```

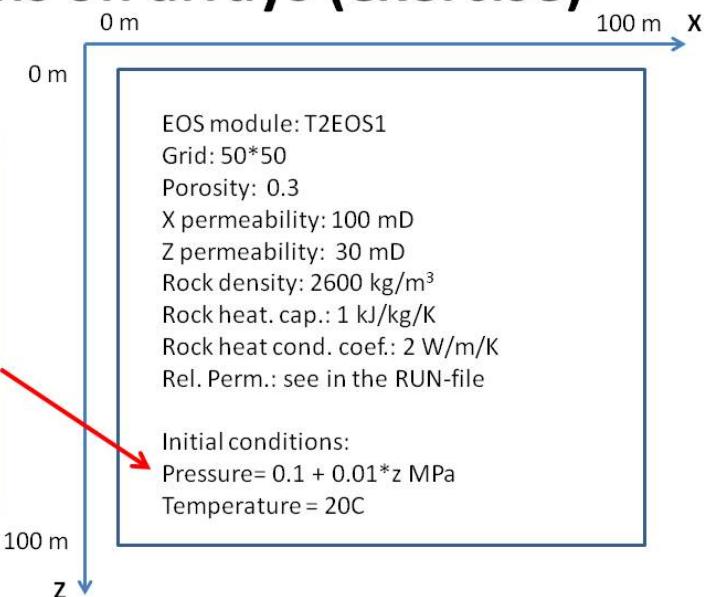
See also the list of available operations in the keyword syntax description

Day 2. Operations on arrays, Regions, Boundary conditions & Point sources

11

Operations on arrays (exercise)

Exercise: Specify the hydrostatic initial distribution of pressure and re-simulate scenario 3



Day 2. Operations on arrays, Regions, Boundary conditions & Point sources

12

Answer

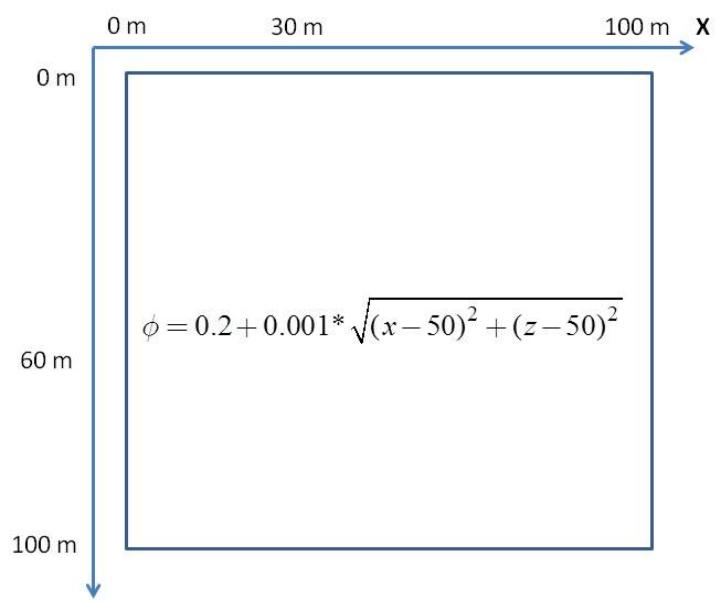
```
Day 2. Task 3
1 -- in INIT section
2
3 OPERATE
4 PRES DEPTH MULTA 0.1 0.01 /
5 /
```

Day 2. Operations on arrays, Regions,
Boundary conditions & Point sources

13

Operations on arrays (exercise)

Exercise: Specify
the following
porosity
distribution
re-simulate
scenario 3.



Day 2. Operations on arrays, Regions,
Boundary conditions & Point sources

14

Answer

Day 2. Task 4

```
1 OPERATE
2   AUXARR1 XCOORD COPY    /
3   AUXARR1 1*   ADD     -50  /
4   AUXARR1 AUXARR1 MULTP  1    2  /
5   AUXARR2 DEPTH  COPY    /
6   AUXARR2 1*   ADD     -50  /
7   AUXARR2 AUXARR2 MULTP  1    2  /
8   AUXARR1 AUXARR2 ADDARR /
9   AUXARR1 AUXARR1 MULTP  1    0.5 /
10  AUXARR1 AUXARR1 MULTA  0.2  0.001 /
11  PORO    AUXARR1 COPY    /
12 /
```

Day 2. Operations on arrays, Regions,
Boundary conditions & Point sources

15

3.2. Regions

Regions

Day 2. Operations on arrays, Regions,
Boundary conditions & Point sources

16

Flags defined in cells

Flag mnemonic	Description
ACTNUM	0 – cell inactive; 1 – cell active; 2 – fixed parameters;
TYPENUM	1 – an ordinary cell, 2 – impermeable cell
ROCKNUM	Rock properties region number
SATNUM	Saturation functions region number
FLUXNUM	Is used for boundary conditions specification
MPINUM	Grid partition
INCONUM	No predefined meaning at present
...	...

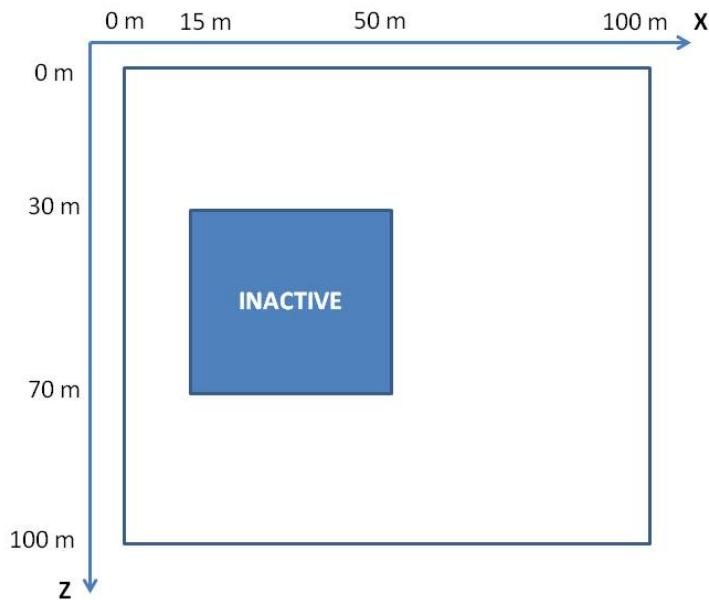
The flags can be defined in the GRID section using operations on arrays.

Day 2. Operations on arrays, Regions,
Boundary conditions & Point sources

17

ACTNUM flag

0 – cell inactive; 1 – cell active; 2 – fixed parameters.



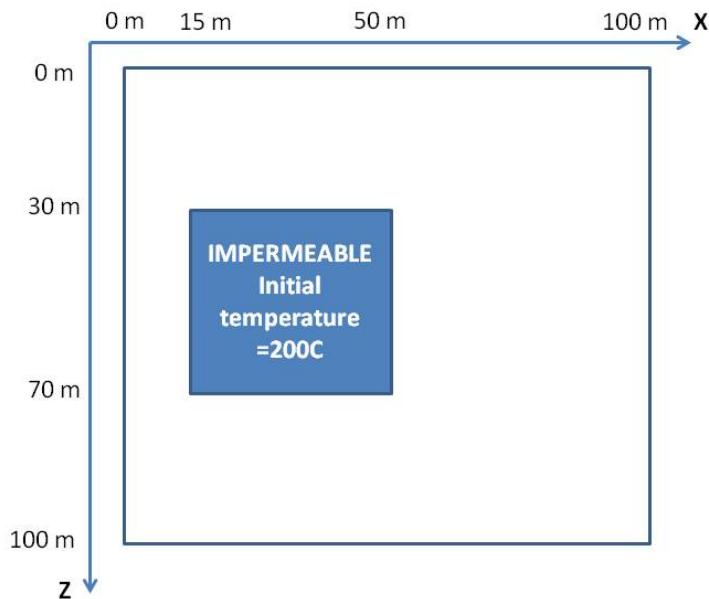
Exercise: Specify this region of the reservoir inactive and re-simulate Scenario 3.

Day 2. Operations on arrays, Regions, Boundary conditions & Point sources

18

TYPENUM flag

1 – ordinary cell; 2 – impermeable cell.



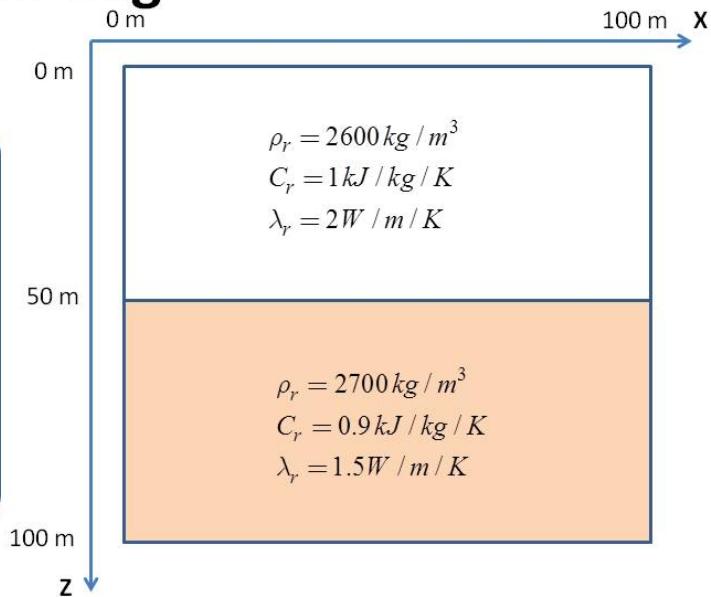
Exercise: Specify this region of the reservoir inactive and re-simulate Scenario 3

Day 2. Operations on arrays, Regions, Boundary conditions & Point sources

19

ROCKNUM flag

Exercise: Specify
non uniform
distribution of
rock material
properties

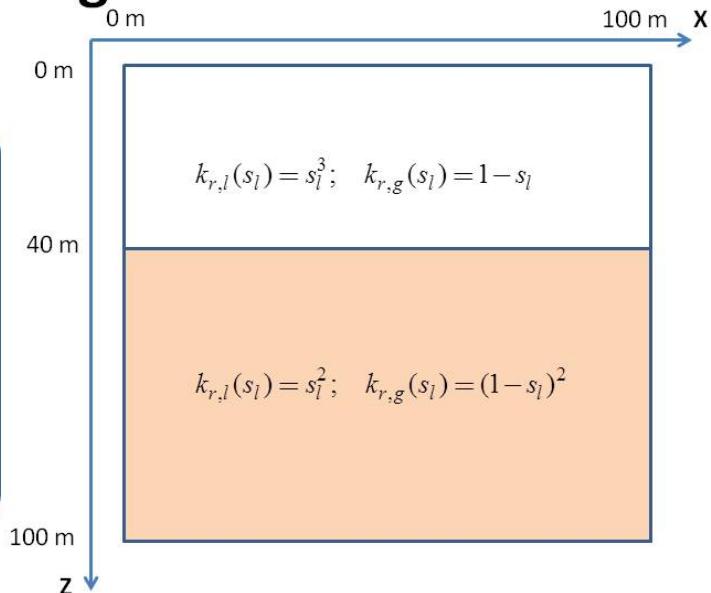


Day 2. Operations on arrays, Regions,
Boundary conditions & Point sources

20

SATNUM flag

Exercise: Specify
different
saturation
functions in the
regions



Day 2. Operations on arrays, Regions,
Boundary conditions & Point sources

21

Keywords for operations on arrays in a region of grid blocks

Keyword	Result
ADDREG	Add
COPYREG	Copy from one array into another array
EQUALREG	Equate to
MAXVAREG	Apply maximum limit
MINVAREG	Apply minimum limit
MULTIREG	Multiply by
OPERAREG	Apply a complicated arithmetic operation

Day 2. Operations on arrays, Regions,
Boundary conditions & Point sources

22

Keyword EQUALREG

```

EQUALREG syntax
1 -- in all sections except RUNSPEC and POST
2
3 EQUALREG
4 mnemonic1 value1 region1 regionID1 /
5 mnemonic2 value2 region2 regionID2 /
6 mnemonic3 value3 region3 regionID3 /
7 ...
8 /
9 =====
10
11 mnemonic# - mnemonic of the property which is modified;
12 value#   - value assigned to the property in the region;
13 region#  - mnemonic of the region in which the property is modified;
14 regionID# - region number.
15 =====
16
17
18 The keyword results in the following:
19
20
21 mnemonic1:=value1 in the region region1=regionID1
22 mnemonic2:=value2 in the region region2=regionID2
23 mnemonic3:=value3 in the region region3=regionID3
24 ...

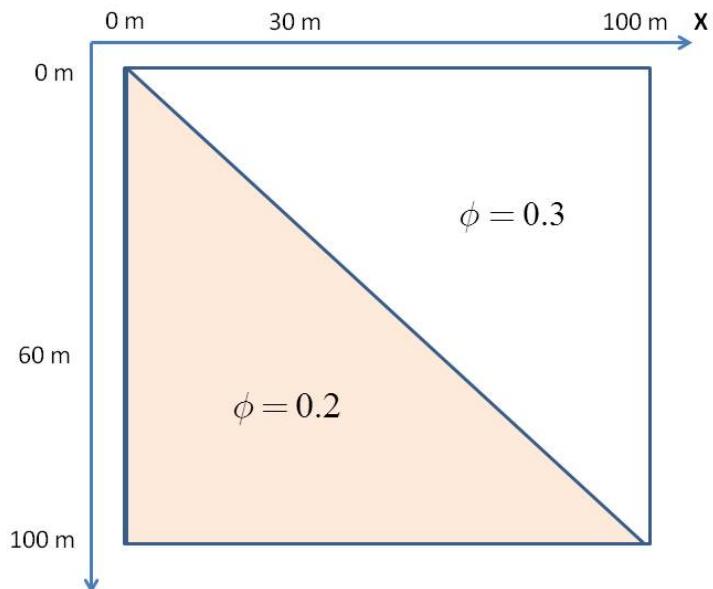
```

Day 2. Operations on arrays, Regions,
Boundary conditions & Point sources

23

Exercise

Exercise: Specify the following porosity distribution and re-simulate scenario 3.



Day 2. Operations on arrays, Regions, Boundary conditions & Point sources

24

Answer

```
1 -- in GRID section
2
3 OPERATE
4   AUXARR1 XCOORD COPY      /
5   AUXARR1 1* MULTIPLY -1 /
6   AUXARR1 DEPTH ADDARR /
7   INCONUM AUXARR1 SETINT 1 0 10000 /
8 /
9 EQUALREG
10  PORO 0.3 INCONUM 0 /
11  PORO 0.2 INCONUM 1 /
12 /
13 /
```

Day 2. Operations on arrays, Regions, Boundary conditions & Point sources

25

3.3. Boundary conditions

Boundary conditions

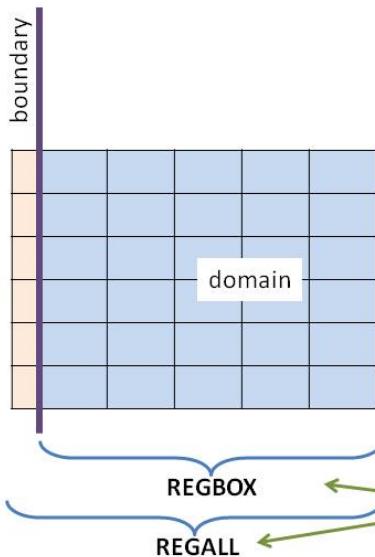
Day 2. Operations on arrays, Regions,
Boundary conditions & Point sources

26

Dirichlet boundary conditions

Additional grid blocks
(in which ACTNUM=2)
created by the
BOUNDARY keyword

The additional blocks
can be referred to by
the **FLUXNUM** number



Day 2. Operations on arrays, Regions,
Boundary conditions & Point sources

27

BOUNDARY keyword

```

1 -- within MAKE-ENDMAKE brackets
2
3 BOUNDARY
4   fluxnum1 imini1 imax1 jmini1 jmax1 kmini1 kmax1 d1_1 d2_1 d3_1 d4_1 d5_1 d6_1
5   type_1 mode_1 nui_1 nu2_1 nu3_1 typenum1 actnum1 /
6   fluxnum2 imin2 imax2 jmin2 jmax2 kmin2 kmax2 d1_2 d2_2 d3_2 d4_2 d5_2 d6_2
7   type_2 mode_2 nui_2 nu2_2 nu3_2 typenum2 actnum2 /
8   fluxnum3 imin3 imax3 jmin3 jmax3 kmin3 kmax3 d1_3 d2_3 d3_3 d4_3 d5_3 d6_3
9   type_3 mode_3 nui_3 nu2_3 nu3_3 typenum3 actnum3 /
10 ...
11 /
12 =====
13
14
15 fluxnum#      - FLUXNUM region number assigned to created grid blocks;
16 imin#-imax#    - the boundaries of the input box along i-indexation axis.
17 By default these values are equal to '1' and the 2nd
18 argument of the keyword MAKE, respectively;
19 jmin#-jmax#    - the boundaries of the input box along j-indexation axis.
20 By default these values are equal to '1' and the
21 argument of the keyword MAKE, respectively.

```

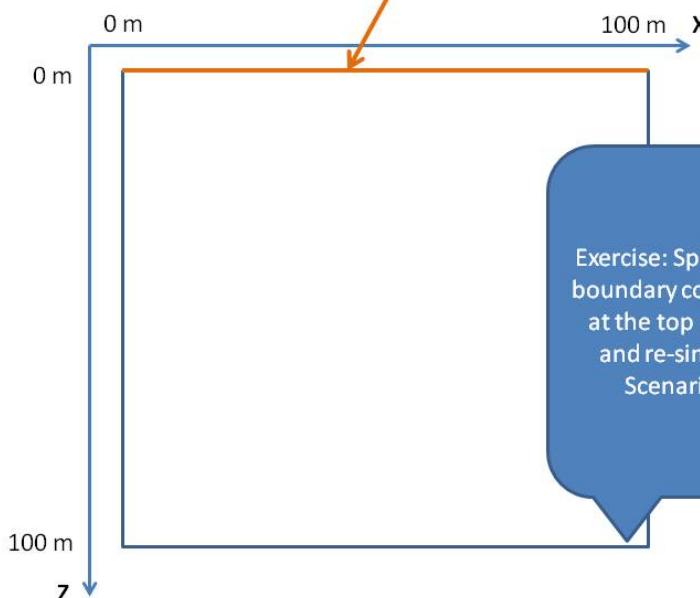
See full description
in the reference
manual

Day 2. Operations on arrays, Regions,
Boundary conditions & Point sources

28

Exercise

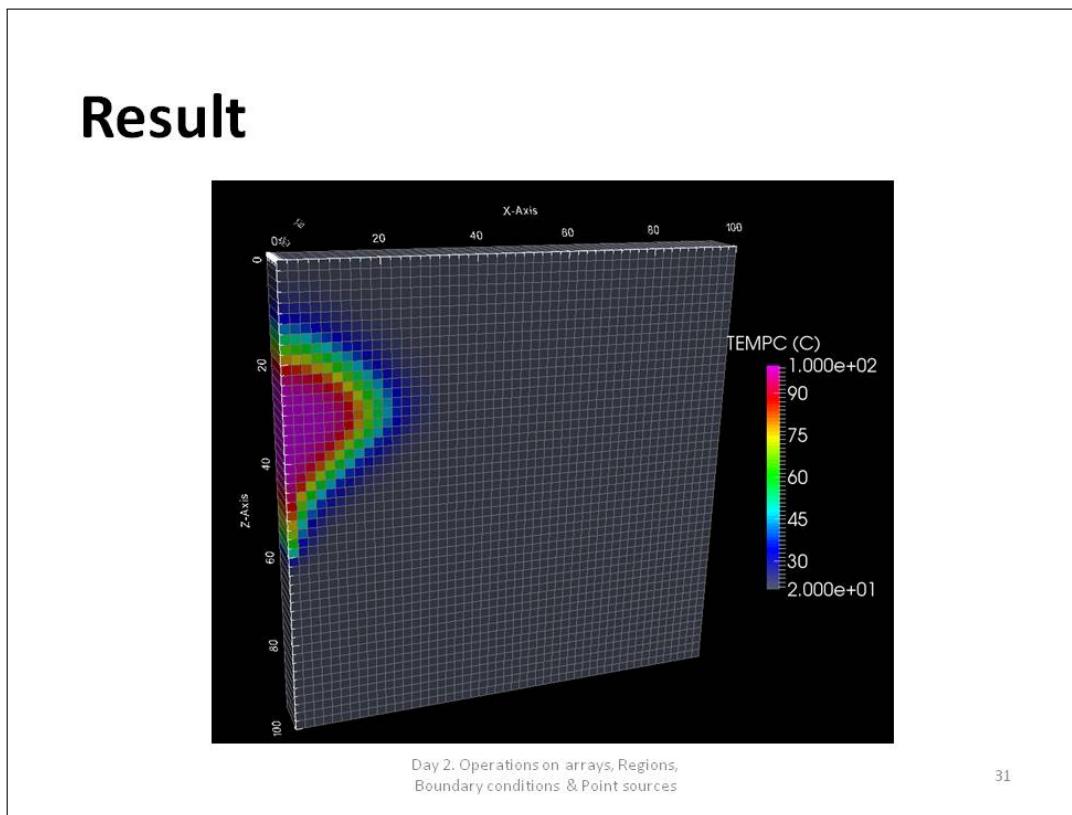
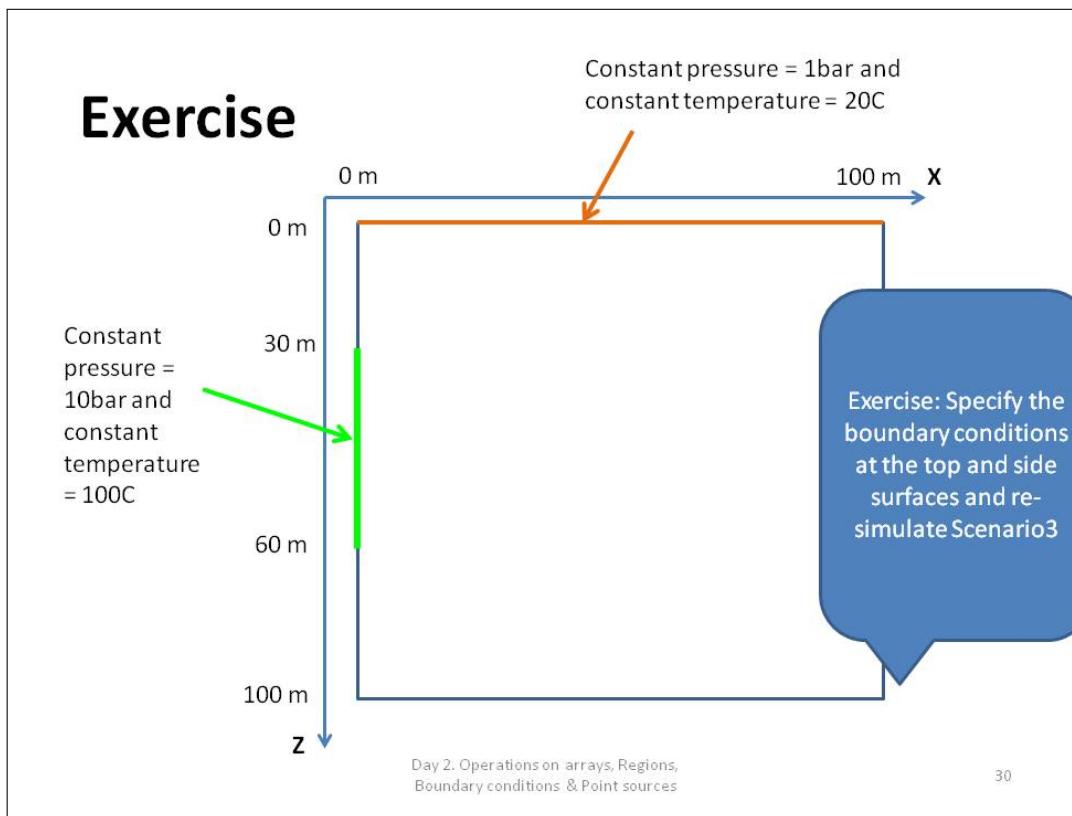
Constant pressure = 1bar and
constant temperature = 20C

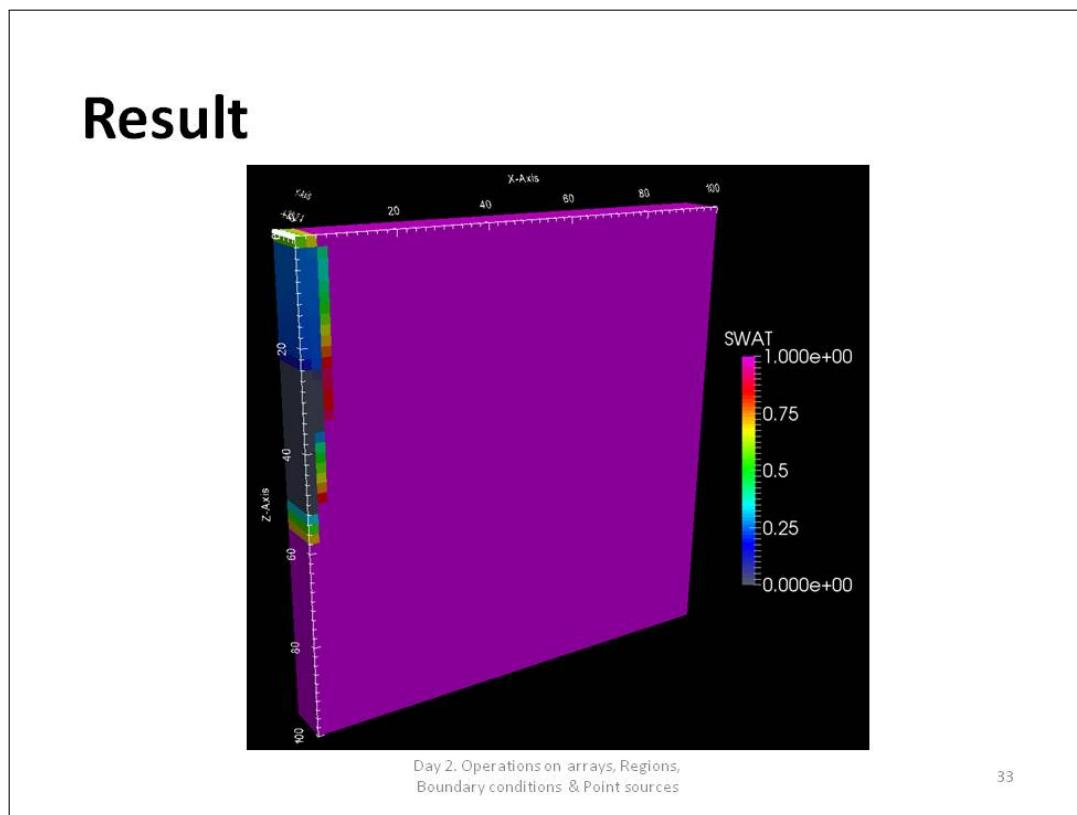
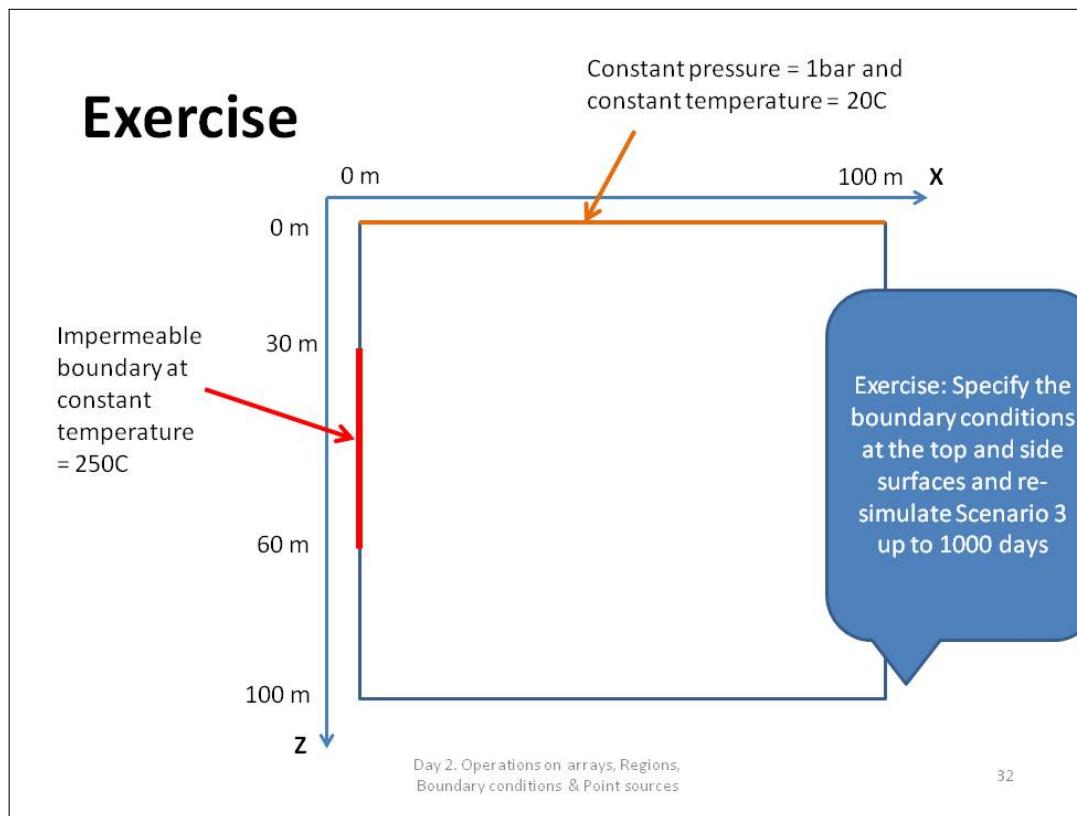


Exercise: Specify the
boundary conditions
at the top surface
and re-simulate
Scenario 3.

Day 2. Operations on arrays, Regions,
Boundary conditions & Point sources

29





3.4. Point sources

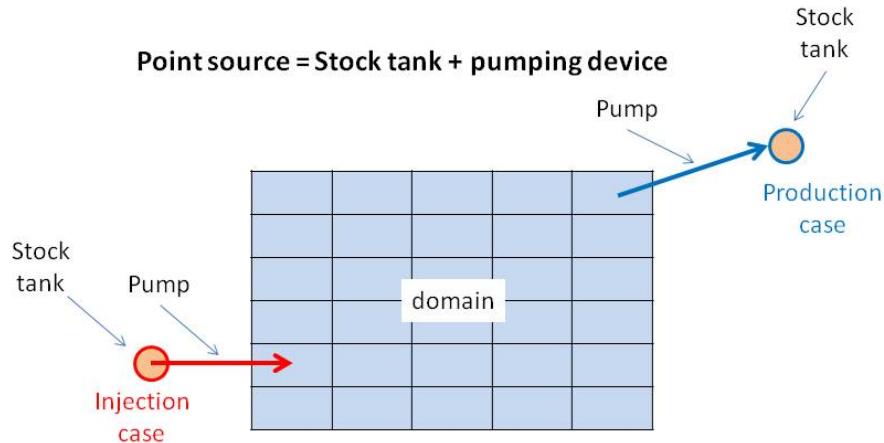
Point sources

Day 2. Operations on arrays, Regions,
Boundary conditions & Point sources

34

Point sources

Point source = Stock tank + pumping device



The parameters of injected fluid are defined in stock tank.

The injection rate is defined in pump properties.

You can refer to both stock tank and pumping device by using the name of point source.

The point source name is 8-byte character.

Day 2. Operations on arrays, Regions,
Boundary conditions & Point sources

35

Keywords for operations on arrays for ‘named’ cells

Using these keywords you can define parameters of fluid in stock tanks.

Keyword	Result
ADDNAM	Add
COPYNAM	Copy from one array into another array
EQUALNAM	Equate to
MAXVANAM	Apply maximum limit
MINVANAM	Apply minimum limit
MULTINAM	Multiply by
OPERANAM	Apply a complicated arithmetic operation

Day 2. Operations on arrays, Regions,
Boundary conditions & Point sources

36

EQUALNAM keyword

```

EQUALNAM syntax
1 -- in all sections except RUNSPEC and POST
2
3 EQUALNAM
4   mnemonic1 value1 template1 /
5   mnemonic2 value2 template2 /
6   mnemonic3 value3 template3 /
7   ...
8 /
9
10 =====
11
12   mnemonic# - mnemonic of the property which is modified;
13   value# - value assigned to the property;
14   template# - character name template.
15
16 =====
17
18   The keyword results in the following:
19
20     mnemonic1:=value1 for all cells which character name (if it
21                           is assigned) belong to template1;
22     mnemonic2:=value2 for all cells which character name (if it
23                           is assigned) belong to template2;
24     mnemonic3:=value3 for all cells which character name (if it
25                           is assigned) belong to template3.
26   ...

```

37

SRCSPEC keyword

This keyword defines the location of the point source.

```
-- within MAKE-ENDMAKE brackets
SRCSPEC
  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).
=====
```

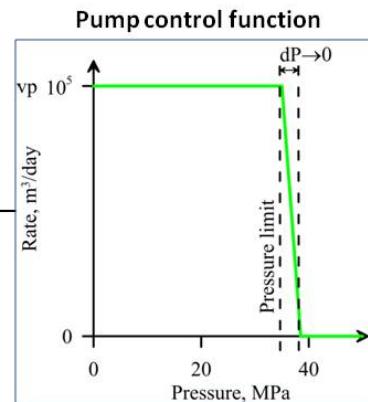
Day 2. Operations on arrays, Regions,
Boundary conditions & Point sources

38

SRCINJE keyword

This keyword defines parameters of injection sources.

```
-- 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 recommended);
plim#    - maximum pressure at the pump outlet;
volrate# - volumetric rate;
massrate# - mass rate;
vp#      - maximum volumetric rate of the pumping device;
dp#      - the pressure gap for control function.
```



Day 2. Operations on arrays, Regions,
Boundary conditions & Point sources

39

Exercise

Constant pressure = 1bar and
constant temperature = 20C

Injection rate = 25 ton/day
Pressure = 1 MPa
Temperature = 100C

Exercise: Re-simulate Scenario 3 subject to the boundary conditions and the injection pointsource

Day 2. Operations on arrays, Regions, Boundary conditions & Point sources

40

SRCPROD keyword

This keyword defines parameters of production sources (sinks).

SRCPROD syntax

```

1 -- in SCHEDULE section
2
3 PUMPPROD
4   name1 targ1 injtype1 plim1 volrate1 massrate1 vp1 dp1 /
5   name2 targ2 injtype2 plim2 volrate2 massrate2 vp2 dp2 /
6   name3 targ3 injtype3 plim3 volrate3 massrate3 vp3 dp3 /
7 ...
8 /
9 =====
10
11   name#      - pump name (8-byte character);
12   targ#      - pump operational target. Available values: MASS - mass rate,
13                  RATEIN - volumetric rate on inlet, RATEOUT - volumetric rate on
14                  outlet;
15   injtype#   - fluid used for operational control (default value recommended);
16   plim#      - minimum pressure at the pump inlet;
17   volrate#   - volumetric rate;
18   massrate#  - mass rate;
19   vp#        - maximum volumetric rate of the pumping device;
20   dp#        - the pressure gap for control function.

```

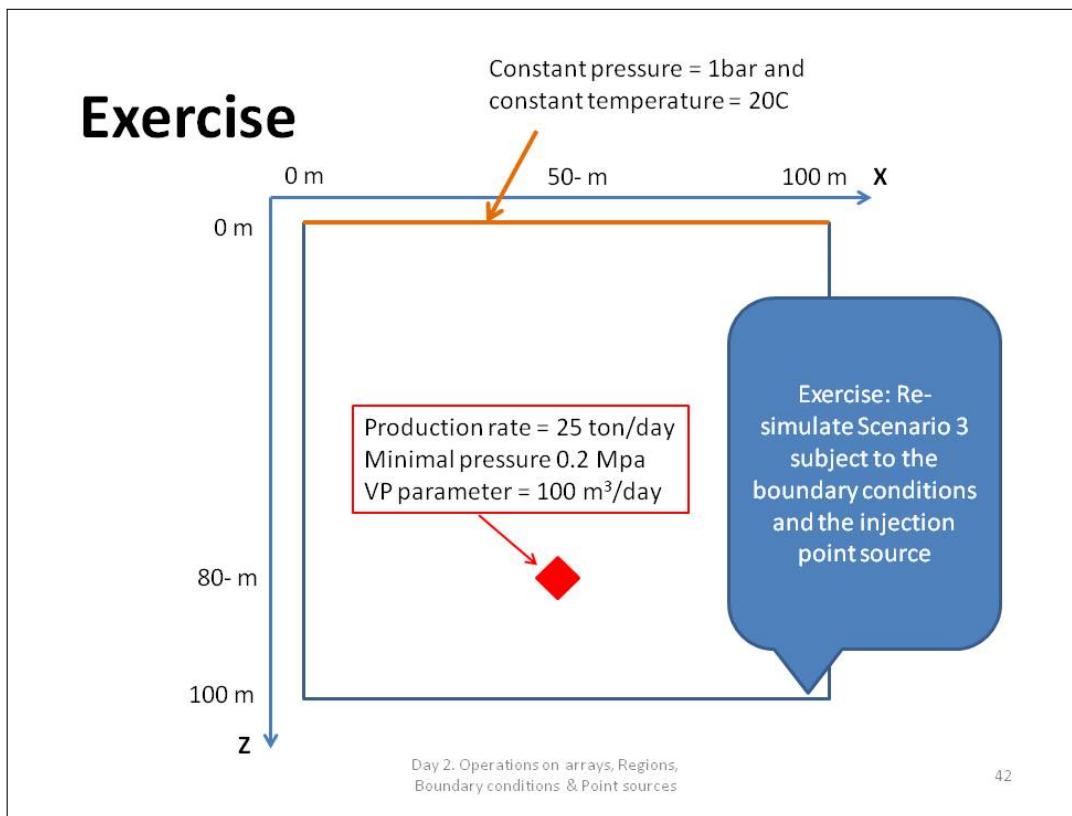
Pump control function

Rate, m^3/day

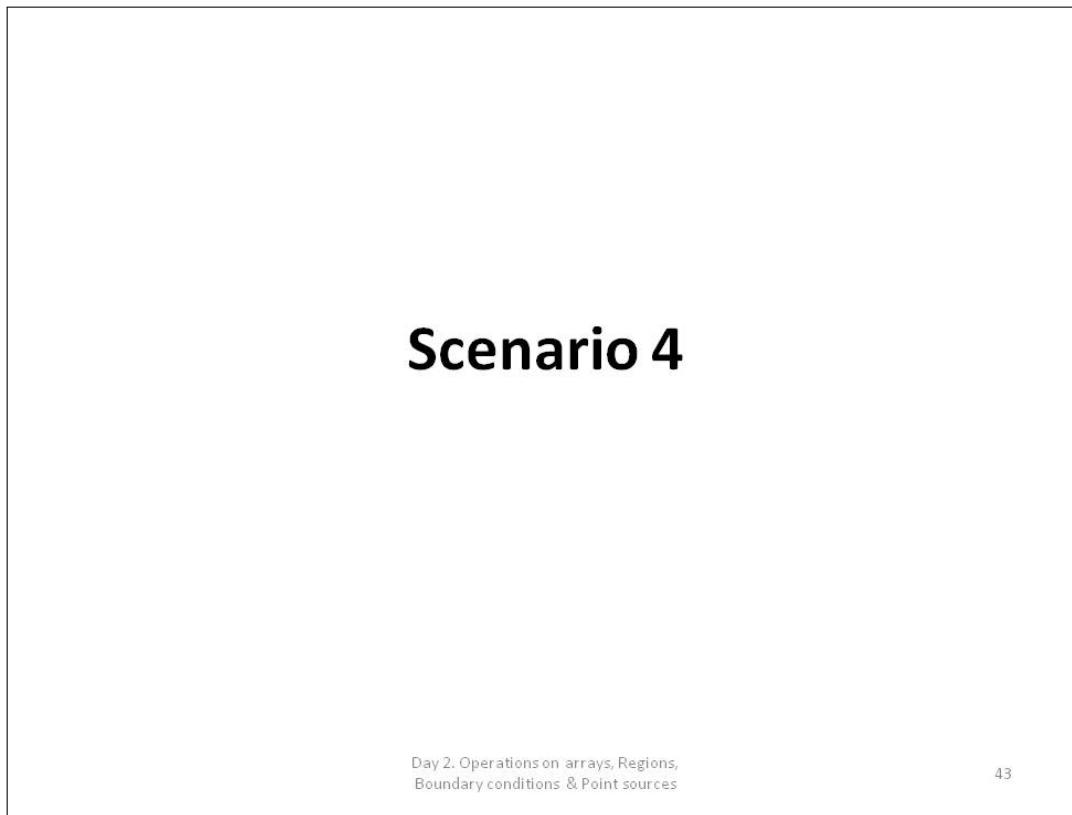
Pressure, MPa

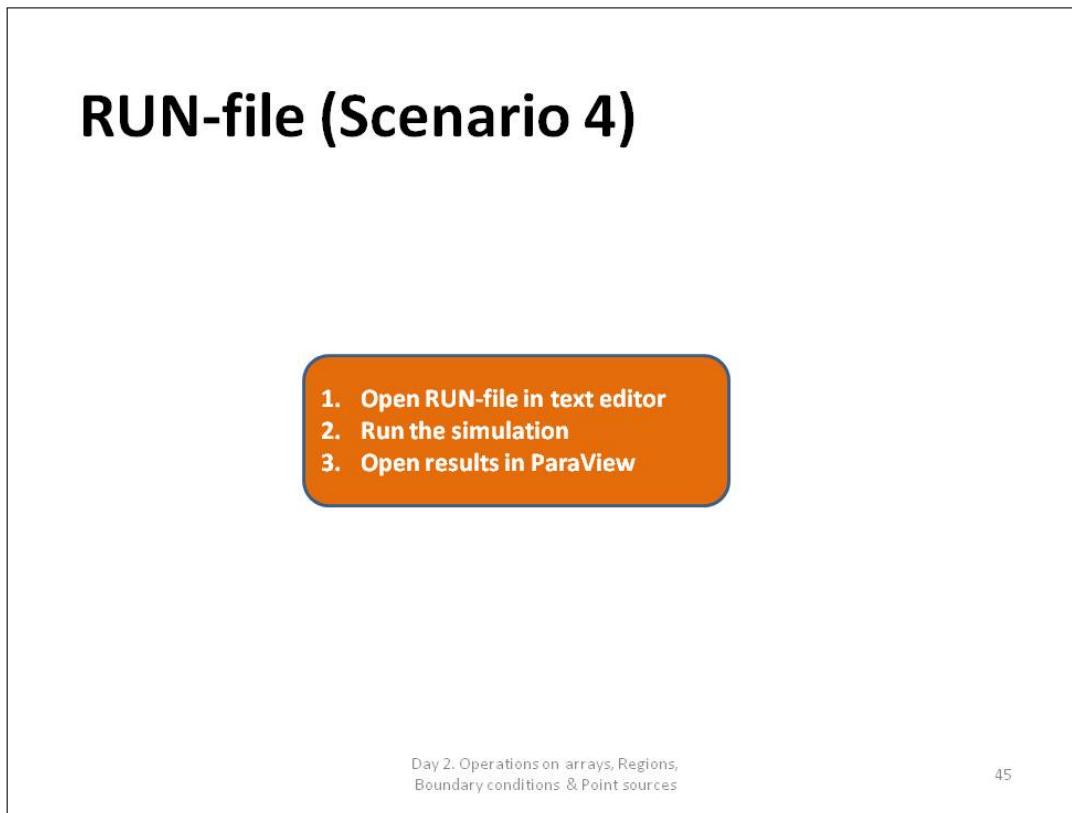
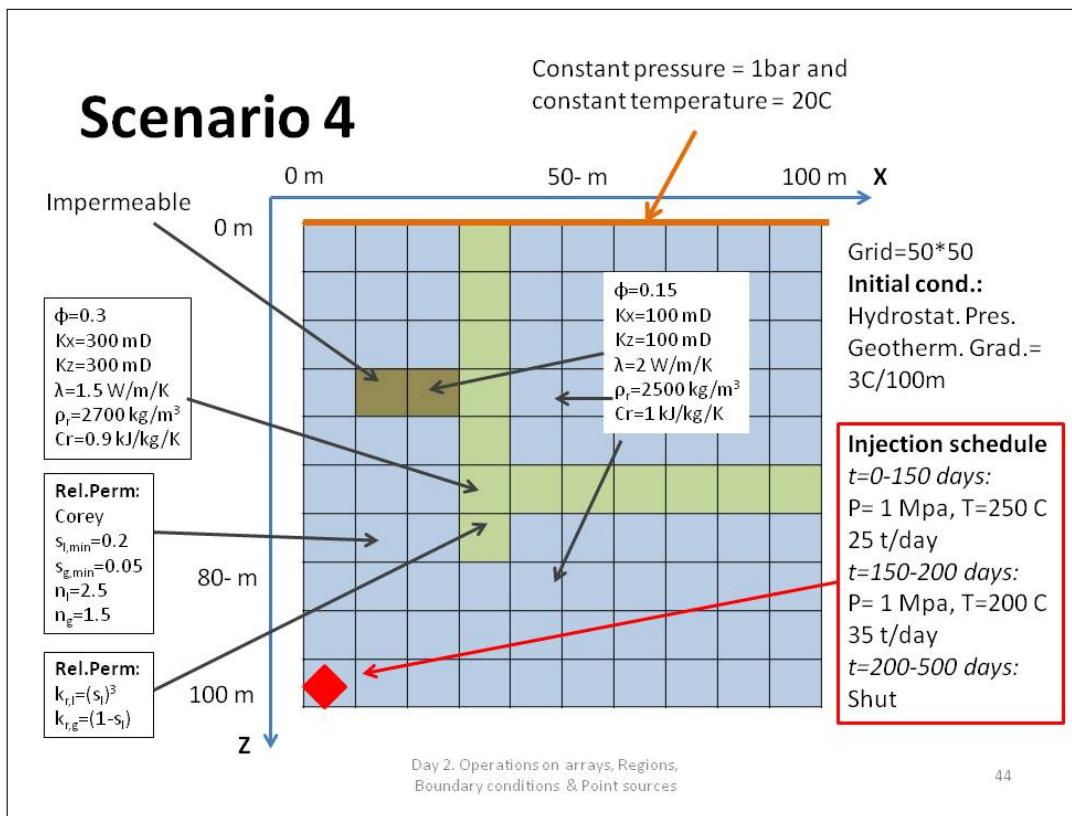
Day 2. Operations on arrays, Regions, Boundary conditions & Point sources

41



3.5. Scenario 4



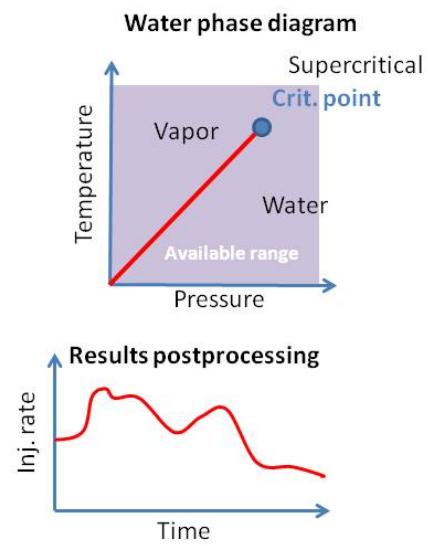
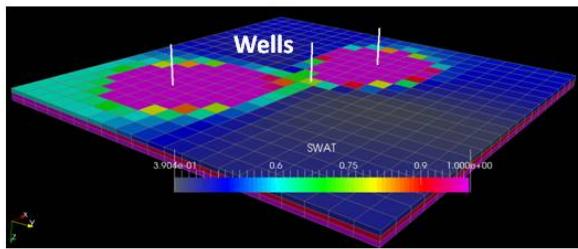


SIMULATIONS/SCENARIO4/0/SCENARIO4.RUN


```
192
193 TUNING
194 1* 2 / The maximal time step is 2 days.
195
196 ===== t=0-150days =====
197
198 SRCINJE We specify that the injection rate is
199   'MYSOURCE' MASS 1* 20. 1* 25.0 /
200   / 25 ton/day. The pressure threshold is
201
202 TSTEP Advancing simulation to 150 days.
203 15*10 /
204
205 ===== t=150-200 days =====
206
207 EQUALNAM
208   TEMPC 150. 'MYSOURCE' / We specify that the temperature is now
209   / 200C in the stock tank.
210
211 INITNAM This keyword is used to apply the
212   'MYSOURCE' / temperature definition in the stock
213
214
215 SRCINJE We increase the injection rate to
216   'MYSOURCE' MASS 1* 1* 1* 35.0 /
217   /
218
219 TSTEP Advancing simulation to 200 days.
220 5*10 /
221
222 ===== t=200-500 days =====
223
224 SRCSHUT The point source is shut down.
225   'MYSOURCE' /
226   /
227
228 TSTEP Advancing simulation to 500 days.
229 30*10 /
230
231 POST ##### POST section begins here #####
232
233 CONVERT Converting output to ParaView format.
234
235 END #####
```

Next day

- EOS-module BINMIXT
- Section POST
- Wells



Day 2. Operations on arrays, Regions,
Boundary conditions & Point sources

46

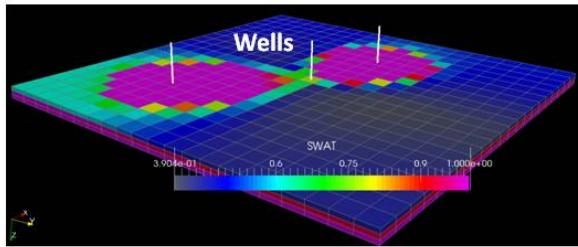
4 Day 3. Module BINMIXT, Section POST & Wells

MUFITS Training Course

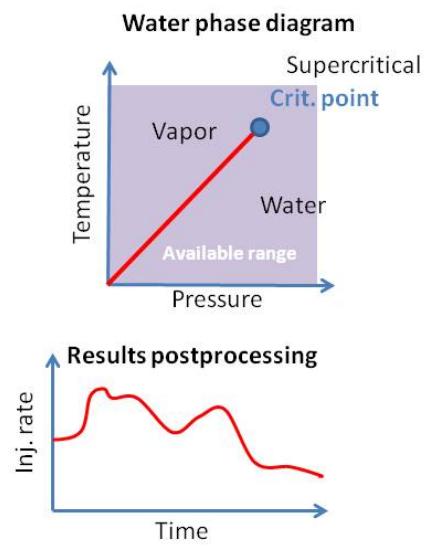
**Day 3
Module BINMIXT, Section POST
& Wells**

Program

- BINMIXT EOS module (scenario 5)
- Section POST
- Wells (scenario 6)



Day 3. Module BINMIXT, Section POST & Wells



2

4.1. BINMIXT EOS module (Scenario 5)

BINMIXT EOS module

Day 3. Module BINMIXT, Section POST & Wells

3

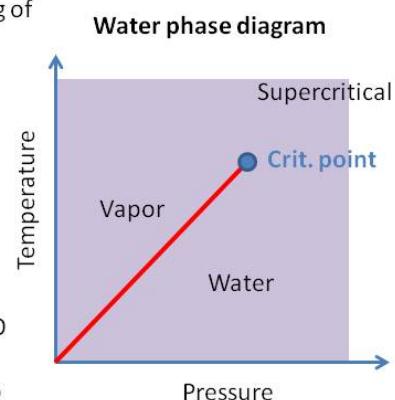
EOS module description

BINMIXT module is designed for nonisothermal modelling of multiphase flows of binary mixture in a wide range of pressures and temperatures and under near critical thermodynamic conditions.

The module is capable of modelling three-phase flows of binary mixture.

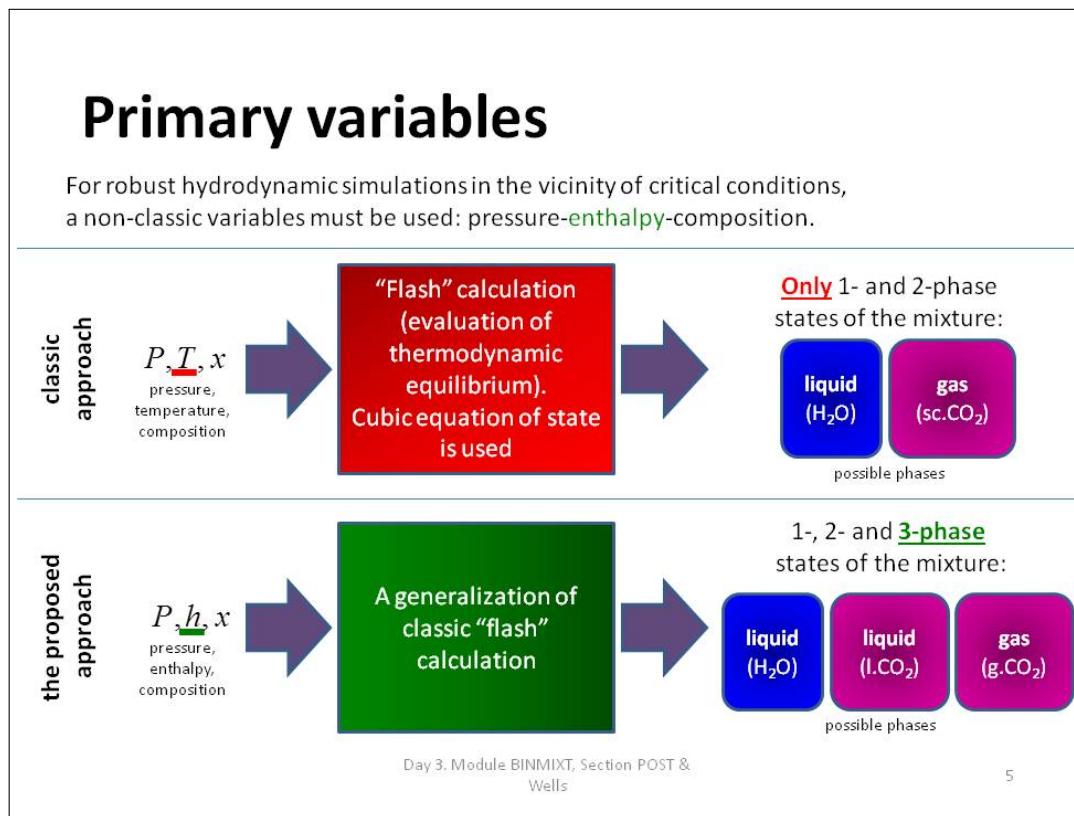
BINMIXT is the default module of the simulator.

This course covers application of the module for CO₂-H₂O flows. The available range of pressures is <0.01 bars to >1500 bars. The available range of temperatures is 0 °C to >900 °C.



Day 3. Module BINMIXT, Section POST & Wells

4



Mathematical model

Balance equation

$$\frac{\partial}{\partial t} \left(m \sum_{i=1}^3 \frac{M_{(j)} x_{i(j)}}{v_i} s_i \right) + \text{div} \left(\sum_{i=1}^3 \frac{M_{(j)} x_{i(j)}}{v_i} \mathbf{w}_i \right) = 0, \quad j = 1, 2$$

$$\frac{\partial}{\partial t} \left(m \sum_{i=1}^3 \frac{e_i}{v_i} s_i + (1-m) \frac{e_s}{v_s} \right) + \text{div} \left(\sum_{i=1}^3 \frac{h_i}{v_i} \mathbf{w}_i - \lambda \mathbf{grad} T \right) = 0$$

$$\mathbf{w}_i = -K \frac{f_i}{\mu_i} \left(\mathbf{grad} P - \frac{M_i}{v_i} \mathbf{g} \right) \quad \text{- Darcy correlation}$$

Equations are in molar variables

Equations for prediction

$$\sigma_t = \sum_{i=1}^3 \sigma_i V_i \rightarrow \max,$$

$$\sigma_i = \sigma(P, h_i, x_i), \quad \sum_{i=1}^3 V_i = 1$$

$$\sum_{i=1}^3 h_i V_i = h_t, \quad \sum_{i=1}^3 x_i V_i = x_t$$

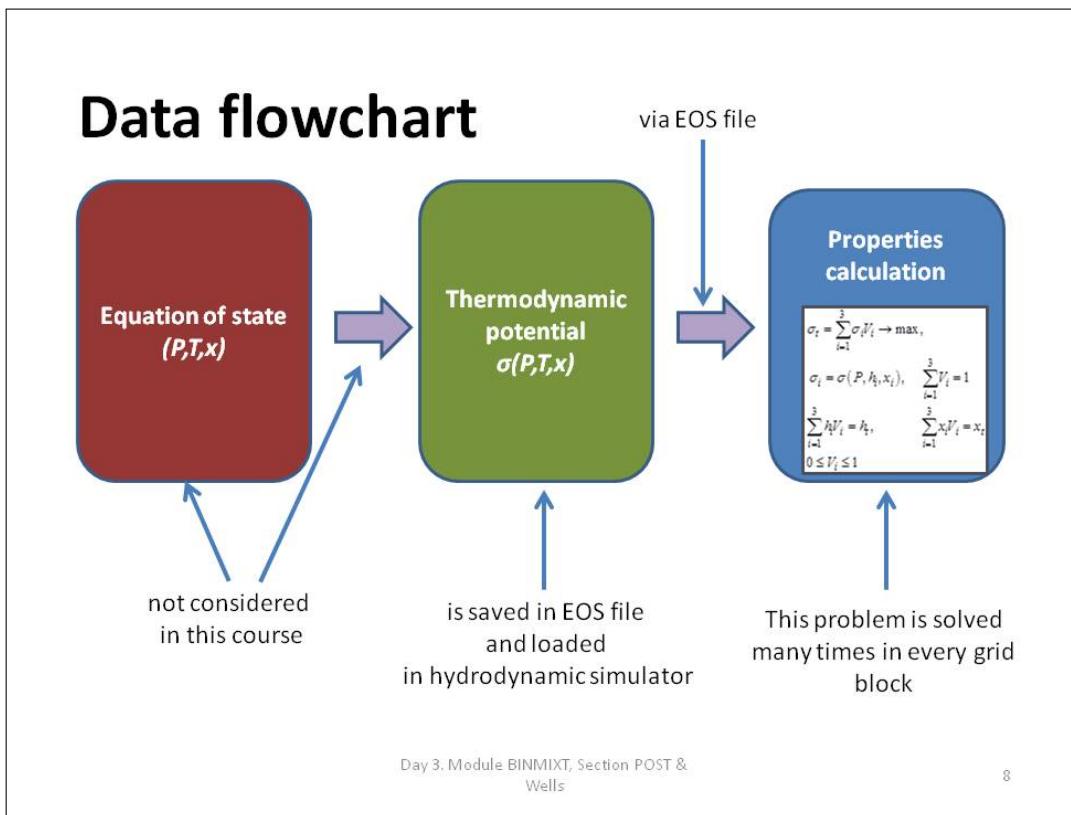
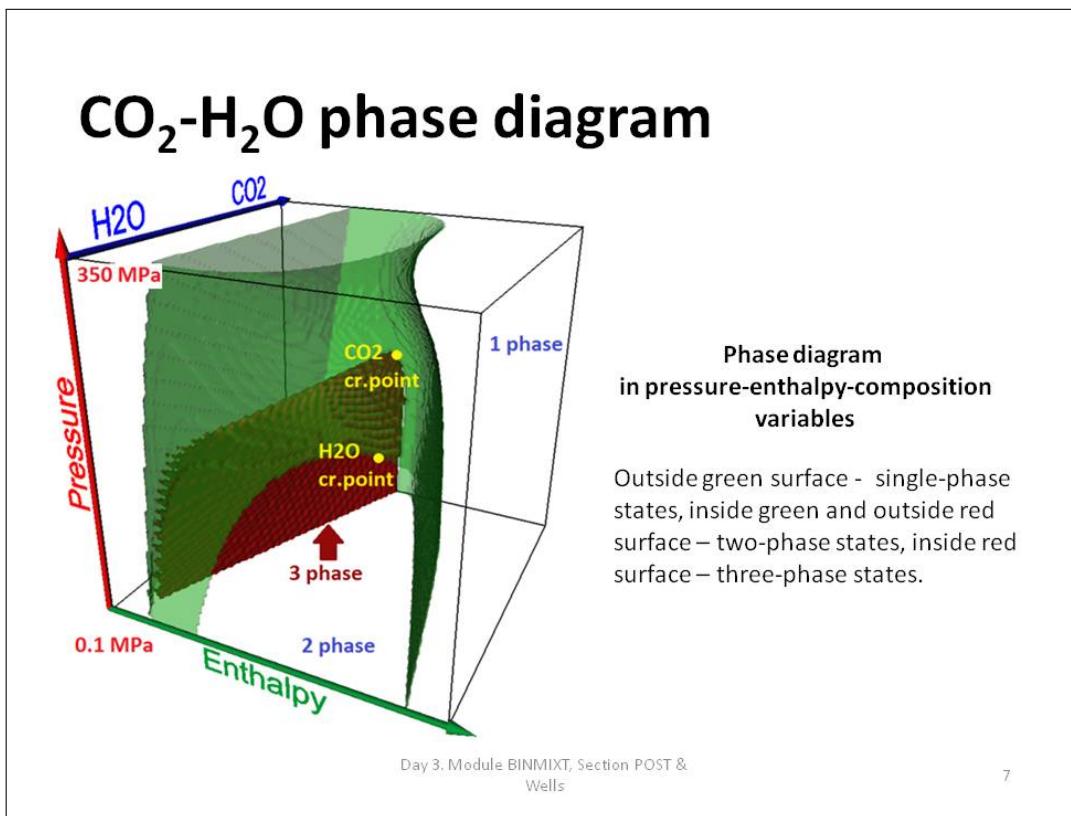
$$0 \leq V_i \leq 1$$

$j = 1 \quad - \quad CO_2$
 $j = 2 \quad - \quad H_2O$

P – pressure, h – enthalpy
 x – composition, σ – entropy

Day 3. Module BINMIXT, Section POST & Wells

6



LOADEOS keyword

The EOS-file contains the thermodynamic potential of the mixture which was calculated in advance. The file should be copied in INCLUDE folder. The file must be loaded by the **LOADEOS** keyword in **PROPS** section.

```

1 -- in PROPS section
2
3 LOADEOS
4   filename /
5
6 =====
7
8   filename - name of EOS file.

```

Additional mnemonics

Mnemonic	Description
ENTHT	Total molar enthalpy (kJ/mol)
COMP1T	Total molar fraction of the 1 st component (CO ₂)
COMP2T	Total molar fraction of the 2 nd component (H ₂ O)
SGASINIT	Initial gas saturation
SLIQINIT	Initial liquid saturation

Initial conditions

Initial conditions options:

- a) Pressure (PRES), Temperature (TEMP or TEMPC), Composition (COMP1T or COMP2T) [priority 1];
- b) Pressure (PRES), Enthalpy (ENTHT), Composition (COMP1T or COMP2T) [priority 2].

For two-phase states defined by a) and b) the saturations of phases can be redefined by the SLIQINIT and SGASINIT mnemonics. Note, that in this case, the total enthalpy (ENTHT) and total compositions (COMP1T or COMP2T) are altered to satisfy the conditions for saturation, however the thermophysical properties of each phase (density, viscosity, etc.) are not altered.

Day 3. Module BINMIXT, Section POST & Wells

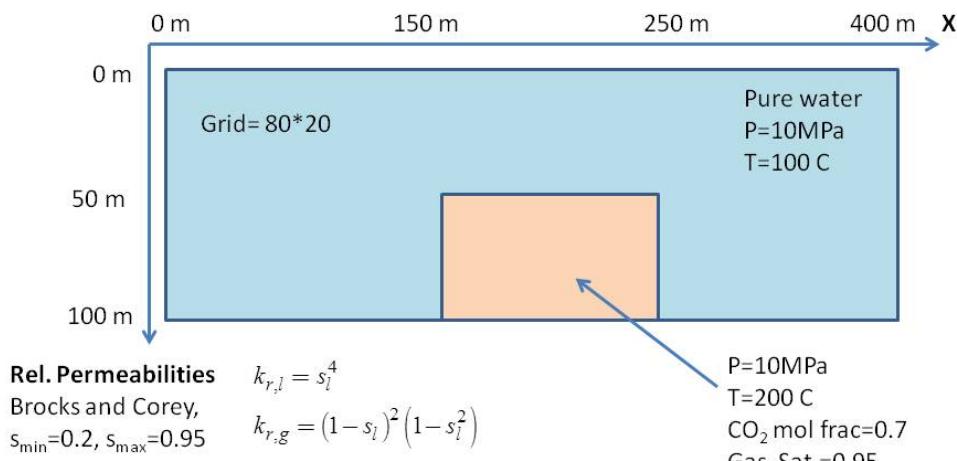
11

Scenario 5

Simulate scenario up to 200 days reporting solution every 10 days.

Rock properties:

Porosity = 0.25;
Permeability = 100 mD ;
Rock density = 2900 kg/m³;
Heat capacity = 1 kJ/kg/K;
Heat conduct. = 2 W/m/K.



Day 3. Module BINMIXT, Section POST & Wells

12

RUN-file (scenario 5)

1. Open RUN-file in text editor
 2. Run the simulation
 3. Open results in ParaView

Day 3, Module BINMIXT, Section POST & Wells

13


```

73 EQUALS
74 PRES      10   /
75 TEMPC     100  /
76 COMP1T    0.0   /
77 -----
78 TEMPC     200  31 50  2* 11 20  /
79 COMP1T    0.7   /
80 SGASINIT  0.95  /
81 /
82
83 RPTSUM
84 PRES TEMPC COMP1T COMP2T PHST /
85
86 SCHEDULE ##### SCHEDULE section begins here #####
87
88 TSTEP
89 20*10 /
90
91 POST ##### POST section begins here #####
92
93 CONVERT
94
95 END #####

```

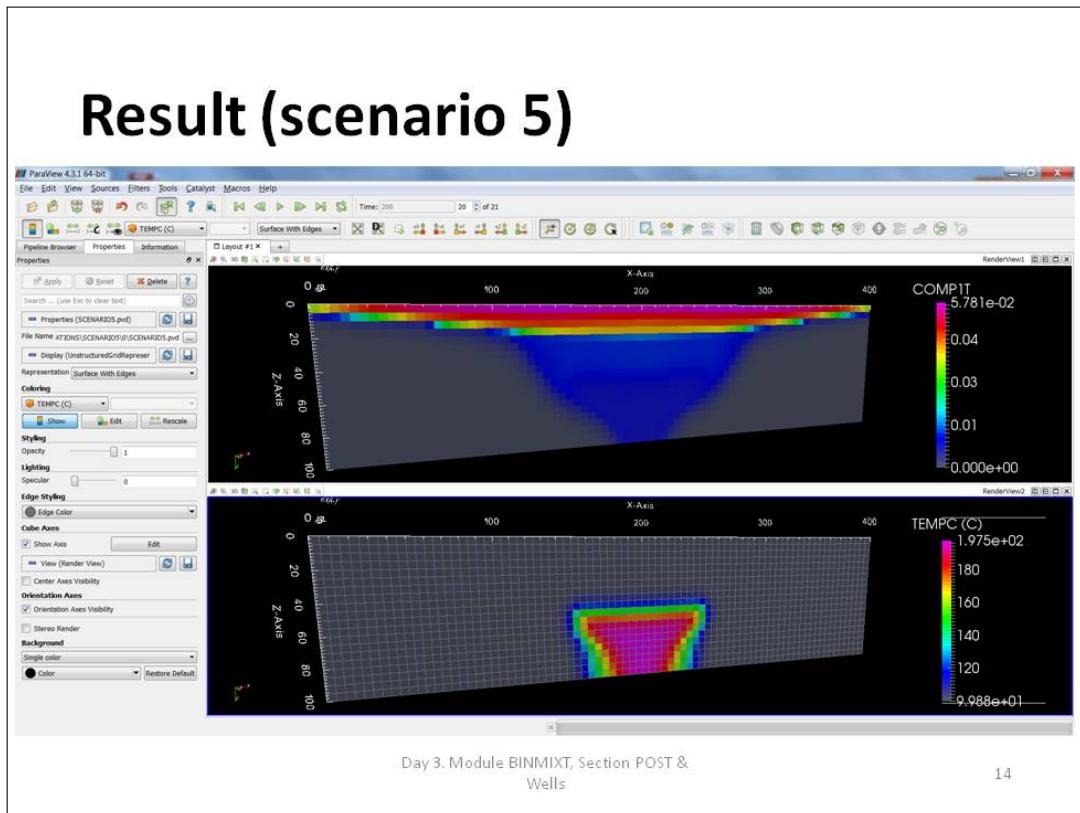
We equate in the whole domain
 pressure to 10 MPa
 temperature to 100 C
 CO_2 mol.fraction to 0.0

We equate in the hot region
 temperature to 200 C
 CO_2 mol.fraction to 0.7
 gas saturation to 0.95

We define the properties saved in SUM files

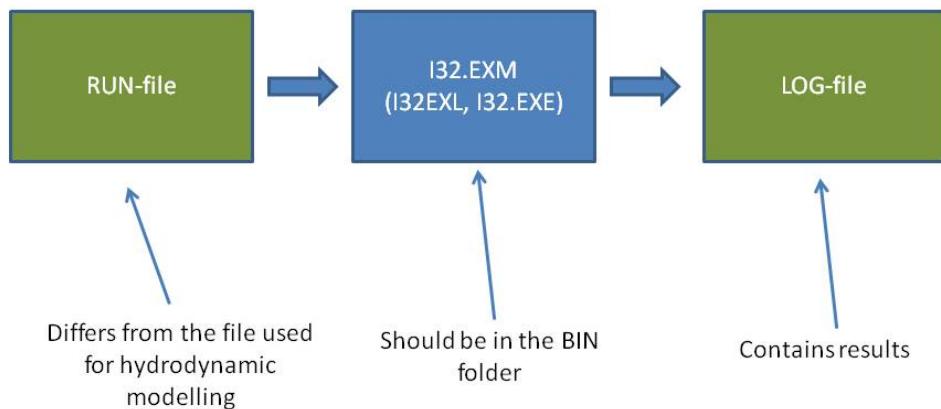
We report solution every 10 days up to 200 days.

Converting output to ParaView format.



PVT program

PVT-program allows to calculate properties of the binary mixture for a given parameters (e.g., pressure, temperature, etc.). The calculation reports are outputted in LOG-file.



Day 3. Module BINMIXT, Section POST & Wells

15

Properties for a given P and T

The properties for a given pressure, temperature and composition can be calculated by the **PHEQPTX** keyword.

```

PHEQPTX          PHEQPTX syntax
1 PHEQPTX
2 pres1 temp1 compit1 comp2t1 /
3 pres2 temp2 compit2 comp2t2 /
4 pres3 temp3 compit3 comp2t3 /
5 ...
6 /
7 =====
8
9 pres# - pressure;
10 temp# - temperature (degrees of Kelvin);
11 compit# - total molar fraction of the 1st component;
12 comp2t# - total molar fraction of the 2nd component.
13

```

Day 3. Module BINMIXT, Section POST & Wells

16

PVT program (exercise)

Exercise: Calculate thermodynamic equilibria for initial conditions in Scenario 5:

- 1) PRES=10MPa, TEMPC=100C, COMP1T=0
- 2) PRES=10MPa, TEMPC=200C, COMP1T=0.7

1. Open RUN-file in text editor
2. Run the simulation
3. Open LOG-file to see results

Day 3. Module BINMIXT, Section POST & Wells

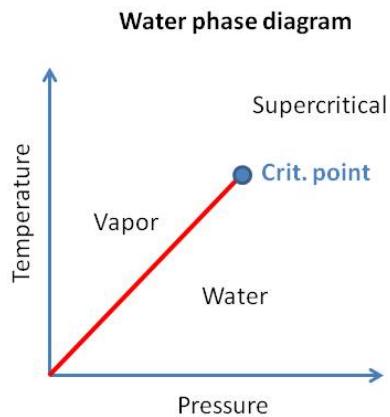
17

```
1 LOADEOS                               SIMULATIONS/PVT/0/PVT.RUN
2   '.../INCLUDE/CO2H2O_V3.0.EOS' /      We load EOS-file
3
4 PHEQPTX                                We calculate properties for:
5   10  373.15  0    /                   1) PRES=10MPa, TEMPC=100C, COMP1T=0
6   10  473.15  0.7 /                   2) PRES=10MPa, TEMPC=200C, COMP1T=0.7
7 /
8
9 END      #####
```

PHASES keyword

When moving around the critical point in the phase diagram there is no strict boundary between liquid phase and gaseous phase.

The PHASES keyword defines characteristic parameters of phases which can be used to output from hydrodynamic simulation the phases saturations, phases densities, viscosities etc.



Day 3. Module BINMIXT, Section POST & Wells

18

PHASES keyword

```

-- in PROPS section
1
2
3 PHASES
4   name1 pres1 entht1 comp1t1 comp2t1 /
5   name2 pres2 entht2 comp1t2 comp2t2 /
6   name3 pres3 entht3 comp1t3 comp2t3 /
7   ...
8 /
9 =====
10
11   name# - phase name (4-byte character);
12   pres# - pressure;
13   entht# - total molar enthalpy;
14   comp1t# - total molar fraction of the 1st component;
15   comp2t# - total molar fraction of the 2nd component.
```

Note, that the enthalpy for a phase definition can be calculated for a given temperature using PVT program.

The recommended parameters of phases (for H₂O subcritical conditions) are

- H₂O-rich phase: PRES= 1 MPa, ENTHHT=5 kJ/mol, COMP1T=0
- CO₂-rich phase: PRES= 10 MPa, ENTHHT=10 kJ/mol, COMP1T=1

Day 3. Module BINMIXT, Section POST & Wells

19

PHASES keyword

The PHASES keyword creates the following mnemonics:

Mnemonic	Description
SAT#name	Saturation
DEN#name	Density
VIS#name	Viscosity
...	See Reference manual

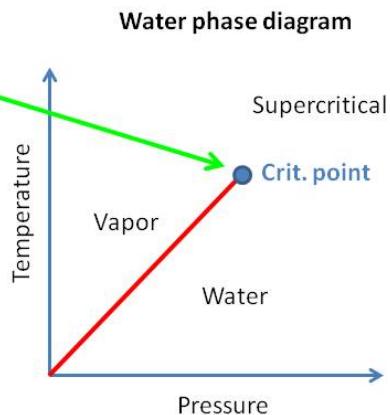
Here, 'name' is the phase name defined by the PHASES keyword.

Using BINMIXT module

Exercise: Re-simulate scenario 5 saving in the summary files
the saturations, densities and viscosities of phases.

Saturation functions scaling

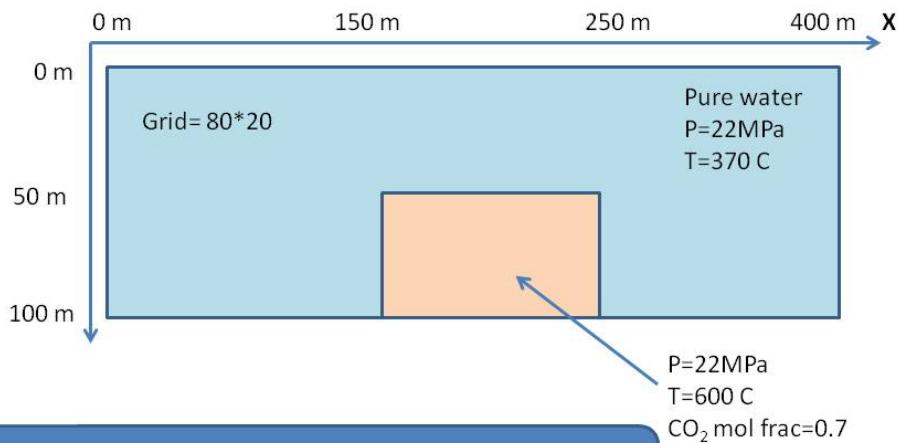
The relative permeabilities must be linear function of saturation under critical conditions. The simulator can automatically scale the saturation functions defined by the keyword SATTAB under near critical conditions. This option is enabled by the SATCRT keyword.



Day 3. Module BINMIXT, Section POST & Wells

22

Using BINMIXT module



Exercise: Re-simulate scenario 5 for new initial conditions

Day 3. Module BINMIXT, Section POST & Wells

23

4.2. POST section

POST section

Day 3. Module BINMIXT, Section POST & Wells

24

POST section

In the POST section the MUFITS output can be postprocessed to produce consolidated files, e.g., for time series data for grid blocks, point sources, wells etc.

Some of the available keywords are

Keyword	Description
POSTBLOC	Output parameters variations with time in grid blocks
POSTSRC	Output variation with time of point sources/sinks parameters
POSTWELL	Output variations with time of wells parameters

Day 3. Module BINMIXT, Section POST & Wells

25

Keyword RPTPOST

The properties saved from the POST section are specified by the RPTPOST keyword.
 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.

```
-- in POST section
1 RPTPOST
2   mnemonic1 mnemonic2 mnemonic3 ... /
3 =====
4   mnemonic# - is the mnemonic of a property saved from the POST section.
5     If one of the mnemonics is ASCII then the formatted file
6       is saved. If one of the mnemonics is ASCII then the binary
7       file is saved. If one of the mnemonics is NOTHING then the
8       output list is cleared.
9
10
11
12
```

Day 3. Module BINMIXT, Section POST &
 Wells

26

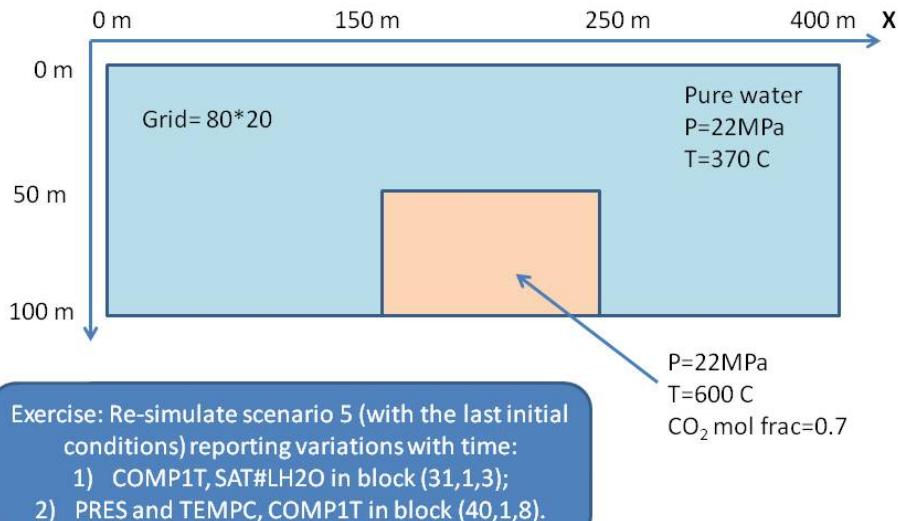
Keyword POSTBLOC

```
in POST section
1 POSTBLOC
2   i1 j1 k1 gridname1 resname1 filename1 /
3   i2 j2 k2 gridname2 resname2 filename2 /
4   i3 j3 k3 gridname3 resname3 filename3 /
5   ...
6   /
7   =====
8
9
10
11
12   i#      - i-index of the grid block;
13   j#      - j-index of the grid block;
14   k#      - k-index of the grid block;
15   gridname# - grid name (e.g., defined by the CARFIN keyword);
16   resname# - reservoir name;
17   filename# - output file name (if not specified the program uses default
18   naming convention).
```

Day 3. Module BINMIXT, Section POST &
 Wells

27

POST section (exercise)



Day 3. Module BINMIXT, Section POST & Wells

28

Keyword POSTSRC

```

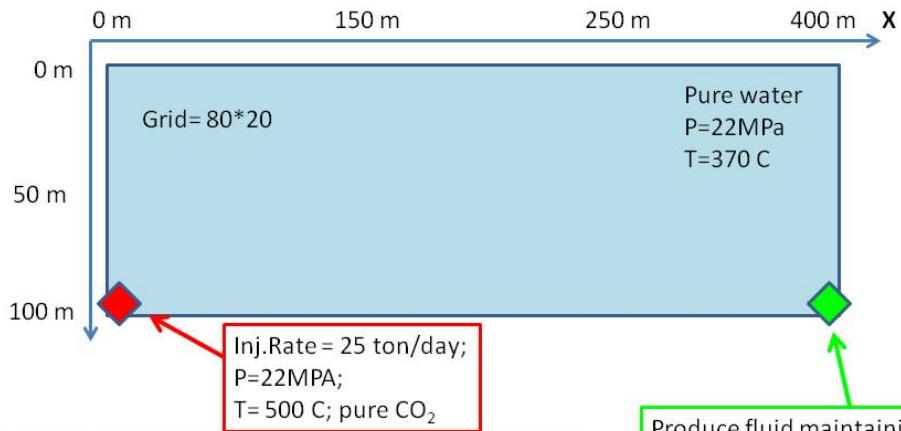
1 -- in POST section
2
3 POSTSRC
4   srcname1 filename1 /
5   srcname2 filename2 /
6   srcname3 filename3 /
7   ...
8   /
9 =====
10
11   srcname# - the point source/sink name for which the output is required;
12   filename# - output file name (if not specified the program uses default
13   naming convention).
14

```

Day 3. Module BINMIXT, Section POST & Wells

29

POST section (exercise)



Day 3. Module BINMIXT, Section POST & Wells

30

RUN-file (scenario 5; exercise)

1. Open RUN-file in text editor
2. Run the simulation
3. Open LOG-file to see results

Day 3. Module BINMIXT, Section POST & Wells

31


```

96 RPTSRC
97   SMIR#1 SMIR#2 SMIT#1 SMIT#2
98   SMPR#1 SMPR#2 SMPT#1 SMPT#2 /
99
100
101
102 SCHEDULE ##### SCHEDULE section begins here #####
103
104 SRCINJE
105   'INJECT' MASS 1* 50. 1* 25.0 /
106 /
107
108 SRCPROD
109   'PRODUCE' MASS 1* 25. 1* 1000. 1000. /
110 /
111
112 TSTEP
113   20*10 /
114
115 POST #####
116
117 CONVERT
118
119 RPTPOST
120   TIME SMIR#1 SMIR#2 SMIT#1 SMIT#2 /
121
122 POSTSRC
123   'INJECT' /
124 /
125
126 RPTPOST
127   NOTHING TIME SMPR#1
128     SMPR#2 SMPT#1 SMPT#2 /
129
130 POSTSRC
131   'PRODUCE' /
132 /
133
134 END #####

```

4.3. Wells (Scenario 6)

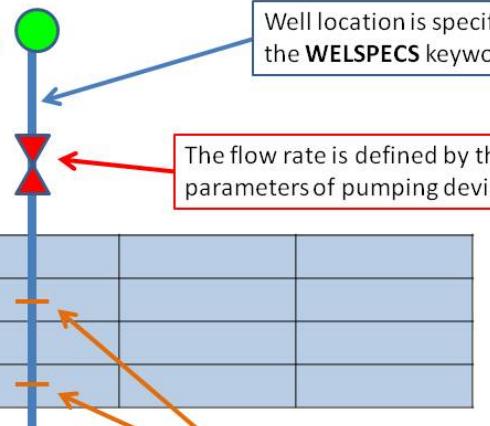
Wells

Day 3. Module BINMIXT, Section POST & Wells

32

Some notes on wells

Like in case of point sources the parameters of the injected fluid are defined in corresponding stock tank referred to by the well name.



The **WELSPECs** and **COMPDAT** keyword should be specified within **MAKE-ENDMAKE** brackets.

Well location is specified by the **WELSPECs** keyword

The flow rate is defined by the parameters of pumping device.

Well completions are specified by the **COMPDAT** keyword

Day 3. Module BINMIXT, Section POST & Wells

33

WELSPECS keyword

```

WELSPECS syntax
1 -- within MAKE-ENDMAKE brackets or in SCHEDULE section
2
3 WELSPECS
4   name1 nui iloc1 jloc1 datum1 nui r0_1 3*nui eosnum1 5*nui fluxnum1 /
5   name2 nu2 iloc2 jloc2 datum2 nui r0_2 3*nu2 eosnum2 5*nu2 fluxnum2 /
6   name3 nu3 iloc3 jloc3 datum3 nui r0_3 3*nu3 eosnum3 5*nu3 fluxnum3 /
7   ...
8 /
9
10 =====
11
12 name#      - well name;
13 nu#        - a parameter not used at present;
14 iloc#      - i-index of the grid block where the well head is located;
15 jloc#      - j-index of the grid block where the well head is located;
16 datum#    - reference depth for bottom hole pressure;
17 r0_#       - drainage radius;
18 eosnum#   - equation of state region number used for the fluid properties
              calculation in the well;
19 fluxnum#  - FLUXNUM region number assigned to all cells of which the well
              is constructed.
20
21

```

Day 3. Module BINMIXT, Section POST & Wells

34

COMPDAT keyword

```

COMPDAT syntax
1 -- within MAKE-ENDMAKE brackets or in SCHEDULE section
2
3 COMPDAT
4   name1 iloc1 jloc1 kmini1 kmax1 mode1 satnum1 tran1 d1 kh1 skin1 nui dir1 r0_1 /
5   name2 iloc2 jloc2 kmin2 kmax2 mode2 satnum2 tran2 d2 kh2 skin2 nu2 dir2 r0_2 /
6   name3 iloc3 jloc3 kmin3 kmax3 mode3 satnum3 tran3 d3 kh3 skin3 nu3 dir3 r0_3 /
7   ...
8 /
9
10 =====
11
12 name#      - well name;
13 nu#        - a parameter not used at present;
14 iloc#/jloc# - i-index and j-index of the grid block where the well is
              completed;
15 kmin#/kmax# - k-index range of grid blocks where the well is completed;
16 mode#      - if OPEN (default) the completion is opened for the fluid
              transport; if SHUT the completion is closed off;
17 satnum#    - saturation functions region (SATNUM) used for the calculation
              of the fluxes through the completion;
18 tran#      - this field is for explicit specification of the
              transmissibility;
19 d#         - wellbore diameter at the connection;
20
21

```

Day 3. Module BINMIXT, Section POST & Wells

35

WELLPROD keyword

The parameters of production wells are specified using **WELLPROD** keyword

```

1 -- in SCHEDULE section
2
3 WELLPROD
4   name1 mode1 targ1 volrate1 massrate1 bhp1 vp1 nui dimflag1 /
5   name2 mode2 targ2 volrate2 massrate2 bhp2 vp2 nu2 dimflag2 /
6   name3 mode3 targ3 volrate3 massrate3 bhp3 vp3 nu3 dimflag3 /
7   ...
8 /
9
10 =====
11
12 name#    - well name or well name template;
13 mode#    - well mode. Available values: OPEN - well open for flow (default),
14           STOP - well stoped above formation, SHUT - well completely
15           isolated form the formation.
16 targ#    - well operational target. Available values: MASS - mass rate,
17           RESV - volumetric rate at reservoir conditions, BHP - constant
18           bottom hole pressure;
19 massrate# - mass rate;
20 volrate#  - volumetric rate;
21 bhp#     - bottom-hole pressure (either limit (MASS,RESV) or target
22           parameter (BHP));

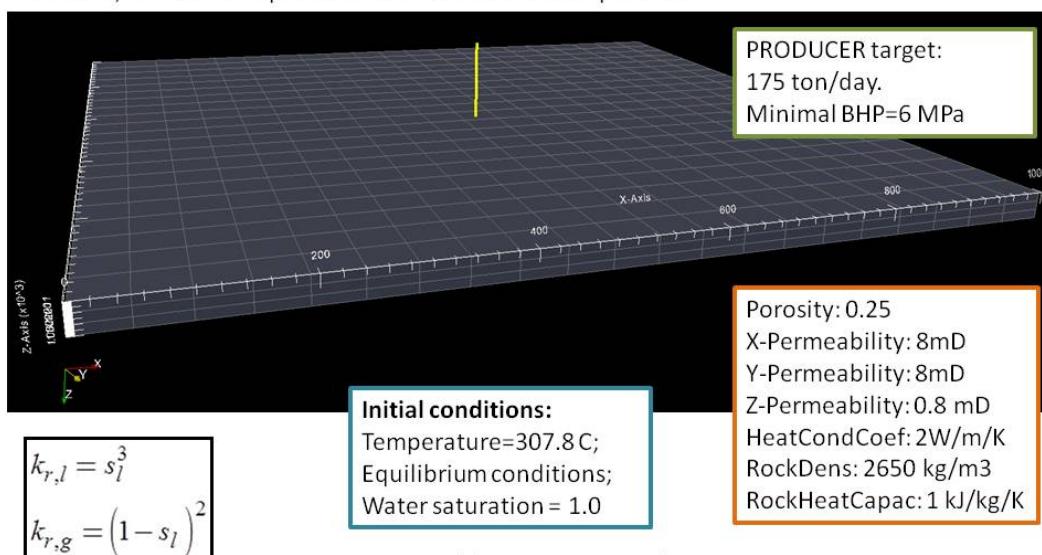
```

Day 3. Module BINMIXT, Section POST &
Wells

36

Scenario 6

Simulate production for 9540 days reporting distributions every 180 days. Create plots for well flow rate, bottom-hole pressure and bottom-hole temperature.



Day 3. Module BINMIXT, Section POST &
Wells

37

RUN-file (scenario 6)

1. Open RUN-file in text editor
 2. Run the simulation
 3. Open LOG-file to see results

Day 3, Module BINMIXT, Section POST & Wells

38


```

121 POST      ##### POST section begins here #####
122
123 CONVERT          We convert the output to ParaView
124
125
126 RPTPOST          We define the properties output from the
127   TIME WBHP WBHTC WMPR#1 WMPT#1 /
128
129 POSTWELL         We save consolidated time series
130   PRODUCER /
131 /
132
133 END      #####

```

WELLINJE keyword

The parameters of injection wells are specified using **WELLINJE** keyword

```

-- in SCHEDULE section
WELLINJE
  name1 mode1 targ1 volrate1 massrate1 bhp1 vp1 injtype1 dimflag1 /
  name2 mode2 targ2 volrate2 massrate2 bhp2 vp2 injtype2 dimflag2 /
  name3 mode3 targ3 volrate3 massrate3 bhp3 vp3 injtype3 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,
           RATE - volumetric rate at stock tank conditions, BHP - constant
           bottom hole pressure;
massrate# - mass rate;
volrate# - volumetric rate;
bhp#     - bottom-hole pressure (either limit (MASS,RATE) or target
           parameter (BHP));

```

Day 3. Module BINMIXT, Section POST &
Wells

39

Scenario 6 (exercise 1)

PRODUCER: i=11, j=11 completed in k=1,2
 INJECT1:i=6, j=6 completed in k=1,2,3
 INJECT2:i=16, j=12 completed in k=1,2,3

INJECT1 target:
85 ton/day.
Maximal BHP=12 MPa

INJECT2 target:
110 ton/day.
Maximal BHP=12 MPa

Exercise: Re-simulate scenario 6 modelling injection of cold water (25 C at 10 MPa) in the reservoir to maintain reservoir pressure. The injection starts at 3240 days. Report parameters of all wells

Day 3. Module BINMIXT, Section POST & Wells

40

RUN-file (scenario 6; exercise 1)

1. Open RUN-file in text editor
 2. Run the simulation
 3. Open LOG-file to see results

Day 3. Module BINMIXT, Section POST & Wells

41

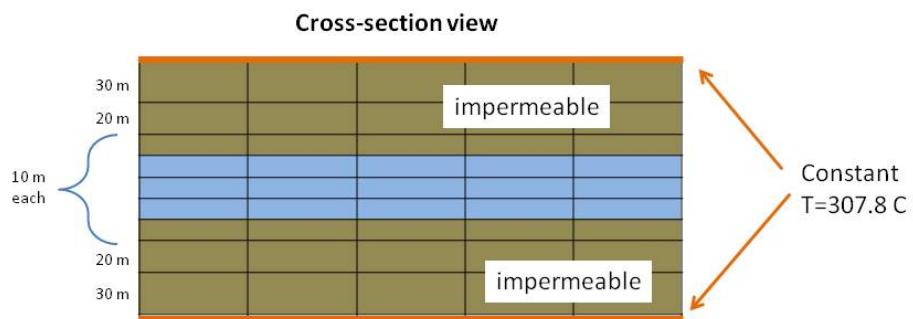

```

144 CONVERT
145
146
147 RPTPOST
148   TIME WBHP WBHTC WMPR#1 WMPT#1 /
149
150 POSTWELL
151   PRODUCER /
152 /
153
154 RPTPOST
155   NOTHING TIME WBHP WBHTC
156     WMIR#1 WMIT#1 /
157
158 POSTWELL
159   INJECT1 /
160   INJECT2 /
161 /
162
163
164 END #####
```

Scenario 6 (exercise 2)

Grid: 21*21*9

Exercise: Re-simulate scenario with 3 wells
modelling heat exchange with impermeable
overburden and underburden layers.



RUN-file (scenario 6; exercise 2)

1. Open RUN-file in text editor
 2. Run the simulation
 3. Open LOG-file to see results

Day 3, Module BINMIXT, Section POST & Wells

43


```

121   /                               as a CSV file.

122
123 INIT      ##### INIT section begins here #####
124
125 EQUALS
126   TEMPC 307.8 /
127   SWAT   1.0 /
128 /
129
130 REGALL
131 EQUALREG
132   TEMPC 307.8 FLUXNUM 111 /
133 /
134
135 EQUALNAM
136   PRES 10 'INJECT*' /
137   TEMPC 25 /
138 /
139
140 RPTSUM
141   PRES SWAT SVAP TEMPC /
142
143 RPTSUM
144   K-IJKRES /
145
146 RPTWELL
147   WBHP WBHTC WMPR#1 WMPT#1
148   WMIR#1 WMIT#1 /
149
150 SCHEDULE #### SCHEDULE section begins here #####
151
152 WELLPROD
153   PRODUCER OPEN MASS 1* 175. 6.0 /
154
155   We define that PRODUCER target is
156   175 tons/day. The minimal bottom-hole
157   pressure is 8 MPa.
158 /
159
160 TSTEP
161   1 2 3 4 10 20 20 30 90 17*180 /
162
163 WELLINJE
164   INJECT1 OPEN MASS 1* 85. 12.0 /
165   INJECT2 OPEN MASS 1* 110. 12.0 /
166 /
167
168 TSTEP

```

```

169   1 2 3 4 10 20 20 30 90 34*180 /      We advance simulation to 9540 days.
170
171 POST      ##### POST section begins here #####
172
173 CONVERT                               We convert the output to ParaView
174                                         compatible format.
175
176 RPTPOST                                We define the properties output from the
177     TIME WBHP WBHTC WMPR#1 WMPT#1 /
178
179 POSTWELL                               We save consolidated time series
180     PRODUCER /
181 /
182
183 RPTPOST                                We define the properties output from the
184     NOTHING TIME WBHP WBHTC
185             WMIR#1 WMIT#1 /
186
187 POSTWELL                               We save consolidated time series reports
188     INJECT1 /
189     INJECT2 /
190 /
191
192 END        #####

```

Scenario 6 (exercise 3)

Exercise: Re-simulate Scenario 6 (exercise 2)
using EOS-module BINMIXT.
Use the following initial conditions in the reservoir
Pressure=9.577
Temperature=307.8C
CO₂ mol.fraction=0.1

RUN-file (scenario 6; exercise 3)

1. Open RUN-file in text editor
 2. Run the simulation
 3. Open LOG-file to see results

Day 3, Module BINMIXT, Section POST & Wells

45


```
121  
122 ENDSAT >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>  
123  
124 PHASES  
125 'LIQ' 1.0 5.0 0.0 / We define two phases which would be  
126 'GAS' 1.0 60.0 0.0 / the liquid water and water vapor. This  
127 / can be checked using PVT program.  
128  
129 RPTSATTA We report the relative permeabilities  
130 / as a CSV file.  
131  
132 INIT ##### INIT section begins here #####  
133  
134 EQUALS We define that the reservoir is at  
135 PRES 9.577 / pressure 9.577 MPa  
136 TEMPC 307.8 / temperature 307.8  
137 COMP1T 0.1 / CO2 mol. fraction = 0.1  
138 /  
139  
140 REGALL  
141 EQUALREG We specify that the temperature at the  
142 TEMPC 307.8 FLUXNUM 111 / top and bottom boundaries is 307.8 C.  
143 /  
144  
145 EQUALNAM We define the parameters of the injected  
146 PRES 10 'INJECT*' / water. We specify that the water  
147 TEMPC 25 / temperature is 25 C at 10 MPa. Pure  
148 COMP1T 0.0 / water is injected.  
149  
150 /  
151 RPTSUM  
152 PRES TEMPC SAT#LIQ SAT#GAS We specify the properties saved in the  
153 COMP1T / summary files.  
154  
155  
156 RPTSUM We enable output of K-IJKRES property  
157 K-IJKRES / which can be used for thresholding  
158  
159  
160 RPTWELL We define the output in summaries  
161 WBHP WBHTC for wells. Note, the second component  
162 WMPR#1 WMPT#1 WMIR#1 WMIT#1 is water  
163 WMPR#2 WMPT#2 WMIR#2 WMIT#2 /  
164  
165  
166  
167 SCHEDULE ##### SCHEDULE section begins here #####  
168
```

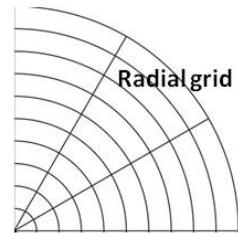
```

169 WELLPROD                                We define that PRODUCER target is
170   PRODUCER OPEN MASS 1* 175. 6.0 /      175 tons/day. The minimal bottom-hole
171   /                                         pressure is 8 MPa.
172
173 TSTEP                                     We advance simulation to 3240 days.
174   1 2 3 4 10 20 20 30 90 17*180 /
175
176 WELLINJE                                  Now we start the injection at t=3240
177   INJECT1 OPEN MASS 1* 85. 12.0 /      days. The injection rate is 85 and 110
178   INJECT2 OPEN MASS 1* 110. 12.0 /      tons/day. BHP limit is 12 MPa.
179 /
180
181 TSTEP                                     We advance simulation to 9540 days.
182   1 2 3 4 10 20 20 30 90 34*180 /
183
184 POST          ##### POST section begins here #####
185
186 CONVERT                                    We convert the output to ParaView
187                                         compatible format.
188
189 RPTPOST
190   TIME WBHP WBHTC WMPR#2 WMPT#2
191           WMPR#1 WMPT#1 /             We define the properties output from the
192                                         following POSTWELL keyword.
193
194 POSTWELL                                   We save consolidated time series
195   PRODUCER /                               report for PRODUCER.
196 /
197
198 RPTPOST
199   NOTHING TIME WBHP WBHTC
200           WMIR#2 WMIT#2 /             We define the properties output from the
201                                         following POSTWELL keyword. NOTHING entry
202                                         clears the output list.
203
204 POSTWELL                                   We save consolidated time series reports
205   INJECT1 /                               for injection wells.
206   INJECT2 /
207 /
208
209 END          #####

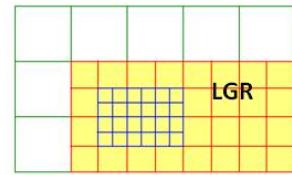
```

Next day

- Radial grids
- Fluid-in-place regions
- Local grid refinements
- Grid decomposition



$$A = \int_{\text{domain}} adV$$



*Simulation is parallel
using 1 core, 2 cores, 3
cores, ... How the
simulation is decomposed
between the cores?*

5 Day 4. Radial grids, Fluid-in-place regions, Local grid refinements & Grid decomposition

MUFITS Training Course

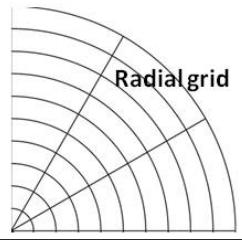
Day 4
Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

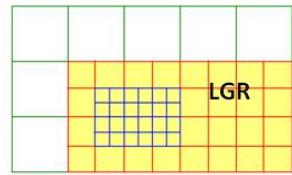
1

Program

- Radial grids (scenario 7)
- Fluid-in-place regions
- Local grid refinements (scenario 8)
- Scenario 9
- Grid decomposition



$$A = \int_{\text{domain}} adV$$



*Simulation is parallel
using 1 core, 2 cores, 3
cores, ... How the
simulation is decomposed
between the cores?*

2

Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

5.1. Radial grids (Scenario 7)

Radial grids

Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

3

Radial grids

The number of grid blocks along every axis is defined by the keyword **MAKE**

MAKE-ENDMAKE syntax

```

1 -- in GRID section
2
3 MAKE
4   gridtype ni nj nk /
5
6 -- other keywords
7
8 ENDMAKE
9
10 =====
11
12   gridtype = CART - Cartesian Grid
13   = RADIAL - Radial Grid
14   = CORNER - Corner-Point grid
15
16   ni - number of grid blocks along i-indexation axis
17   nj - number of grid blocks along j-indexation axis
18   nk - number of grid blocks along k-indexation axis

```

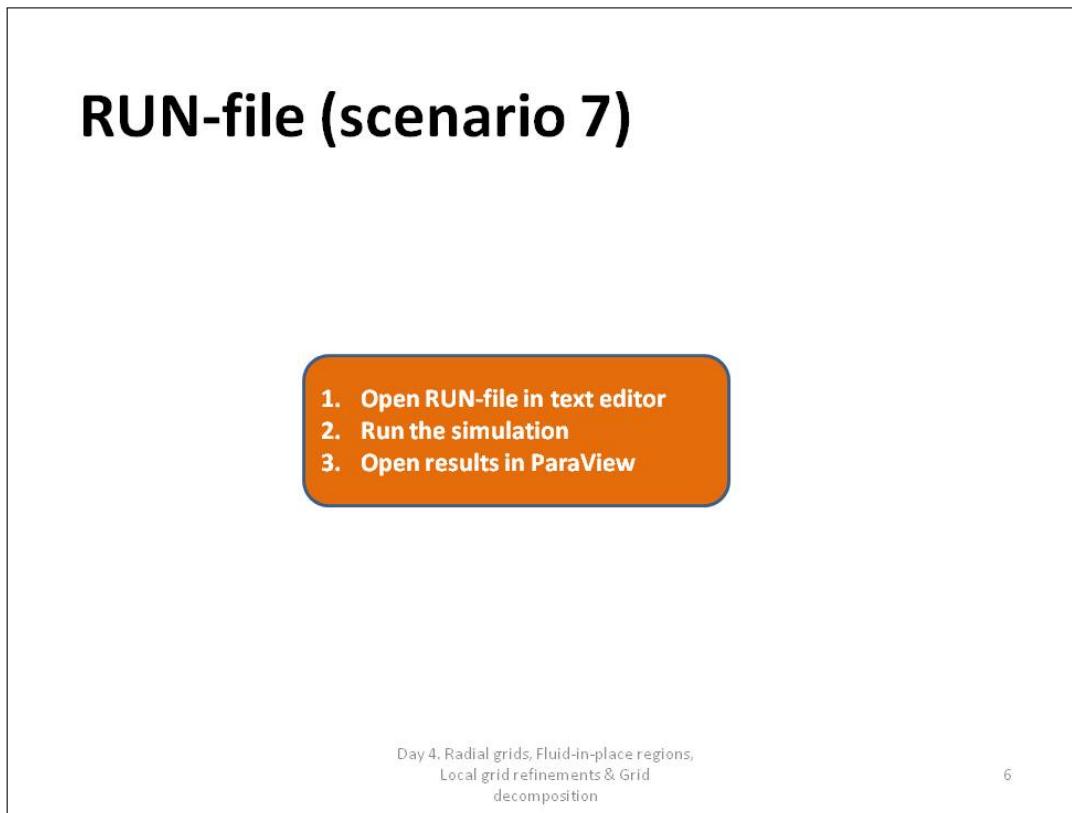
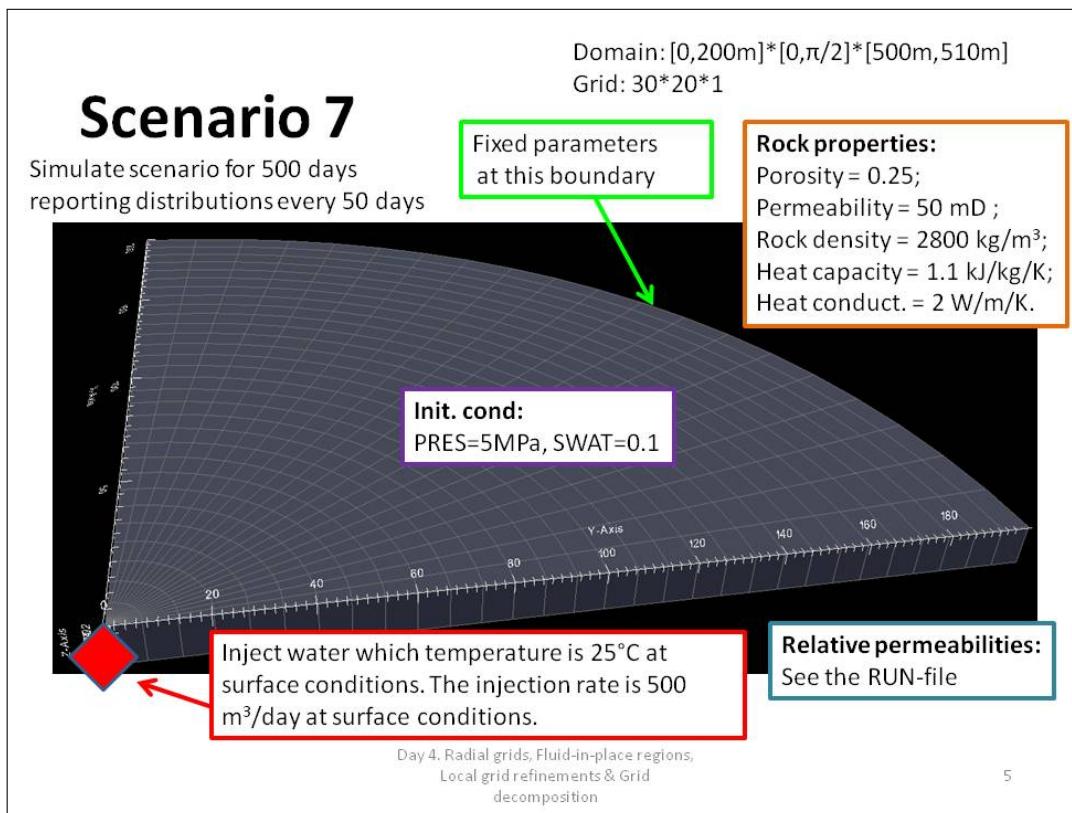
A diagram showing a single blue dot representing a grid point. Three arrows point away from the dot, labeled 'r (i-index)' pointing right, 'θ (j-index)' pointing up-right, and 'z (k-index)' pointing down.

Select this option

A red arrow points to the line 'gridtype = RADIAL - Radial Grid' in the code snippet.

Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

4




```

96 REGALL
97
98 EQUALREG
99   PRES 5. SATNUM 0 /
100  SWAT 0.1 SATNUM 0 /
101 /
102
103 EQUALNAM
104   PRES 0.1 'INJE*' /
105  TEMPC 25 /
106 /
107
108 RPTSUM
109   PRES SWAT SVAP TEMPC /
110
111
112 SCHEDULE ##### SCHEDULE section begins here #####
113
114 SRCINJE
115   'INJE*' RATEIN 1* 15. 25. /
116 /
117
118 TSTEP
119   10*50 / We advance simulation to 500 days.
120
121 POST ##### POST section begins here #####
122
123 CONVERT
124
125
126 END #####

```

Radial grids

The domain boundaries are defined by the keyword **RTZBOUND**

```
RTZBOUND syntax
1 -- within MAKE/ENDMAKE brackets.
2
3 RTZBOUND
4   rmin rmax tmin tmax zmin zmax rincr tincr zincr /
5
6 =====
7
8   rmin/rmax - the domain boundaries along axis r (rmin<rmax)
9   tmin/tmax - the domain boundaries along axis theta (tmin<tmax) [rad]
10  zmin/zmax - the domain boundaries along axis z (zmin<zmax)
11  rincr - the increment of the grid block sizes along axis R. With
12    increasing i-index every next grid block is xincr times larger
13    than the previous block;
14  tincr - the increment of the grid block sizes along axis Theta. With
15    increasing j-index every next grid block is tincr times larger
16    than the previous block;
17  zincr - the increment of the grid block sizes along axis Z. With
           zincr times larger
```

Exercise: Re-simulate scenario 7 using theta range $[0, \pi/4]$

Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

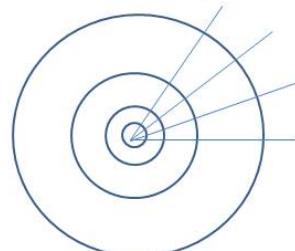
7

Radial grids

The grid block extensions can be redefined using the **DRV**, **DTHETAV**, **DZV** keywords

```
DRV syntax
1 -- within MAKE-ENDMAKE brackets
2
3 DRV
4   dr(1) dr(2) dr(3) ... dr(nr) /
5
6 =====
7
8   dr(#) - grid blocks extensions along axis R.
9   nr - number of grid block along axis R. nr is the 2nd argument of the
       keyword MAKE.
```

Exercise: Re-simulate scenario 7 applying a grid refinement to the center.



Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

8

5.2. Fluid-in-Place regions

Fluid-in-place regions

Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

9

FIPNUM regions

FIP = Fluid-in-Place

FIPNUM region numbers can be used for

- calculate average value in a region of reservoir domain (e.g., average pressure, temperature);
- integrate a property in a region (e.g., calculate total mass of a component in domain);
- calculate parameters for boundary between two regions of domain (e.g., calculate total mass flux between two regions).

To use Fluid-in-Place option you should

1. Define different Fluid-in-Place regions in **GRID** or **INIT** sections using mnemonic **FIPNUM** (by default in all cells **FIPNUM=0**).
2. Specify the properties to be reported for the regions using **RPTFIP** keyword.

You can create consolidated time series data for FIPNUM regions in the **POST** section using **POSTFPCE** and **POSTFPCO** keywords.

Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

10

Keyword RPTFIP

The output for FIPNUM regions in the file SCENARIO%.####.SUM is controlled by the RPTFIP keyword

```
-- 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.
```

Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

11

Keyword POSTFPCE

By using this keyword you can create consolidated time series data for FIPNUM regions.

```
-- in POST section
POSTFPCE
  fipnum1 filename1 /
  fipnum2 filename2 /
  fipnum3 filename3 /
...
/
=====
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).
```

Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

12

Keyword POSTFPCO

By using this keyword you can create consolidated time series data for boundary between two FIPNUM regions.

```

1 -- in POST section
2
3 POSTFPCO
4   fipnumai  fipnumbi  filename1 /
5   fipnuma2  fipnumb2  filename2 /
6   fipnuma3  fipnumb3  filename3 /
7   ...
8   /
9
10 =====
11
12   fipnuma# - two fluid-in-place region numbers for which the output is
13     -fipnumb# required. The flow rate is reported in the direction
14       from fipnuma# to fipnumb#.
15   filename# - output file name (if not specified the program uses default
16     naming convention).

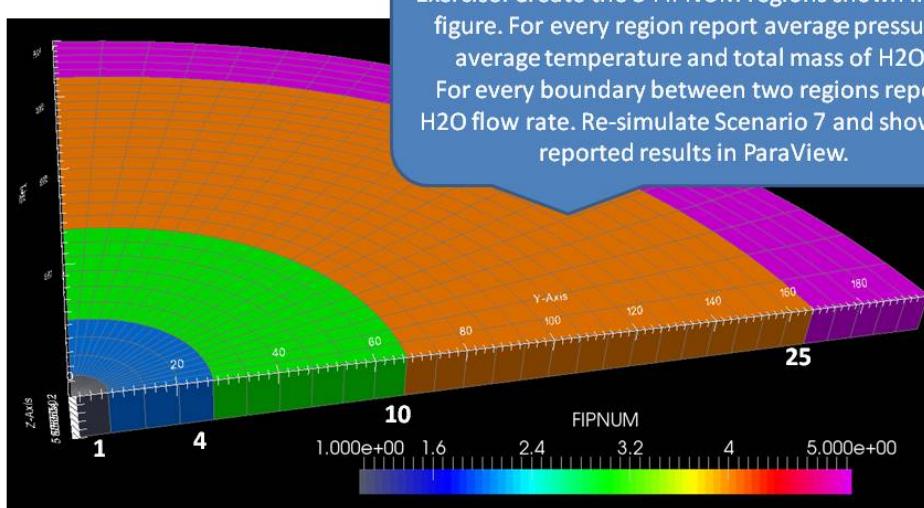
```

Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

13

Using FIPNUM regions

Exercise: Create the 5 FIPNUM regions shown in the figure. For every region report average pressure, average temperature and total mass of H₂O. For every boundary between two regions report H₂O flow rate. Re-simulate Scenario 7 and show all reported results in ParaView.



Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

14

Exercise

Re-simulate scenario 7 using provided heterogeneous distribution of permeability and taking into account heat exchange with impermeable rocks

Cross-section view

Constant $T=244\text{C}$

impermeable

impermeable

Constant $T=284\text{C}$

The grid block sizes are increasing along axis r with increment 1.1.

Day 4. Radial grids, Fluid-in-place regions, Local grid refinements & Grid decomposition

15

RUN-file (scenario 7, exercise)

1. Open RUN-file in text editor
2. Run the simulation
3. Open results in ParaView

Day 4. Radial grids, Fluid-in-place regions, Local grid refinements & Grid decomposition

16


```
144  
145 EQUALNAM  
146 PRES 0.1 'INJE*' /  
147 TEMPC 25 /  
148 /  
149  
150 RPTSUM  
151 PRES TEMPC SWAT SVAP K-IJKRES/  
152  
153 RPTFIP We specify the properties saved in the  
154 PRES TEMPC MASS#1 FLUX#1 / summary files for Fluid-in-Place  
155 regions.  
156  
157 SCHEDULE ##### SCHEDULE section begins here #####  
158  
159 TUNING  
160 2* 0.01 / The initial timestep is 0.01 days.  
161  
162 SRCINJE We specify that the injection rate is  
163 'INJE*' RATEIN 1* 6. 25. 1* 100./ 25 m3/day for every source (500 m3/day in  
164 / total).  
165  
166 TSTEP  
167 1 2 3 4 10 30 9*50 / We advance simulation to 500 days.  
168  
169 POST ##### POST section begins here #####  
170  
171 CONVERT We convert the output to ParaView  
172 compatible format.  
173  
174 RPTPOST We define the properties reported in  
175 NOTHING TIME PRES TEMPC MASS#1 / the following POSTFPCE keyword.  
176  
177 POSTFPCE We report the properties the average  
178 1 / pressure, average temperature and H2O  
179 2 / in place for the FIPNUM regions 1,2,3,  
180 3 / 4,5. The time series data for every  
181 4 / region is saved in a separate file.  
182 5 /  
183 /  
184  
185 RPTPOST We define the properties reported in  
186 NOTHING TIME FLUX#1 / the following POSTFPC0 keyword.  
187  
188 POSTFPC0 We report the H2O fluxes from FIPNUM  
189 1 2 / region 1 into region 2, from region 2  
190 2 3 / into 3 etc.  
191 3 4 /
```

```

192    4 5 /
193    /
194
195 END #####
```

SIMULATIONS/SCENARIO7/EXERCISE/Permeability.txt

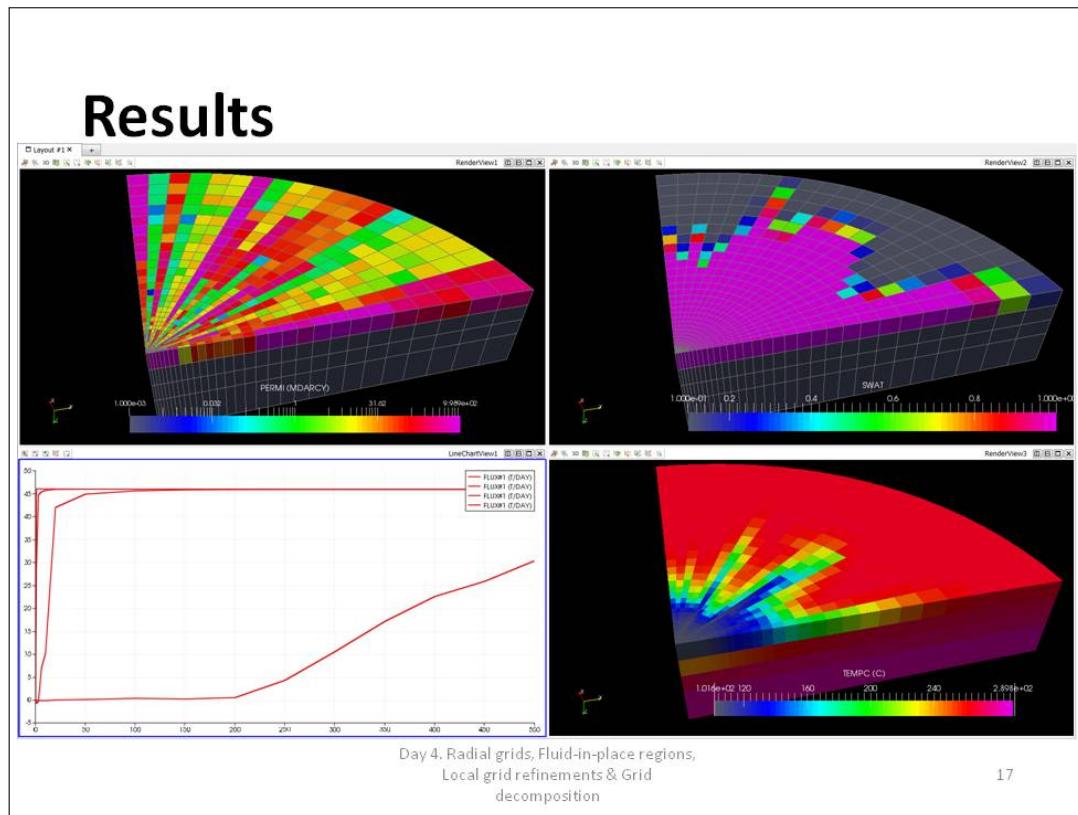
PERMI						
69.449000	84.463100	21.825500	25.383500	47.996300	31.218200	
51.464800	61.415200	35.397200	27.507700	31.216800	69.207300	
42.197300	45.498200	82.056800	80.731100	68.551500	85.716100	
52.069500	93.817400	59.224400	700.291400	571.556500	544.728100	
991.184900	542.092300	796.867900	845.611900	660.189600	783.654800	
6.309900	16.137400	8.674400	15.116400	3.086600	0.076200	
0.261900	0.069200	0.904100	0.205800	0.304200	0.932700	
0.918800	0.357000	0.315200	0.869600	0.012300	0.722600	
0.272900	0.528700	0.827500	0.706400	0.536800	0.862400	
0.987700	0.108900	0.036900	0.316800	0.277200	0.314200	
14.613900	8.267500	4.830400	10.494800	11.968300	116.889800	
243.240400	262.338700	321.874500	10.907900	93.542000	18.474800	
337.243100	288.969400	434.425600	347.226200	494.961500	68.086100	
139.373100	161.747500	437.310300	315.368000	99.635800	21.453500	
61.183200	39.316500	57.010000	98.307000	26.968400	96.663400	
892.243300	7.295800	7.588700	7.454200	19.279700	2.137700	
3.009400	18.778200	9.759700	15.783200	17.090600	13.568700	
12.079500	5.905200	0.294000	0.609600	0.145700	0.334700	
0.527100	0.277000	0.165600	0.588700	0.674700	0.253400	
0.551500	0.041500	0.597900	0.908900	0.789500	0.834100	
34.641300	63.649500	95.858500	85.384300	8.539900	16.795000	
8.749200	11.770500	1.772900	5.359300	13.800600	5.430200	
7.281200	8.045600	4.682100	4.252300	6.811500	9.434800	
9.703400	1.232000	6.064400	13.014900	9.284200	2.928200	
5.007100	11.493500	1.267300	18.775600	6.185900	14.110600	
431.914600	463.460100	70.263500	271.128800	409.645500	376.508800	
308.891100	243.426400	324.371000	8.846900	15.694000	12.396400	
6.001200	11.886900	9.226800	19.339000	18.066000	15.572000	
5.201300	10.284700	16.039400	5.947600	13.259700	17.508000	
5.175600	14.907700	1.908600	16.413100	8.374300	9.757600	
39.734600	83.350100	91.766900	70.747500	91.304000	88.092200	
96.164300	64.831100	47.384000	25.425800	46.561000	713.725400	
676.238700	844.775300	919.047900	918.110600	877.838800	702.216400	
910.619900	898.286400	702.151400	639.246700	559.843400	832.528900	
553.404700	846.593500	998.241200	541.203800	782.604200	641.448900	
6.115000	3.751100	19.251200	18.555400	8.368500	15.548800	
2.799600	8.927300	18.182900	8.451600	15.307600	6.198200	
19.272800	9.624400	0.945200	0.313600	0.358700	0.833000	
0.679900	174.508100	382.882300	224.823900	174.288600	306.522000	
119.256800	272.367600	35.044700	0.291600	0.828900	0.911000	
0.639200	0.200600	0.298900	0.941500	0.951400	0.119200	

43	0.843300	0.125900	0.506100	0.329900	0.427700	0.086900
44	92.840700	95.061700	29.396900	20.471800	44.101500	64.942700
45	38.128700	83.640300	82.159600	81.159200	79.975600	11.121400
46	16.925000	10.439500	16.236800	18.097300	16.657900	11.116000
47	34.862400	27.738600	84.432100	89.855900	29.461800	42.185500
48	48.312400	96.015700	70.316600	28.215900	83.747800	0.084700
49	0.235600	0.886600	0.569100	0.591500	0.335500	0.746400
50	210.194000	156.685400	301.504500	104.947500	298.659700	165.084700
51	426.777300	347.142100	269.159800	3.140700	15.297900	18.982600
52	0.882300	0.998200	11.433400	7.151500	2.281700	5.641100
53	10.152500	4.468000	11.218000	18.589900	0.431600	0.967600
54	0.242300	0.537200	0.058600	0.315500	0.786500	0.839900
55	0.650900	0.598300	0.376700	0.803600	0.064600	89.190700
56	79.961100	63.186100	72.318100	82.103800	29.322600	39.382300
57	10.042900	8.767900	9.847800	6.625300	9.487000	9.313500
58	10.529200	8.765800	11.405900	4.521900	11.976400	7.698600
59	7.404400	15.272800	9.347500	11.677000	18.459400	16.708300
60	5.204100	52.623100	70.453200	50.719200	30.834500	90.799800
61	78.452600	49.012800	32.675600	44.001300	26.851200	48.622100
62	693.505600	750.802900	910.860700	553.991600	963.727900	681.770600
63	645.655100	663.817400	657.752000	975.945700	959.276800	812.566800
64	711.971600	952.900400	818.148800	720.935300	552.526600	539.140700
65	787.815200	575.927000	70.248500	97.897000	60.270600	59.936700
66	29.349200	34.827300	40.770100	22.854300	98.966900	83.609500
67	0.889900	0.909000	0.329700	0.438300	0.911300	0.005500
68	0.818700	0.192500	0.512300	0.496700	0.736600	0.554400
69	0.466100	348.883700	109.161600	460.456600	152.947800	169.771700
70	52.371000	382.580100	273.012300	342.310100	276.821000	205.262100
71	0.524600	0.706200	0.421100	0.999400	0.394100	0.153600
72	92.230200	22.745000	60.781600	83.890200	87.654000	37.989000
73	94.303500	62.266400	34.935200	80.993500	98.847200	55.331100
74	87.773900	88.173300	0.353500	0.050800	0.735900	0.546400
75	0.160300	0.680200	0.685900	0.680200	0.927600	0.411800
76	0.393100	0.811800	10.701900	1.268800	7.686900	4.003200
77	34.398200	34.897600	37.214700	85.471700	46.847200	74.325100
78	63.740500	78.872800	98.334600	58.009300	22.783300	61.327300
79	97.672900	73.336600	93.741200	52.682700	48.596000	86.365600
80	98.434200	98.958400	24.760100	14.496200	4.824800	16.994500
81	11.626400	18.366200	10.123700	6.036600	12.173000	12.239300
82	15.498500	2.633500	7.222000	10.554000	8.106700	9.543000
83	8.188200	2.811700	16.793000	16.999800	9.954900	18.123200
84	14.393200	2.083800	5.053700	12.741100	15.212500	11.547900
85	3.141800	16.746800	7.338600	18.130400	1.315300	9.363400
86	17.953800	4.867400	11.421000	3.268200	7.647600	11.405300
87	444.629600	54.471100	26.319900	67.314400	21.303500	44.300900
88	93.133100	55.258800	83.509600	39.503800	684.254000	851.922700
89	821.307000	939.792500	55.942300	32.271300	93.980200	53.973600
90	18.490600	5.573600	7.708900	17.436200	10.324600	6.383900

```

91      4.392600    3.787800    1.003200   13.399300    8.174700    7.898800
92      0.001000    0.943600    0.002400    0.015800    0.026500    0.181000
93      97.923100   85.437700   73.014600   33.221800   70.562000   77.756400
94      71.638400   47.406700   89.987300   21.255700   81.867300   46.448200
95      70.895100   70.534100   87.889600   88.812200   85.715700   26.674200
96      69.127900   76.948400   377.412500  298.479400  154.755600  173.179300
97      500.000000  696.991000  990.610800  531.251500  949.130700  998.915400
98      676.511700  768.961600   7.588500    8.443600    86.626300   32.870300
99      76.550600   43.860600   38.771800   60.392100   93.371000   781.887200
100     896.694900  805.648200  590.350600  923.162800  926.181200  838.321700
101     826.446100  348.530400  224.614200  129.852800  435.660800  361.416500
102     /

```



5.3. Local grid refinements (Scenario 8)

Local grid refinements (LGRs)

Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

18

Scenario 8

Simulate scenario up to 100000 days reporting distributions every 1000 days

Grid: 30*5. EOS-module:
BINMIXT

interpretation

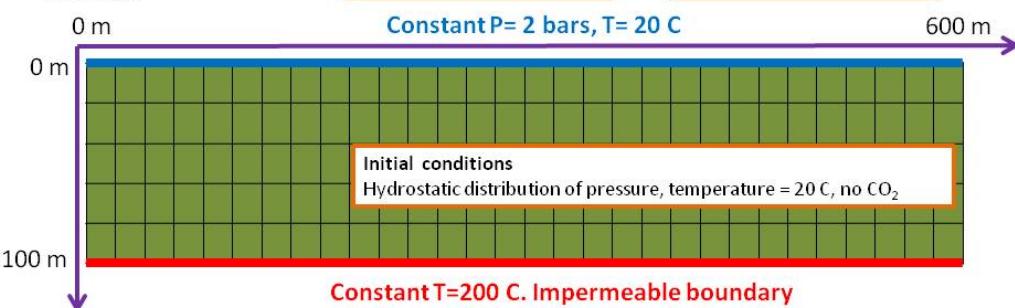
Atmosphere

Water (e.g., a lake)

Porous media
(convection develops)

High temperature (impermeable)

Rel. perm:
Brooks & Corey, $s_{min} = 0.2$
 $s_{max} = 0.95$



Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

19

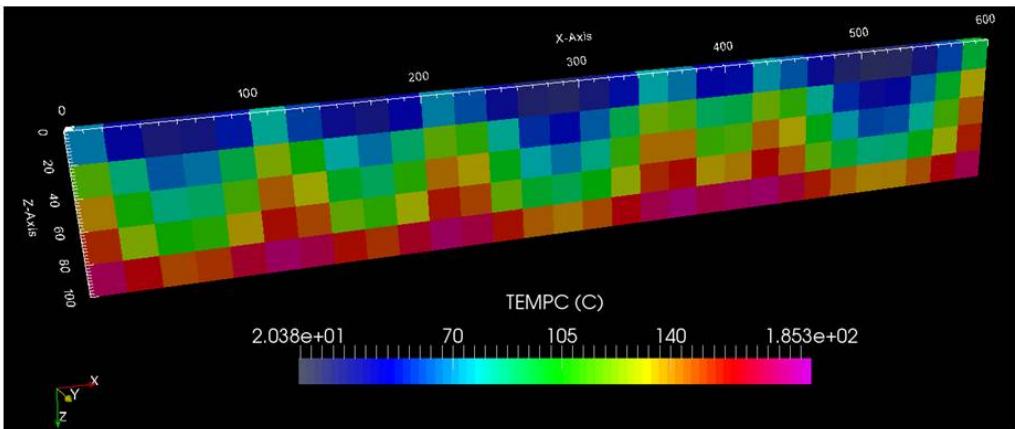
RUN-file (scenario 8)

1. Open RUN-file in text editor
 2. Run the simulation
 3. Open results in ParaView

Day 4. Radial grids, Fluid-in-place regions, Local grid refinements & Grid decomposition

20

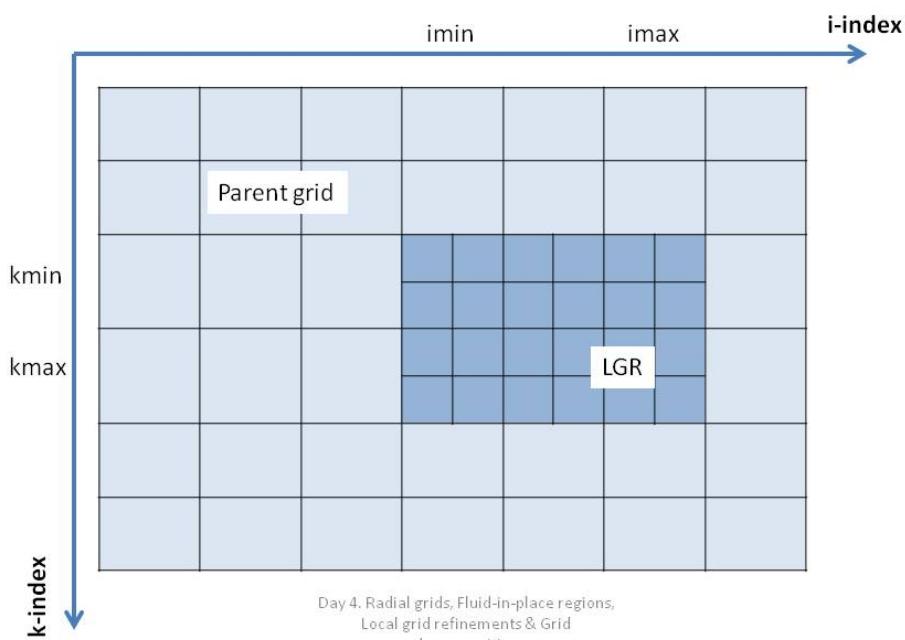
Result (scenario 8)



Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

21

Local grid refinements



Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

22

CARFIN keyword

The **CARFIN** keyword defines local grid refinements

```
CARFIN syntax
1 -- within MAKE-ENDMAKE brackets
2
3 CARFIN
4   name  iminimax jminjmax kminkmax nx ny nz parent /
5
6 =====
7
8   name      - name of the refined grid;
9   iminimax - the boundaries of the refined grid along i-index direction
10  in the parent grid;
11  jminjmax - the boundaries of the refined grid along j-index direction
12  in the parent grid;
13  kminkmax - the boundaries of the refined grid along j-index direction
14  in the parent grid;
15  nx        - the number of grid blocks in the refined grid along i-index
16  direction;
17  ny        - the number of grid blocks in the refined grid along j-index
18  direction;
19  nz        - the number of grid blocks in the refined grid along k-index
20  direction;
21  parent    - the parent grid name.
```

Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

23

REFINE & ENDFIN keywords

Keyword **REFINE** selects a grid to be active. It affects the **BOX** keyword and arrays loading. After the keyword **CARFIN** the created grid is active.

```
REFINE syntax
1 -- in every section except RUNSPEC and POST
2
3 REFINE
4   gridname  resname /
5
6 =====
7
8   gridname - grid name (8-byte character);
9   resname  - the name of reservoir in which the grid is defined.
```

Keyword **ENDFIN** resumes the active grid to the initial grid encompassing the whole reservoir

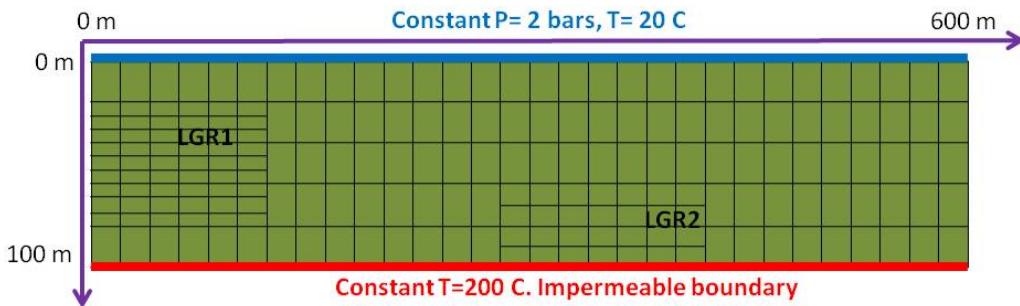
```
ENDFIN syntax
1 -- in every section except RUNSPEC and POST
2
3 ENDFIN
```

Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

24

Local grid refinements

Exercise: Re-simulate scenario 8 using the following grid.



Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

25

Answer

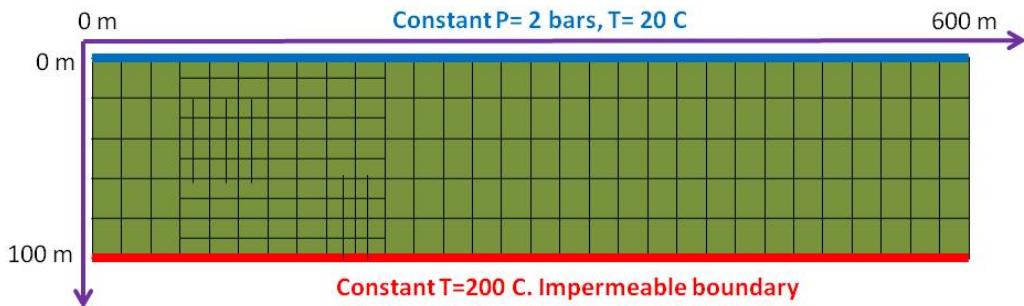
```
1 -- within MAKE-ENDMAKE brackets Day 4. Answer
2
3 CARFIN
4   LGR1  1  6  1 1  2 4      6  1  9 /
5 CARFIN
6   LGR2  15 21  1 1  4 5      7  1  4 /
7 ENDFIN
```

Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

26

Nested LGRs

Exercise: Re-simulate scenario 8 using the following grid.

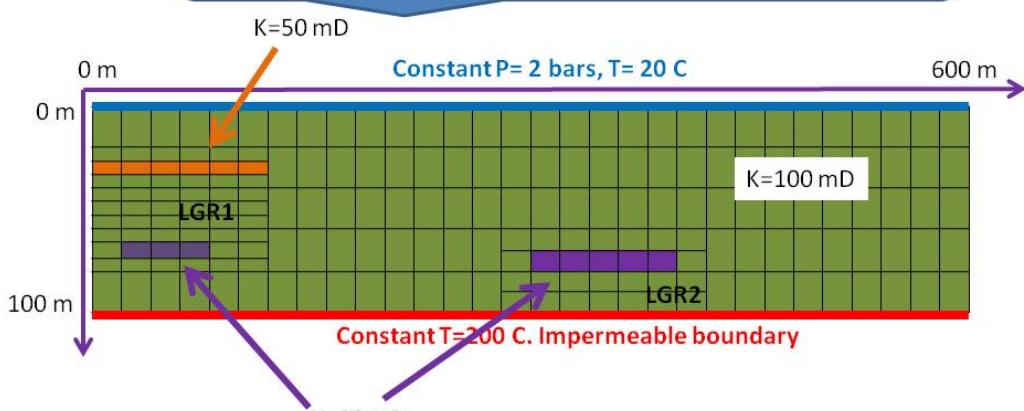


Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

27

LGRs; Arrays loading

Exercise: Re-simulate scenario 8 using the following permeability distribution.



Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

28

LGRs; Arrays loading (answer)

```
1 -- in GRID section
2
3 ENDFIN
4 EQUALS
5   PERMX 100 /
6 /
7 REFINE
8   LGR1 /
9 EQUALS
10  PERMX 50  4*      2*2 /
11  PERMX 30  2 4 2*  2*8 /
12 /
13 REFINE
14   LGR2 /
15 EQUALS
16  PERMX 30  2 6 2*  2*2 /
17 /
18 ENDFIN
19 COPY
20  PERMX PERMZ /
21 /
```

Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

29

Local grid refinements

Exercise: Re-simulate scenario 8 using twice as more refined grid as
the initial grid. Do not change the MAKE keyword.

Exercise: Apply random porosity variations to the refined grid.

Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

30

More complicated LGRs

More complicated LGRs can be created using HXFIN, HYFIN, HZFIN, NXFIN, NYFIN, NZFIN keywords (see the Reference manual).

Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

31

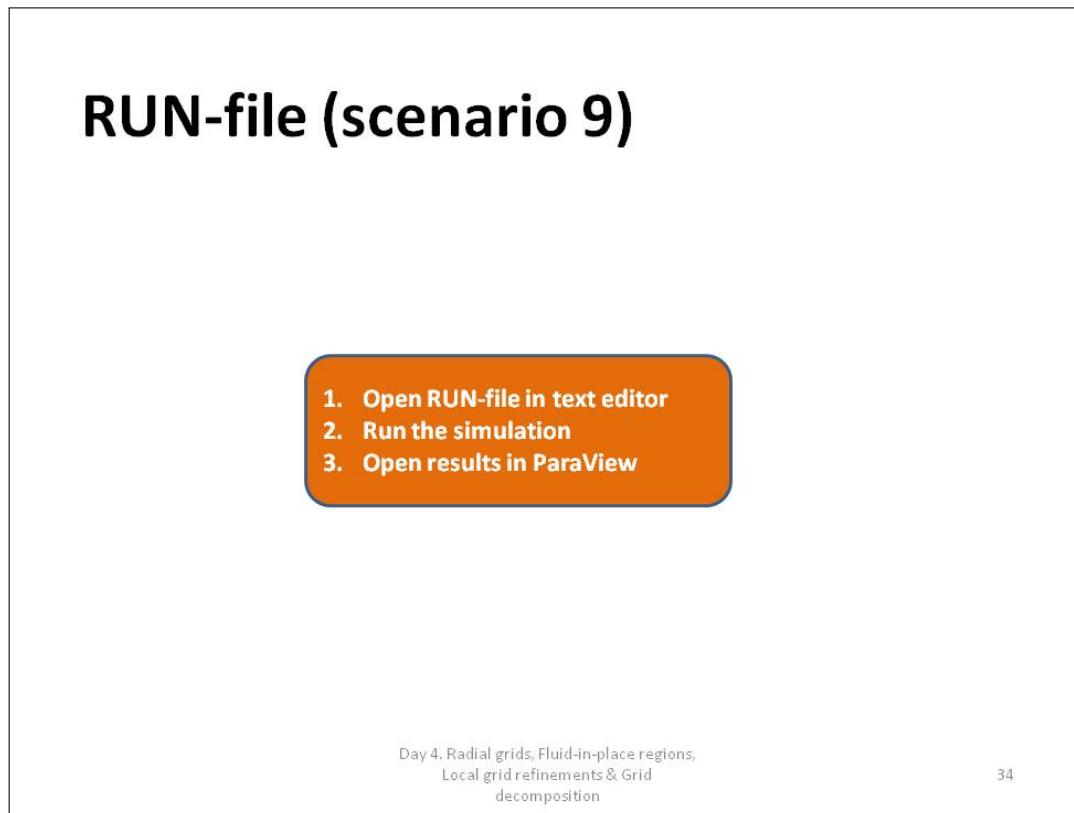
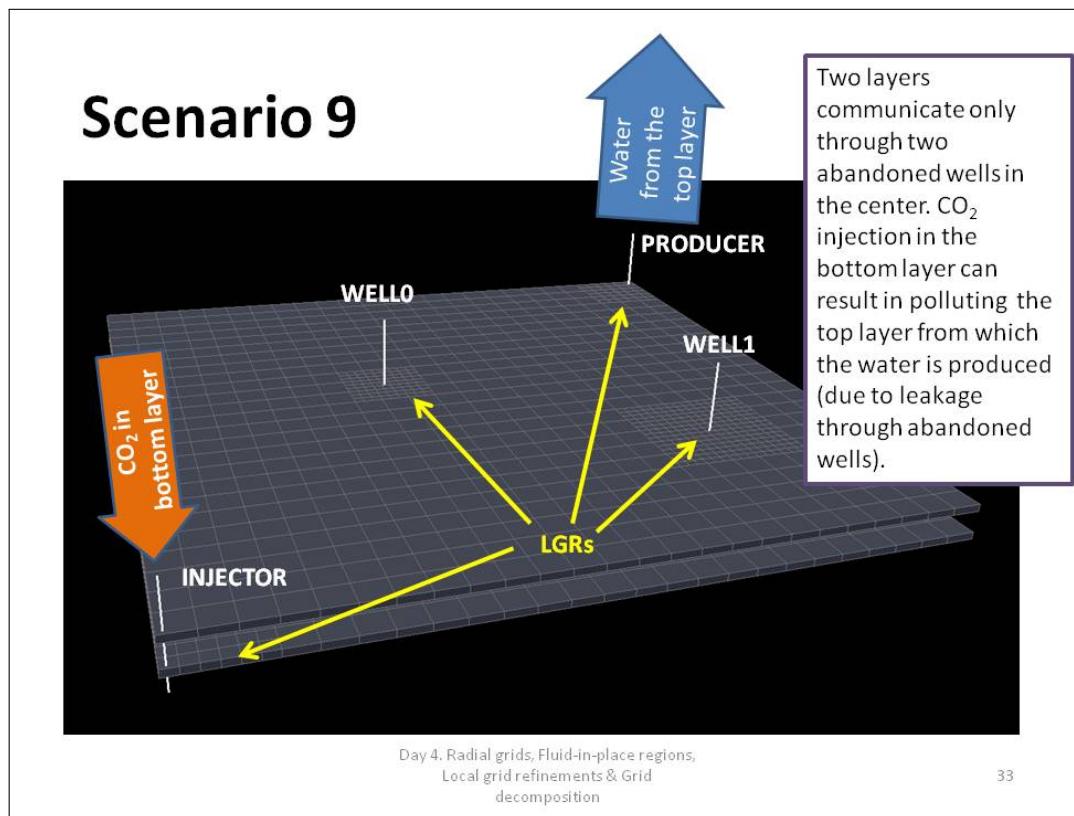
5.4. Scenario 9

Scenario 9

(simulations with both LGRs and Wells)

Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

32

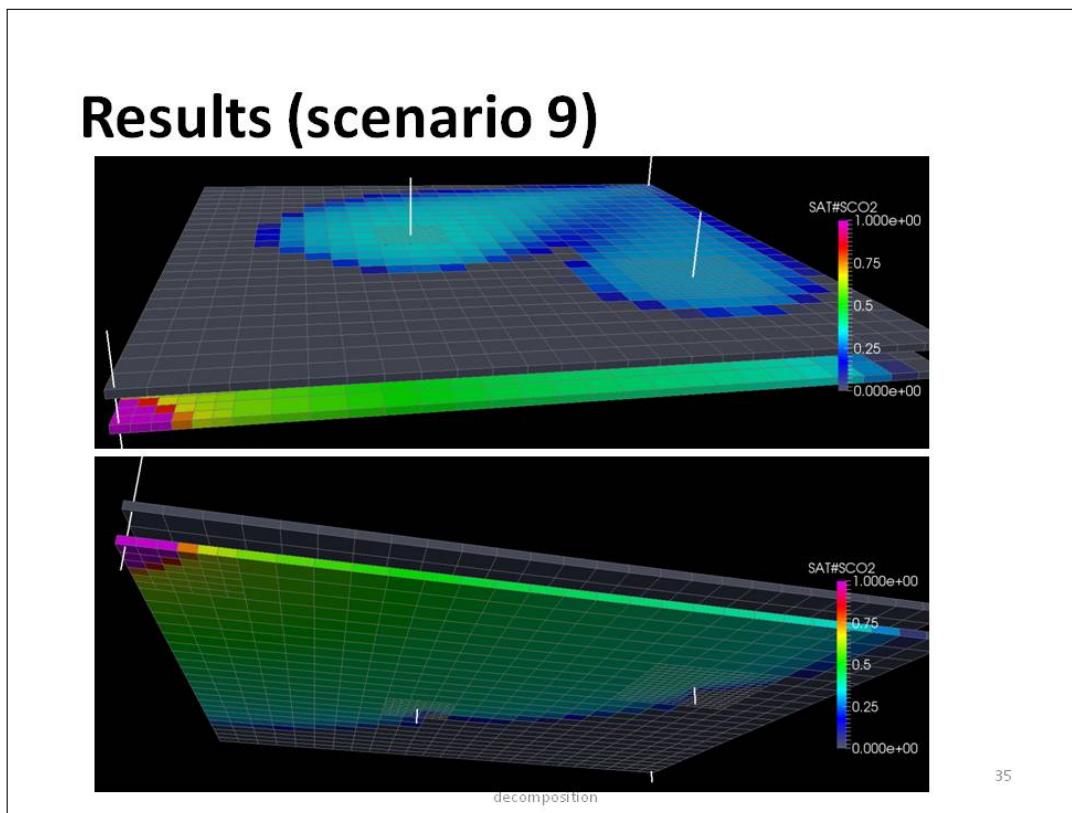


SIMULATIONS/SCENARIO9/0/SCENARIO9.RUN

```

1 RUNSPEC ##### We enable the option FAST.
2
3 FAST
4
5 GRID #####
6
7 MAKE <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
8   CART 25 25 3 / We specify that the Cartesia grid
9           is 25*25*3 grid blocks.
10
11 XYZBOUND We specify the reservoir extent
12   -- xmin-xmax    ymin-ymax    zmin-zmax along every axis. The top layer is
13     0.0 1000.0    0.0 1000.0    1000 1050 / at 1 km depth.
14
15 DZV The distance between the top and
16   10 30 10 / the bottom layers is 30 meters.
17
18 BOX We make inactive the layer 2.
19   4* 2*2 /
20 ACTNUM
21   1000000*0 /
22 ENDBOX
23
24 CARFIN <<<<<< We specify LGR near injection well <<<<<<<<<<<<<<<<<<<<<<<<<<<
25   LGRINJE      1 3 1 3 3 3      6 6 1 /
26 WELSPECS We specify that INJECTOR
27   ----- wellhead is in block i=1, j=1
28   INJECTOR 1* 1 1 1005 6* 222 / of the grid LGRINJE.
29 /
30 COMPDAT The INJECTOR is completed only in the layer 1 of grid LGRINJE
31   -----
32   INJECTOR 1* 1* 1 1 OPEN 1* 1* 0.2 /
33 /
34
35 CARFIN <<<<<<< We specify LGR near abandoned WELL0 <<<<<<<<<<<<<<<<<<<<<<<<<<<
36   LGRWELLO 14 16 9 11 1 3 9 9 3 /
37 WELSPECS We specify that WELL0 wellhead
38   ----- is in block i=5, j=5 of the
39   WELL0 1* 5 5 1005 6* 333 / grid LGRWELLO.
40 /
41 COMPDAT The WELL0 is completed in the layers 1 and 3.
42   -----
43   WELL0 1* 1* 1 1 OPEN 1* 1* 0.2 /
44   WELL0 1* 1* 3 3 OPEN 1* 1* 0.2 /
45 /
46
47 CARFIN <<<<<<<< We specify LGR near abandoned WELL1 <<<<<<<<<<<<<<<<<<<<<<<<<<<
```

```
192    100*100 /           at every 100 days.  
193  
194 POST      #####  
195  
196 CONVERT          We convert the output to ParaView  
197               compatible format.  
198  
199 RPTPOST  
200   TIME WBHP WBHTC WMPR#2 WMPT#2  
201           WMPR#1 WMPT#1 /           We define the properties outputed from  
202               THE following POSTWELL keyword.  
203  
204 POSTWELL          We save consolidated time series  
205   PRODUCER /           report for PRODUCER.  
206 /  
207  
208 RPTPOST  
209   NOTHING TIME WBHP WBHTC  
210           WMIR#1 WMIT#1 /           We define the properties outputed from  
211               the following POSTWELL keyword. NOTHING  
212               entry clears the output list.  
213  
214 POSTWELL          We save consolidated time series reports  
215   INJECTOR /           for injection well.  
216 /  
217  
218 END      #####
```



5.5. Grid decomposition

Grid decomposition

Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

36

Grid decomposition

The number of parallel processes (cores) assigned to simulation are specified in the command line when you launch the simulation (an example of commands for 3 processes are below).

Mac:

```
mpirun -n 3 ./../BIN/H64.EXM SCENARIO9.RUN > SCENARIO9.LOG
```

Linux:

```
mpirun -n 3 ./../BIN/H64.EXL SCENARIO9.RUN > SCENARIO9.LOG
```

Windows:

```
"...mpiexec.exe" -n 3 ./../BIN/H64.EXE SCENARIO9.RUN > SCENARIO9.LOG
```

By default the simulator automatically balances simulation between cores.
The user assistance is not required.

Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

37

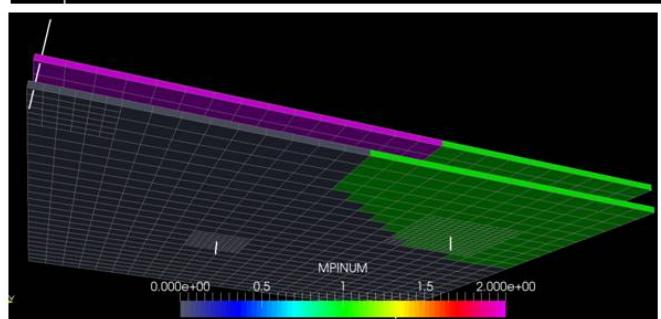
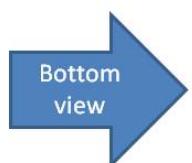
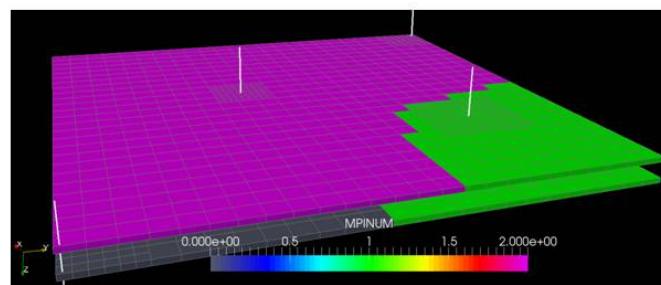
Automatic grid partition

Exercise: Re-simulate Scenario 9 on 3 cores.
Save the MPIRANK and MPINUM property
from the GRID section

Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

38

Result



Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

39

Decomposition specified by user

There are two options to specify decomposition by user:

1. Using MPINUM keyword;
2. Using the PARTIT keyword.

Normally, when these options are chosen the NOAUTO keyword should be specified within brackets MAKE-ENDMAKE. The NOAUTO switch disables automatic grid partition.

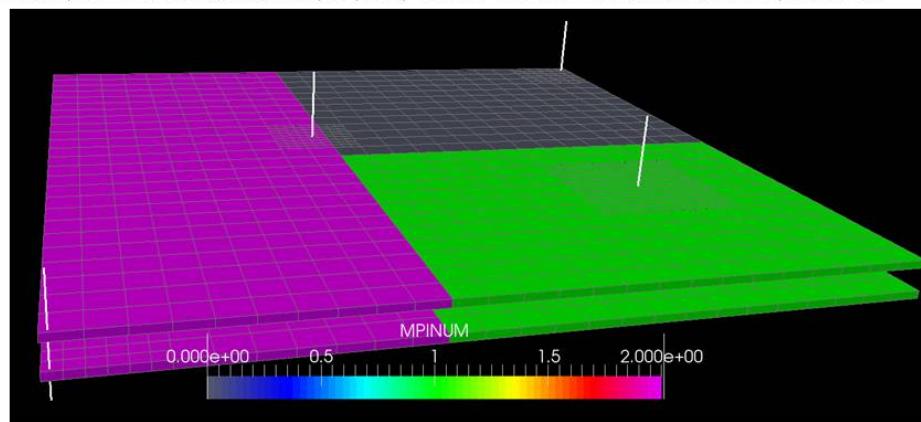
Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

40

MPINUM region

Exercise: Specify the grid decomposition using MPINUM keyword and re-simulate scenario 9 using 3 processes. Save MPIRANK and MPINUM properties from GRID section.

The MPINUM keyword can be used to specify a user-defined decomposition within MAKE-ENDMAKE brackets. The cells with equal MPINUM number are assigned to the same process. The MPIRANK property can be used to check the created partition.



Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

41

Answer

```

1 NOAUTO
2
3 MPINUM
4   100000*0 /
5
6 BOX
7   1 12 /
8
9 MPINUM
10  100000*1 /
11
12 BOX
13  1 25 1 10 /
14
15 MPINUM
16  100000*2 /
17
18 ENDBOX

```

Day 4. Answer

Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

42

Keyword PARTIT

The PARTIT keyword creates a Cartesian grid decomposition. Normally, this keyword should be used with the NOAUTO keyword which disables automatic grid partition.

```

1 -- within MAKE-ENDMAKE brackets          PARTIT syntax
2
3 PARTIT
4   ni  nj  nk  nsrt /
5
6 =====
7
8   ni  - number of the grid partition regions along i-index direction;
9   nj  - number of the grid partition regions along j-index direction;
10  nk  - number of the grid partition regions along k-index direction;
11  nsrt - the initial number from which the partition regions are numbered.
12      The MPINUM property is assigned to the regions with i-index cycling
13      the fastest following by the j- and k-indexes.
14

```

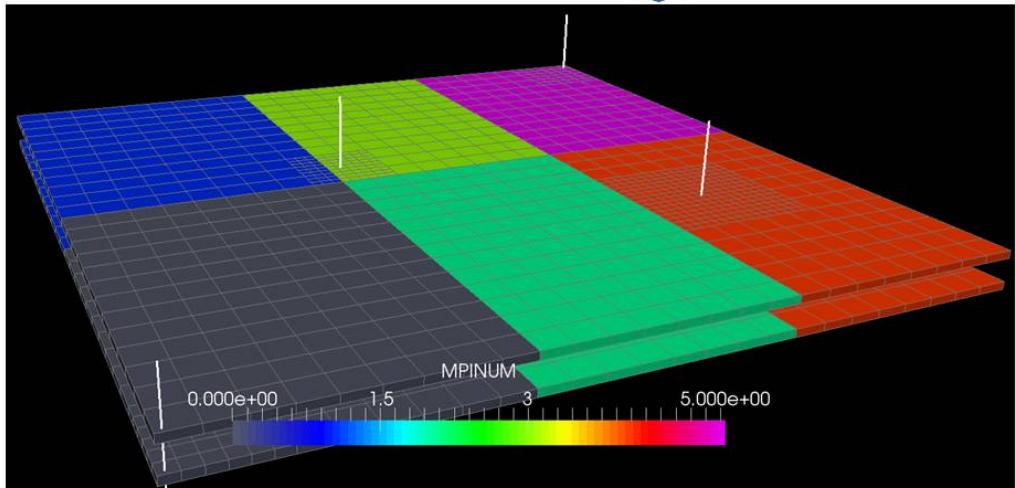
More complicated partitions can be created using PARTI, PARTJ and PARTK keywords associated with the PARTIT keyword.

Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

43

Exercise

Exercise: Specify the grid decomposition using PARTIT keyword and re-simulate scenario 9 using 6 processes. Save MPIRANK and MPINUM properties from GRID section.



Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

44

Answer

```
1 -- within MAKE-ENDMAKE brackets
2
3 NOAUTO
4
5 PARTIT
6   2 3 /
```

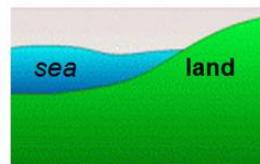
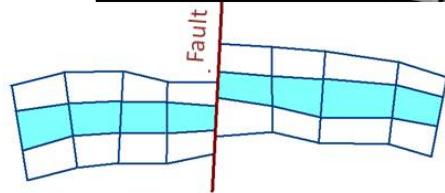
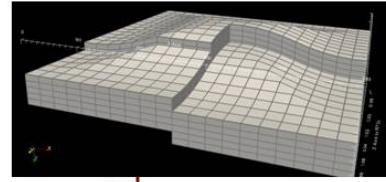
Day 4. Answer

Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

45

Next day

- Corner-point grids
- Faults
- Aquifers
- Onshore/offshore boundary conditions



Day 4. Radial grids, Fluid-in-place regions,
Local grid refinements & Grid
decomposition

46

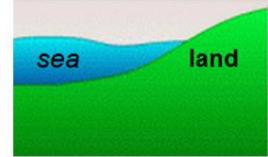
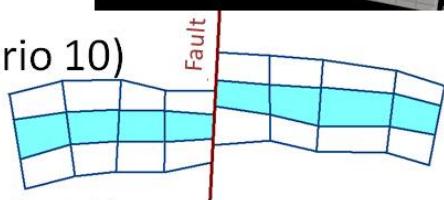
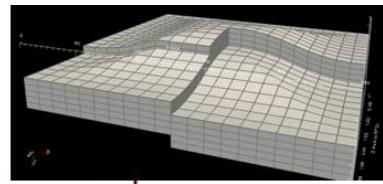
6 Day 5. Corner-point grids, Faults, Aquifers, & Onshore/offshore

MUFITS Training Course

Day 5
**Corner-point grids, Faults, Aquifers
& Onshore/offshore**

Program

- Corner-point grids (Scenario 10)
- Faults
- Modeling aquifers (Scenario 11)
- Onshore/offshore boundary conditions (Scenario 12)



Day 5. Corner-point grids, Fault, Aquifers &
Onshore/offshore

2

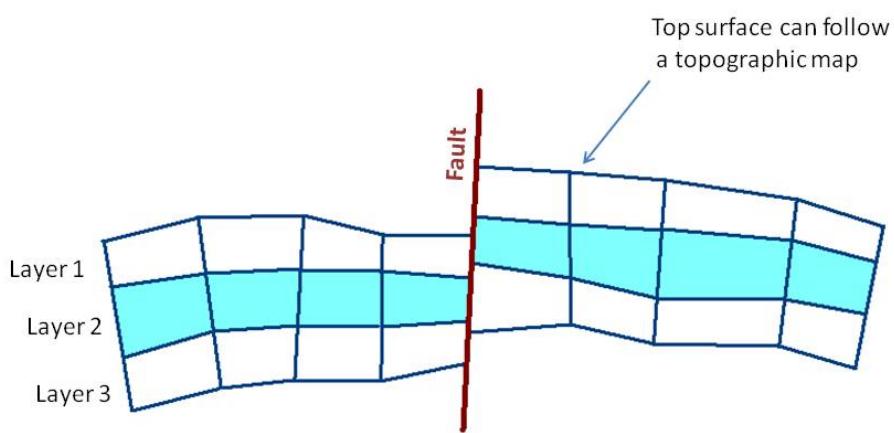
6.1. Corner-point grids (Scenario 10)

Corner-point grids

Day 5, Corner-point grids, Fault, Aquifers & Onshore/offshore

3

Corner-point grids



Day 5, Corner-point grids, Fault, Aquifers & Onshore/offshore

4

Corner-point grids

```
MAKE-ENDMAKE syntax
1 -- in GRID section
2
3 MAKE
4   gridtype ni nj nk /
5
6 -- other keywords
7
8 ENDMAKE
9
10 =====
11
12   gridtype = CART - Cartesian Grid
13     = RADIAL - Radial Grid
14     = CORNER - Corner-Point grid
15
16   ni - number of grid blocks along i-indexation axis
17   nj - number of grid blocks along j-indexation axis
18   nk - number of grid blocks along k-indexation axis
```

Select this option

Day 5. Corner-point grids, Fault, Aquifers &
Onshore/offshore

5

CARTesian/RADIAL to CORNER

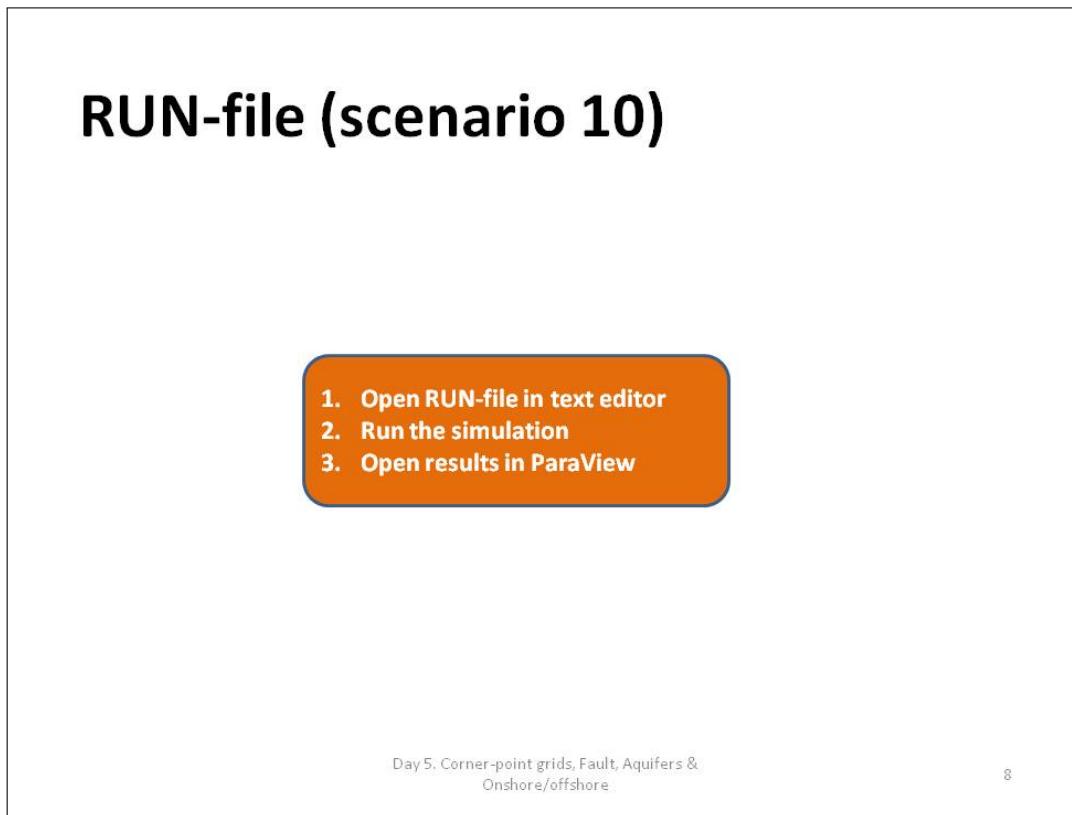
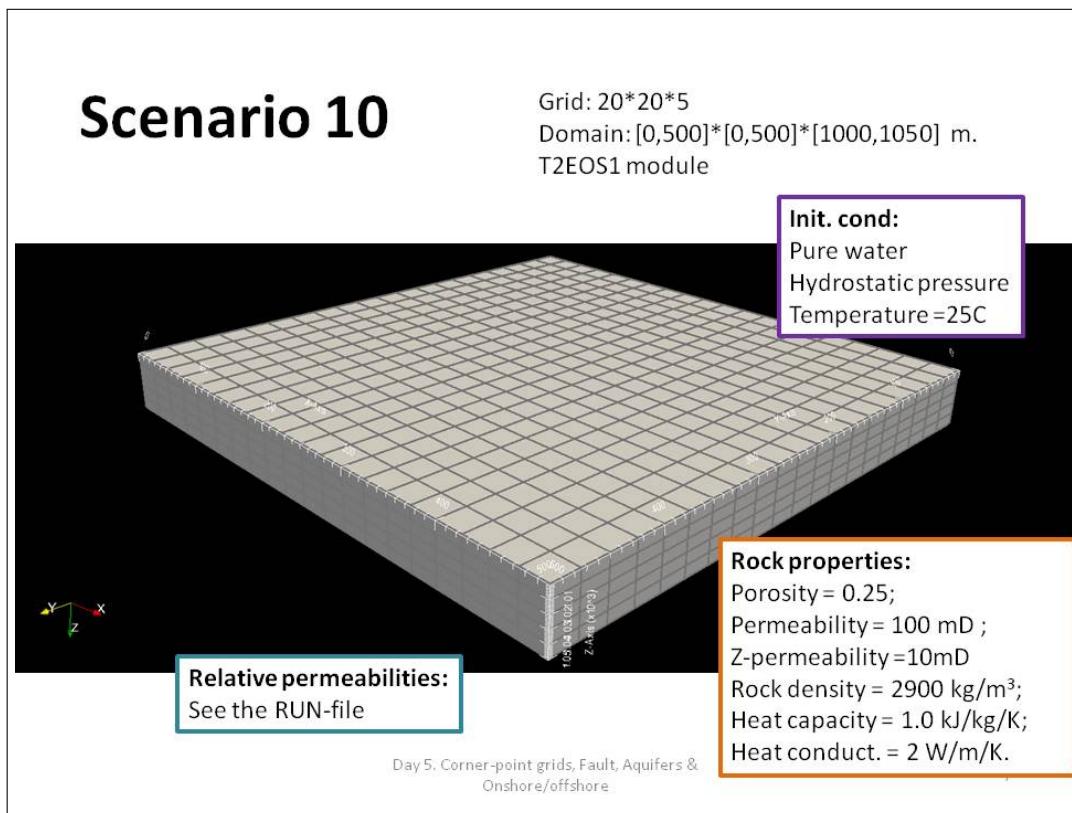
The simulator automatically converts the **Cartesian** and **Radial** grids into **Corner-Point** format within **MAKE-ENDMAKE** brackets. The corner-point grid can be exported by the **SAVECPG** keyword. The saved grid file can be later used in another simulation. The grid file can also be created/modified by a third-party software and be imported in the simulator by using **INCLUDE** keyword.

The **SAVECPG** keyword saves formatted grid file.

```
SAVECPG syntax
1 -- within MAKE-ENDMAKE brackets
2
3 SAVECPG
4   filename imin imax   jmin jmax   kmin kmax /
5
6 =====
7
8   filename - output file name
9   imin/imax - the boundaries of the box along i-index axis, for which the
10    grid file is saved.
11   jmin/jmax - the boundaries of the box along j-index axis, for which the
12    grid file is saved.
13   kmin/kmax - the boundaries of the box along k-index axis, for which the
14    grid file is saved.
```

Day 5. Corner-point grids, Fault, Aquifers &
Onshore/offshore

6



SIMULATIONS/SCENARIO10/0/SCENARIO10.RUN

```

1 RUNSPEC ##### RUNSPEC section begins here #####
2
3 HCROCK                                We enable heat conduction.
4
5 T2EOS1                                 We use T2EOS1 module
6
7 GRID ##### GRID section begins here #####
8
9     The grid is specified within brackets MAKE-ENDMAKE
10 MAKE <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
11 -- cartesian                            We select Cartesia gridding
12 -- grid      nx  ny  nz                option and specify the number of
13   CART       20  20   5   /               grid blocks along every axis.
14
15 XYZBOUND
16 -- xmin-xmax  ymin-ymax  zmin-zmax    we specify the domain extent.
17   0      500     0      500    1000 1050 / It is [0,500]*[0,50]*[1000,1050] meters
18
19 --INCLUDE                                This commented out keyword can be used to
20   'TASK1.INC' /                         include the answer to the exercises.
21
22 ENDMAKE >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
23
24 EQUALS                                  We specify uniform distributions:
25   PORO      0.25 /                     porosity = 0.25
26   PERMX     100  /                     X-permeability = 100 mD
27   HCOND CFX 2.   /                   Heat cond. coeff. = 2 W/m/K
28 /
29
30 COPY
31   PERMX      PERMY     /              We copy X-direction properties into
32   PERMX      PERMZ     /              Y- and Z-direction properties.
33   HCOND CFX HCOND CFY /             /
34   HCOND CFX HCOND CFX /             /
35 /
36
37 MULTIPLY
38   PERMZ  0.1 /                     Permeability ratio is 0.1.
39 /
40
41 RPTGRID                                 We define the output form the GRID sect.
42   PORO PERMX PERMZ /
43
44 PROPS ##### PROPS section begins here #####
45
46 ROCK properties are specified within brackets ROCK-ENDROCK
47 ROCK <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
```

```

96 POST      ##### POST section begins here #####
97
98 CONVERT          We convert the output to ParaView
99           compatible format.
100
101 END      #####

```

COORD keyword

COORD syntax

```

1 -- within brackets MAKE-ENDMAKE
2
3 COORD
4   xa1 ya1 za1  xb1 yb1 zb1 /
5   xa2 ya2 za2  xb2 yb2 zb2 /
6   xa3 ya3 za3  xb3 yb3 zb3 /
7 ...
8   xaN yaN zaN  xbN ybN zbN /
9 /
10 =====
11
12   xa#-ya#-zb# and - coordinates of two different points on a pillar
13   xb#-yb#-zb#
14
15           N - the total number of pillars in the current input box.
16           N=(imax-imin+2)*(jmax-jmin+2). The i-index is cycling
17           the fastest following by the j-index.

```

ZCORN keyword

```
1 -- within MAKE-ENDMAKE brackets
2
3 ZCORN
4   depth1 depth2 depth3 ... depthN /
5
6 =====
7
8   depth# - depth of a grid block corner.
9   N - the total number of the grid block corners in the current input
10    box. N=2*(imax-imin+1)*2*(jmax-jmin+1)*2*(kmax-kmin+1).
```

Day 5. Corner-point grids, Fault, Aquifers &
Onshore/offshore

10

Exercise

Exercise: Re-simulate scenario 10 exporting the grid in
Corner-point format. Open the grid file and inspect it.

Day 5. Corner-point grids, Fault, Aquifers &
Onshore/offshore

11

ADDZCORN keyword

Can be used to alter the corner depths in a box of grid blocks

```
-- within MAKE-ENDMAKE brackets
1
2
3 ADDZCORN
4   value1 imini imax1 jmini jmax1 kmini kmax1 m1_1 m2_1 m3_1 m4_1 m5_1 m6_1 /
5   value2 imin2 imax2 jmin2 jmax2 kmin2 kmax2 m1_2 m2_2 m3_2 m4_2 m5_2 m6_2 /
6   value3 imin3 imax3 jmin3 jmax3 kmin3 kmax3 m1_3 m2_3 m3_3 m4_3 m5_3 m6_3 /
7
8 ...
9
10 =====
11
12   value#      - the value added to ZCORN array in the input box;
13   imin#/imax# - the boundaries of the input box along i-indexation axis.
14   By default these values are equal to the arguments 1 and 2
15   of the keyword BOX.
16   jmin#/jmax# - the boundaries of the input box along j-indexation axis.
17   By default these values are equal to the arguments 3 and 4
18   of the keyword BOX.
19   kmin#/kmax# - the boundaries of the input box along k-indexation axis.
20   By default these values are equal to the arguments 5 and 6
21   of the keyword BOX.
22   mi_#        - (i=1,...,6). The mode:
23   'I-' - the operation is also applied to the adjacent face of
24   the grid block connected to the considered block in
25   the input box in the negative direction of i-index
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
```

Onshore/offshore

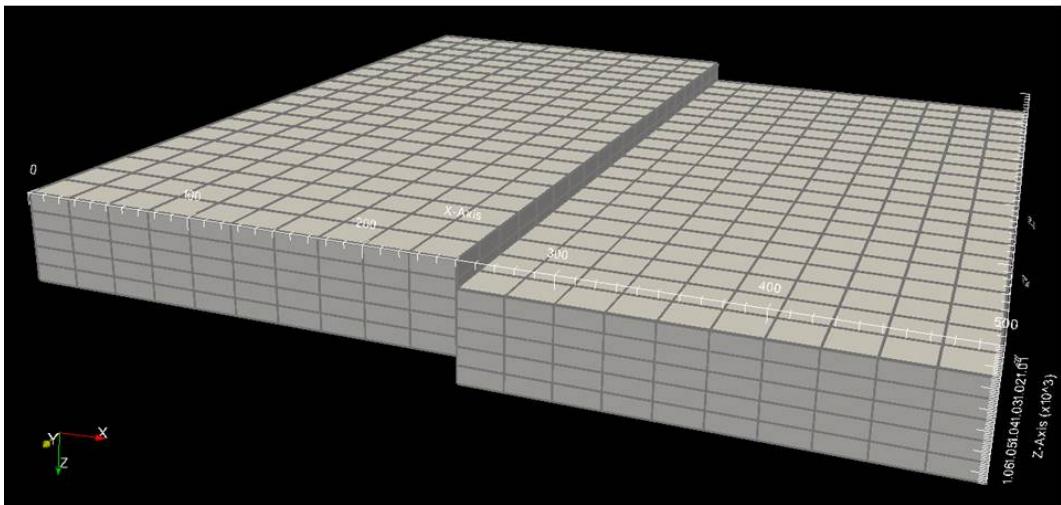
See also
EQLZCORN

See full
description
in the
Reference
manual

12

Exercise

Exercise: Create the following grid using ADDZCORN



Day 5. Corner-point grids, Fault, Aquifers &
Onshore/offshore

13

Answer

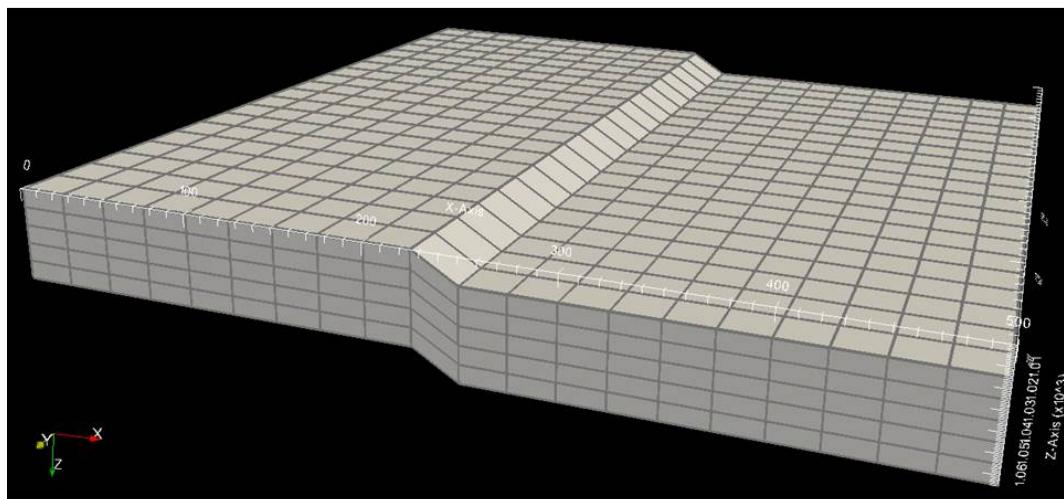
```
SIMULATIONS/SCENARIO10/0/TASK1.INC
1 ADDZCORN
2   15.0  11 20 4* 15.0 /
3 /
```

Day 5. Corner-point grids, Fault, Aquifers &
Onshore/offshore

14

Exercise

Exercise: Create the following grid using ADDZCORN



Day 5. Corner-point grids, Fault, Aquifers &
Onshore/offshore

15

Answer

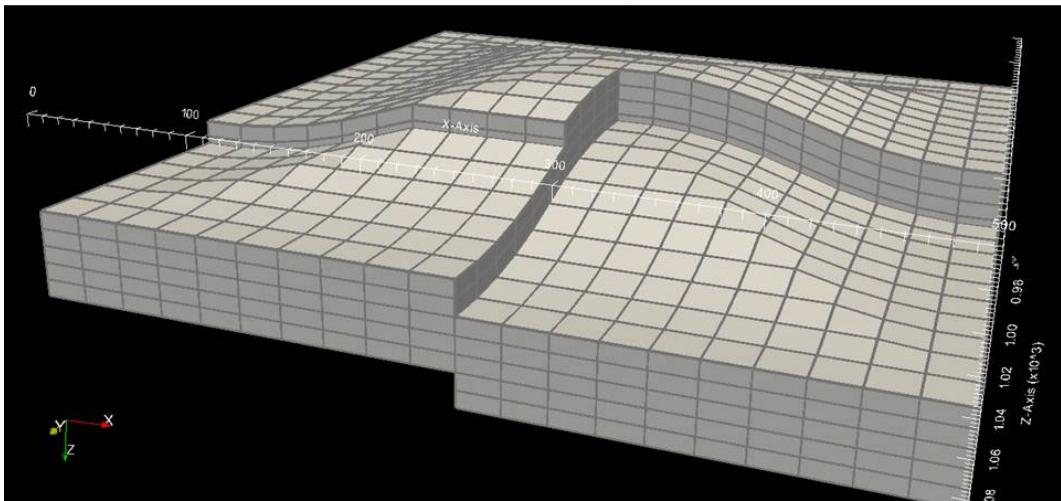
```
SIMULATIONS/SCENARIO10/0/TASK2.INC
1 ADDZCORN
2   15.0  11 20 4* 15.0  'I-' /
3 /
```

Day 5. Corner-point grids, Fault, Aquifers &
Onshore/offshore

16

Exercise

Exercise: Create the following grid using ADDZCORN



Day 5. Corner-point grids, Fault, Aquifers &
Onshore/offshore

17

Answer

```

1 ADDZCORN          SIMULATIONS/SCENARIO10/0/TASK3.INC
2 -2.0   3 18 3 18 2*  'I-' 'I+' 'J-' 'J+' /
3 -4.0   4 17 4 17 2*  'I-' 'I+' 'J-' 'J+' /
4 -6.0   5 16 5 16 2*  'I-' 'I+' 'J-' 'J+' /
5 -8.0   6 15 6 15 2*  'I-' 'I+' 'J-' 'J+' /
6 -8.0   7 14 7 14 2*  'I-' 'I+' 'J-' 'J+' /
7 -6.0   8 13 8 13 2*  'I-' 'I+' 'J-' 'J+' /
8 -4.0   9 12 9 12 2*  'I-' 'I+' 'J-' 'J+' /
9 -2.0  10 11 10 11 2*  'I-' 'I+' 'J-' 'J+' /
10 35.0 11 20 11 20 2* /
11 15.0 1 10 15 20 2* /
12 /

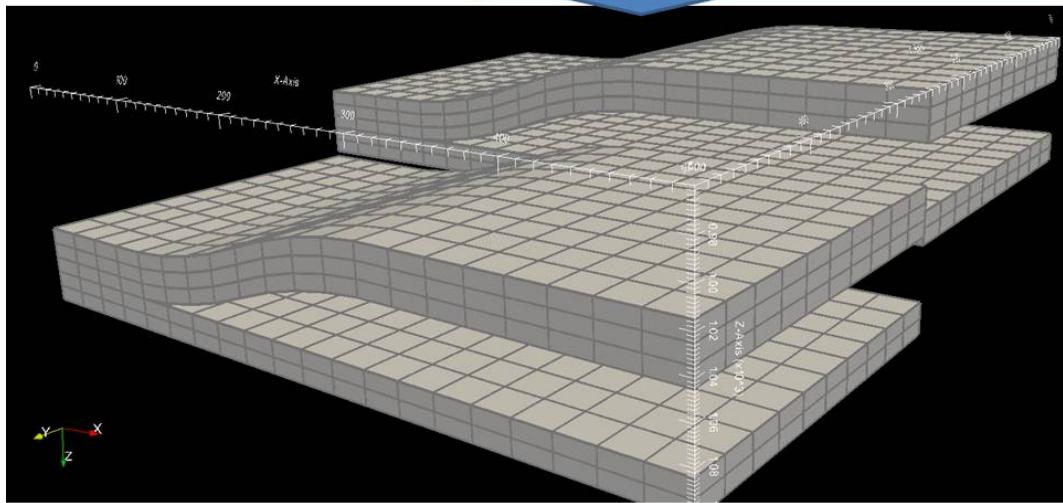
```

Day 5. Corner-point grids, Fault, Aquifers &
Onshore/offshore

18

Exercise

Exercise: Create the following grid using ADDZCORN



Day 5. Corner-point grids, Fault, Aquifers &
Onshore/offshore

19

Answer

```

1 ADDZCORN          SIMULATIONS/SCENARIO10/0/TASK4.INC
2   -2.0   6 20  2*   1 3  'I-' 'I+'
3   -4.0   7 30  2*   1 3  'I-' 'I+'
4   -6.0   8 20  2*   1 3  'I-' 'I+'
5   -8.0   9 20  2*   1 3  'I-' 'I+'
6   -8.0  10 20  2*   1 3  'I-' 'I+'
7   -6.0  11 20  2*   1 3  'I-' 'I+'
8   -4.0  12 20  2*   1 3  'I-' 'I+'
9   -2.0  13 20  2*   1 3  'I-' 'I+'
10  55.0   1 20  11 20 1 5 /
11 /

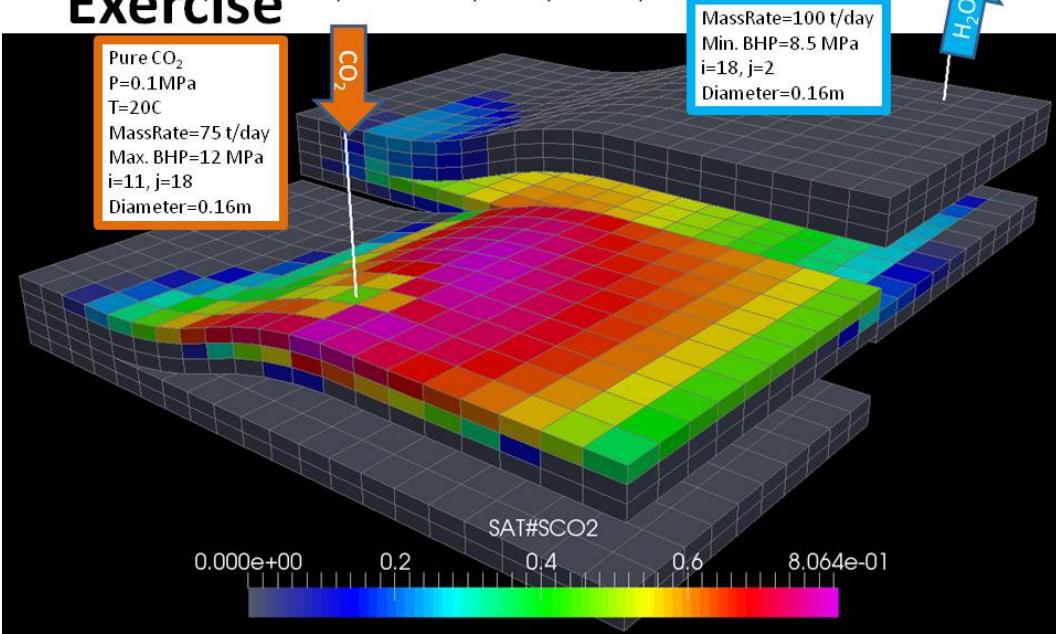
```

Day 5. Corner-point grids, Fault, Aquifers & Onshore/offshore

20

Exercise

Using the last grid simulate injection of CO₂ using BINMIXT module. There are two wells perforated in every layer. Report distributions up to 4000 days every 100 days.



Day 5. Corner-point grids, Fault, Aquifers & Onshore/offshore

21

RUN-file (scenario 10, Exercise)

1. Open RUN-file in text editor
 2. Run the simulation
 3. Open results in ParaView

Day 5. Corner-point grids, Fault, Aquifers & Onshore/offshore

22


```
121  
122 EQUALREG  
123   TEMPC 25 ROCKNUM 0 /      The initial temperature is 25 C  
124   COMP1T 0.0 /  
125 /  
126  
127 EQUALNAM          We inject pure CO2 which temperature at surface  
128   PRES 0.1 INJECTOR /      conditions (P=0.1 MPa) is 20 degrees of Celsius.  
129   TEMPC 20. /  
130   COMP1T 1.0 /  
131 /  
132  
133 RPTSUM  
134   PRES TEMPC PHST COMP1T SAT#LH20 SAT#SC02/  
135           We specify the properties saved at every  
136           report time.  
137  
138 SCHEDULE ##### SCHEDULE section begins here #####  
139  
140 WELLPROD          We define that PRODUCER target is  
141   PRODUCER OPEN MASS 1* 100. 8.5 / 100 tons/day. The minimal bottom-hole  
142 /           pressure is 8.5.  
143  
144 WELLINJE          The INJECTOT target is 75 tons/day.  
145   INJECTOR OPEN MASS 1* 75. 12.0 / The maximal bottom-hole pressure is  
146 /           12 MPa  
147  
148 TSTEP              We advance simulation to 4000 days reporting  
149   40*100 /           distributions every 100 days.  
150  
151 POST    ##### POST section begins here #####  
152  
153 CONVERT             We convert the output to ParaView  
154           compatible format.  
155  
156 END     #####
```

6.2. Faults

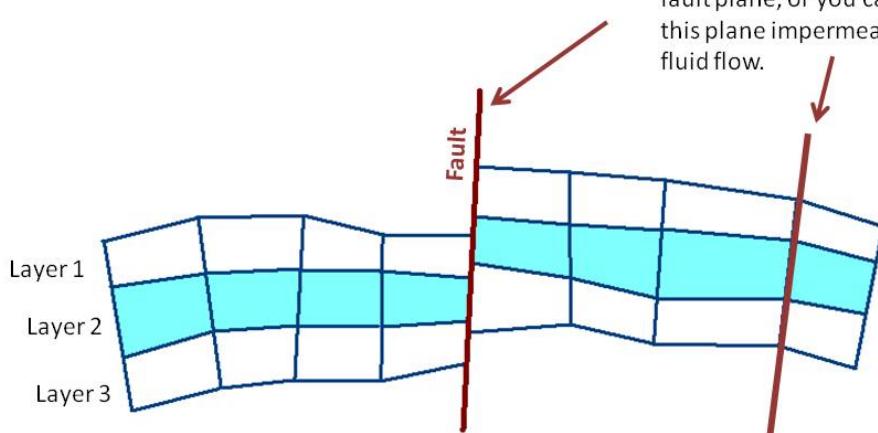
Faults

Day 5. Corner-point grids, Fault, Aquifers &
Onshore/offshore

23

Faults

Using the Faults option you
can reduce permeability
(transmissibility) across the
fault plane, or you can make
this plane impermeable for
fluid flow.



Day 5. Corner-point grids, Fault, Aquifers &
Onshore/offshore

24

FAULTS keyword

The fault faces are introduced using the FAULTS keyword.

```
FAULTS syntax
1 -- within MAKE-ENDMAKE brackets
2
3 FAULTS
4   name1 imin1 imax1 jmin1 jmax1 kmin1 kmax1 face1 /
5   name2 imin2 imax2 jmin2 jmax2 kmin2 kmax2 face2 /
6   name3 imin3 imax3 jmin3 jmax3 kmin3 kmax3 face3 /
7   ...
8 /
9
10 =====
11
12   name#      - a character ID of the fault;
13   imin#-imax# - the boundaries of the input box along i-indexation axis.
14   By default these values are equal to 'i' and the 2nd
15   argument of the keyword MAKE, respectively;
16   jmin#-jmax# - the boundaries of the input box along j-indexation axis.
17   By default these values are equal to 'i' and the 3rd
18   argument of the keyword MAKE, respectively;
19   kmin#-kmax# - the boundaries of the input box along k-indexation axis.
20   By default these values are equal to 'i' and the 4th
21   argument of the keyword MAKE, respectively;
22   face#      - fault face tag. Must be one of
23   'I-','X-' - fault face coincides with the grid block face
24   the negative direction of the i-index coordinate
25   line;
Day 5. Corner-point grids, Fault, Aquifers &
Onshore/offshore
```

See full
description
in the
Reference
manual

25

MULTFLT keyword

The transmissibility multipliers across the fault are introduced using the MULTFLT keyword.

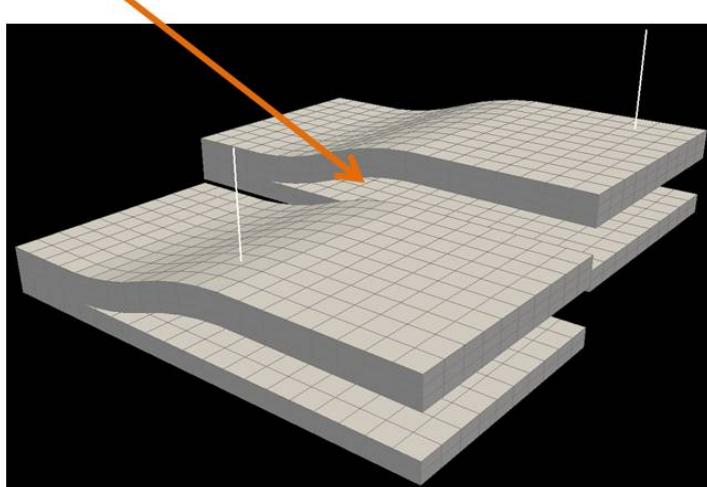
```
MULTFLT syntax
1 -- within MAKE-ENDMAKE brackets
2
3 FAULT
4   name1 multi /
5   name2 mult2 /
6   name3 mult3 /
7   ...
8 /
9
10 =====
11
12   name# - character ID of the fault;
13   mult# - transmissibility multiplier across the fault.
```

Day 5. Corner-point grids, Fault, Aquifers &
Onshore/offshore

26

Exercise

Re-simulate the last version of the Scenario 10 (with wells) making the Fault impermeable



Day 5. Corner-point grids, Fault, Aquifers &
Onshore/offshore

27

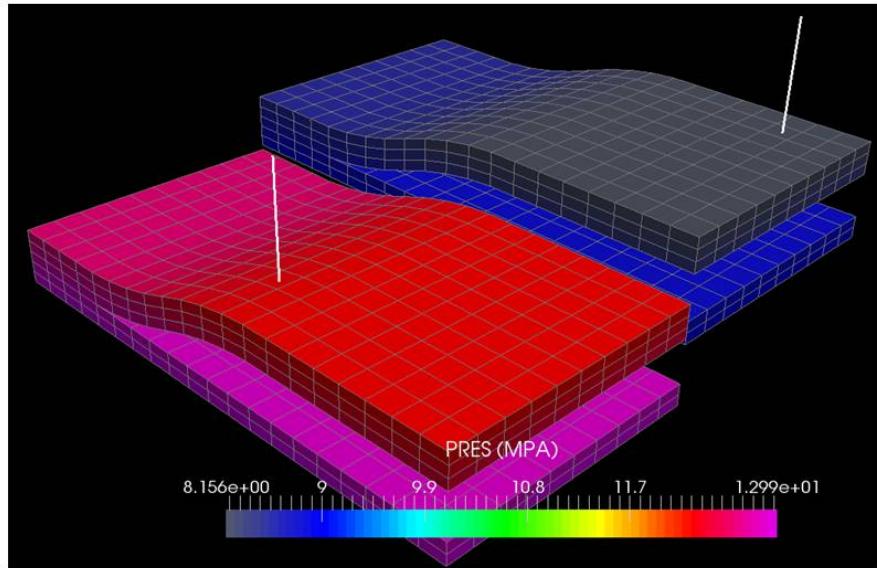
Answer

```
SIMULATIONS/SCENARIO10/EXERCISE/FAULTS.INC
1 -- within MAKE-ENDMAKE brackets
2
3 FAULTS
4   'MYFAULT' 1 20 10 10 1 5 'Y+' /
5 /
6
7 MULTFLT
8   'MYFAULT' 0.0 /
9 /
```

Day 5. Corner-point grids, Fault, Aquifers &
Onshore/offshore

28

Result of the simulation



Day 5. Corner-point grids, Fault, Aquifers &
Onshore/offshore

29

6.3. Modeling aquifers

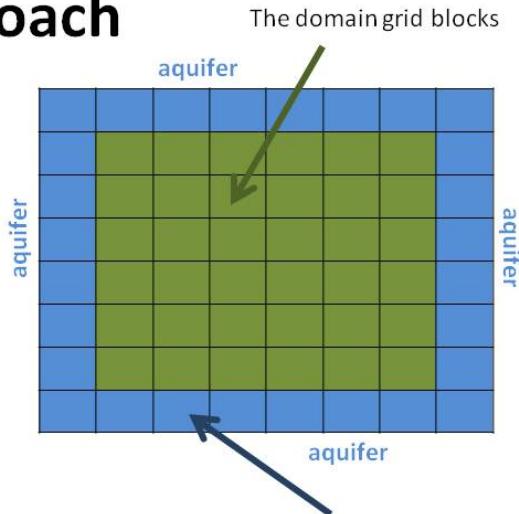
Modeling aquifers

Day 5. Corner-point grids, Fault, Aquifers & Onshore/offshore

30

Modeling approach

1. Encompass the domain in a circle of boundary grid blocks using SAMESIZE option. The grid blocks must be active.
2. Specify an estimate of initial distribution of pressure (and temperature).
3. Simulate the flow (without modeling injection/production) for a period of time until hydrostatic distribution forms.
4. Fix parameters in boundary grid blocks (set ACTNUM=2).
5. Farther, you can model the injection/production.



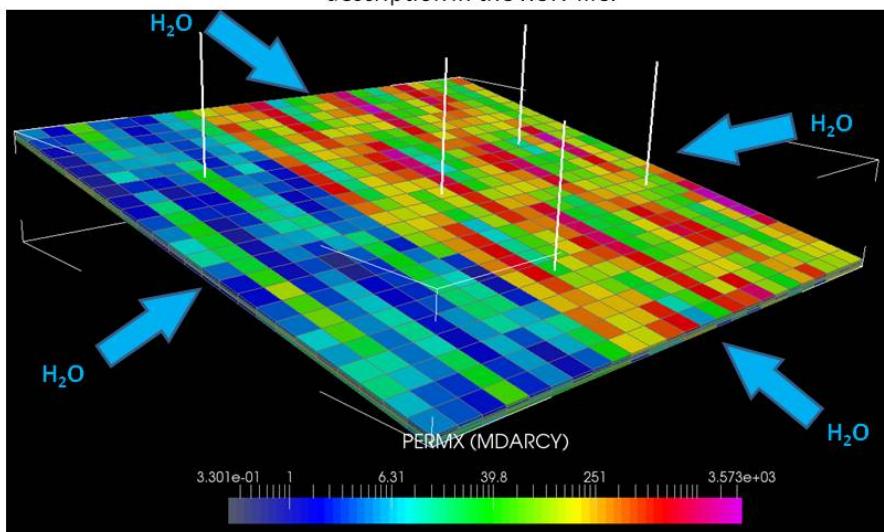
The boundary grid blocks used for modeling influx from the aquifer must be created using SAMESIZE option of the BOUNDARY keyword.

Day 5. Corner-point grids, Fault, Aquifers & Onshore/offshore

31

Scenario 11

Steam production via 5 production wells. The pressure in the field maintains by the fluid inflow from lateral boundaries (aquifer). The grid, porosity and permeability are loaded via grid file. See the problem description in the RUN-file.



Day 5. Corner-point grids, Fault, Aquifers &
Onshore/offshore

32

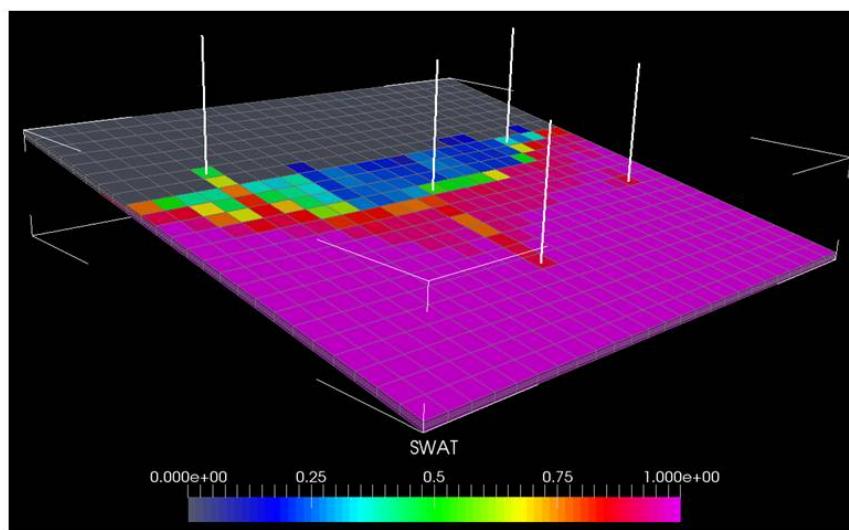
RUN-file (scenario 11)

1. Open RUN-file in text editor
2. Run the simulation
3. Open results in ParaView

Day 5. Corner-point grids, Fault, Aquifers &
Onshore/offshore

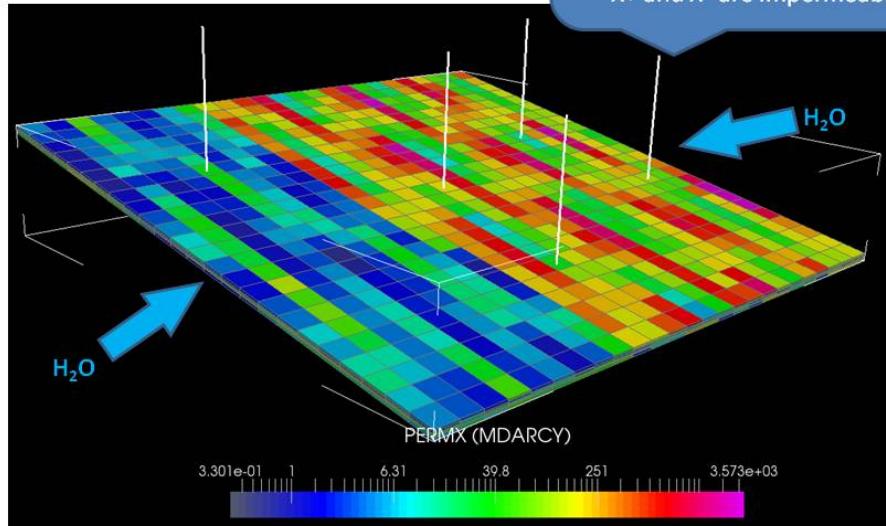
33

Scenario 11 (result)



Scenario 11 (exercise)

Exercise: Re-simulate scenario 11 supposing that the fluid can flow into the reservoir only through Y+ and Y- lateral boundaries, whereas X+ and X- are impermeable.



Day 5. Corner-point grids, Fault, Aquifers & Onshore/offshore

35

6.4. Onshore/offshore boundary conditions

Onshore/offshore boundary conditions

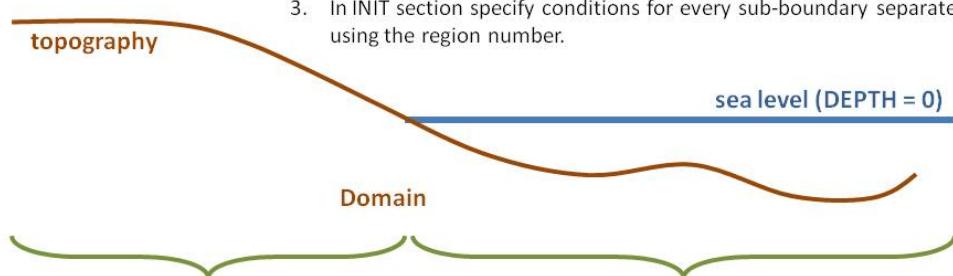
Day 5. Corner-point grids, Fault, Aquifers & Onshore/offshore

36

Onshore/offshore conditions

To model the boundary conditions:

1. Create the boundary grid blocks for the top surface of the domain using the BOUNDARY keyword;
2. Divide the boundary into two sub-boundaries (onshore and offshore) using the DEPTH property. Use a region number (e.g., INCONUM) to distinguish the sub-boundaries.
3. In INIT section specify conditions for every sub-boundary separately using the region number.

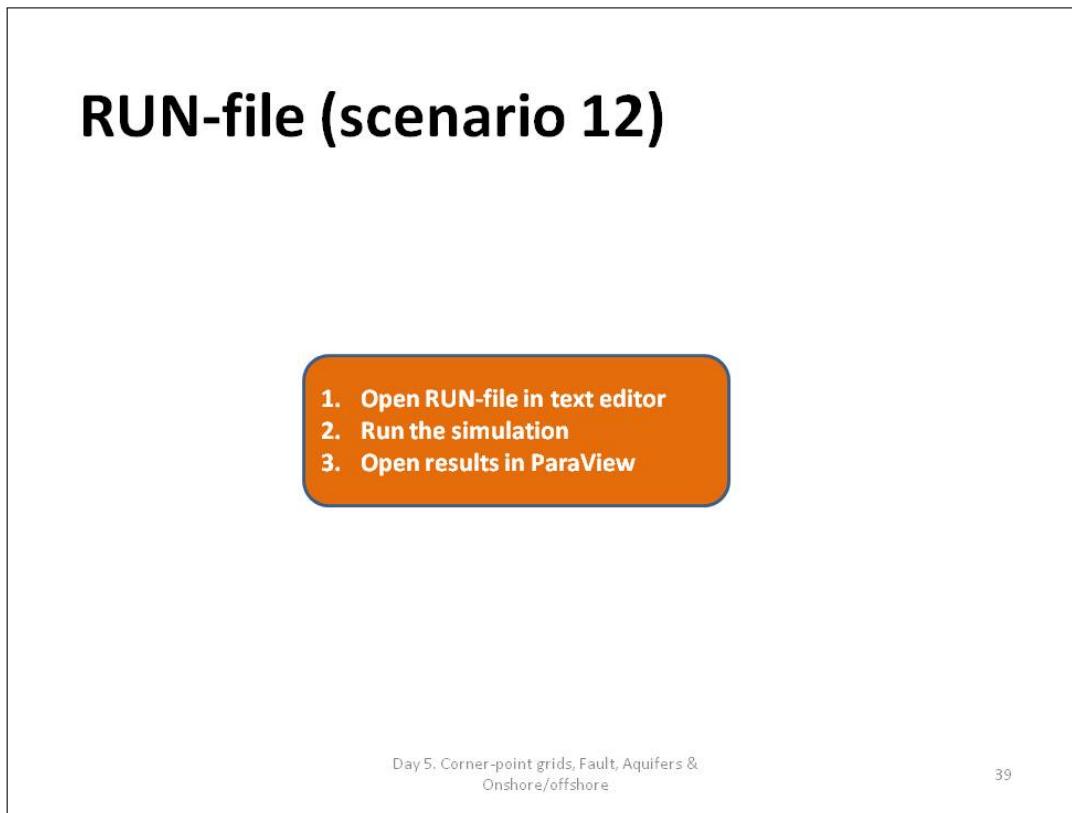
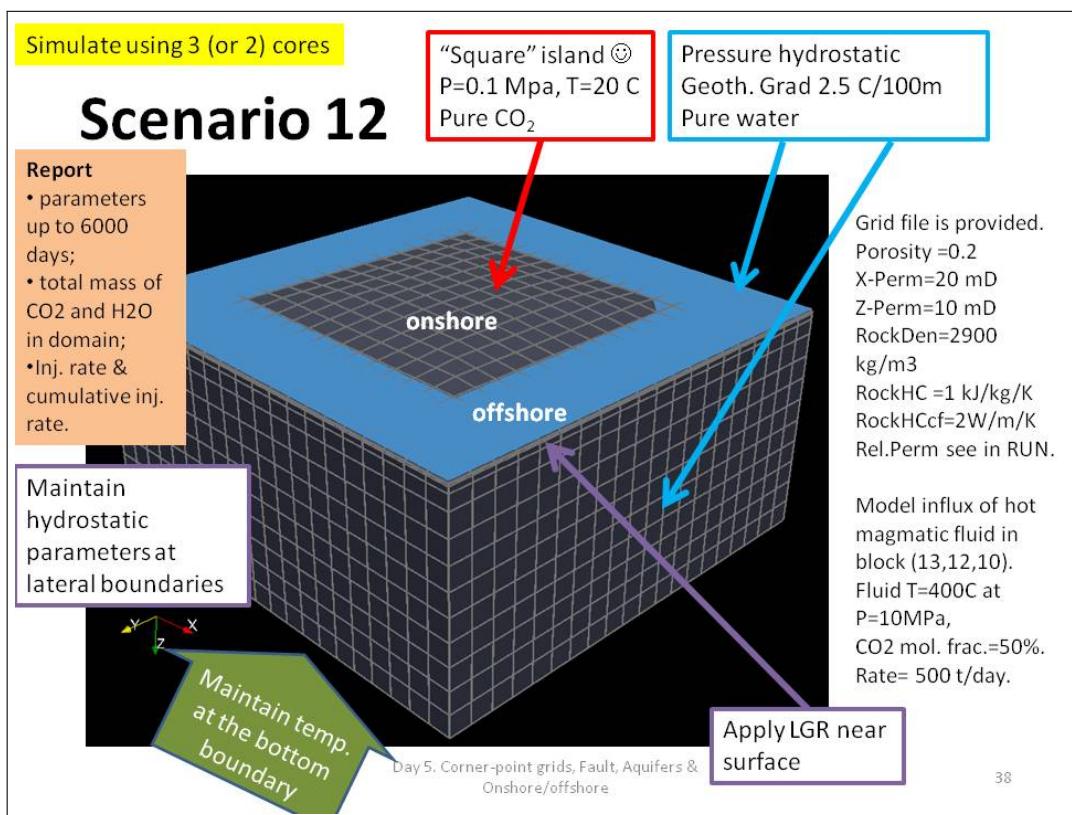


At the top surface of the domain: air (we use CO₂ instead), P=1 Bar, T=20°C

At the top surface of the domain: water, P = hydrostatic in water, T=20°C (or other value).

Day 5. Corner-point grids, Fault, Aquifers & Onshore/offshore

37




```

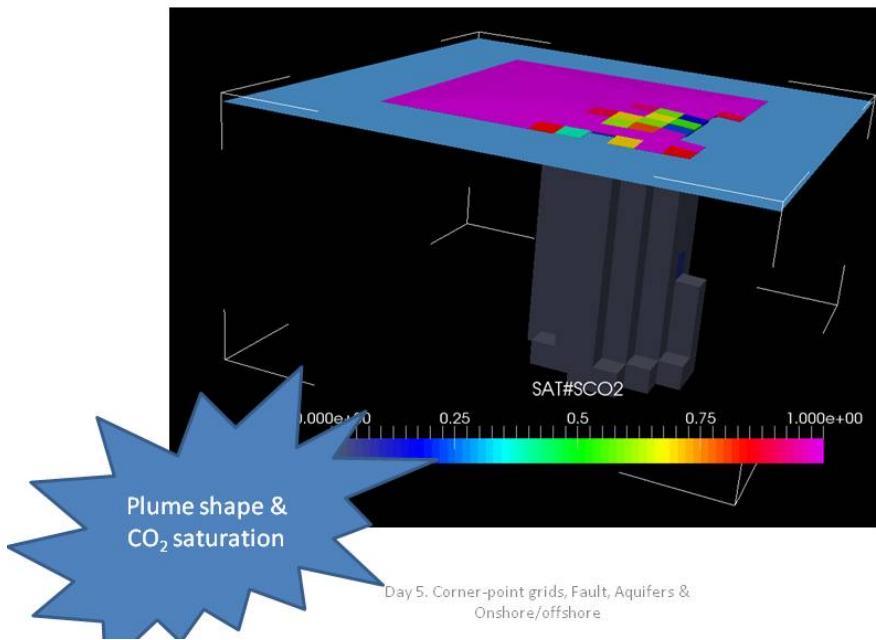
96
97 INIT ##### INIT section begins here #####
98
99 REGALL We enable application of the following three keywords both to
100 domain grid blocks and boundary grid blocks.
101
102 ECHO We enable more output in LOG-file (the number of cells modified
103 by every of the following operation is outputted in LOG).
104
105 OPERAREG We note specify INCONUM=1
106 INCONUM DEPTH SATNUM 0 SETEXT 1 0 10000. / in all blocks above sea level.
107 / All other blocks (below sea
108 level) are marked as
109 INCONUM=0, by default.
110
111 OPERAREG We specify parameters in blocks below sea level:
112 PRES DEPTH INCONUM 0 MULTA 0.1 0.01 / PRES=0.1+0.01*DEPTH
113 TEMPC DEPTH INCONUM 0 MULTA 20 0.025 / TEMPC=20+0.025*DEPTH
114 COMP1T 1* INCONUM 0 EQUALS 0.0 / Pure water (no CO2)
115 /
116
117 We specify parameters in blocks above sea level:
118 EQUALREG
119 PRES 0.1 INCONUM 1 / Pressure =0.1 MPa
120 TEMPC 20 / Temperature=20C
121 COMP1T 1.0 / Pure CO2 (we model CO2 atmosphere)
122 /
123
124 EQUALNAM We specify parameters of magmatic source
125 PRES 10. 'MAGMASRC' / The injected fluid temerature is 400 C
126 TEMPC 400. / at P=10MPa. The fluid molar composition
127 COMP1T 0.5 / is 50% CO2 and 50% H2O
128 /
129
130 NOECHO We disable additional output in LOG,
131
132 RPTSUM We report the following parameters
133 PRES TEMPC PHST COMP1T COMP2T SAT#SCO2 SAT#LH2O /
134
135 RPTSUM and also these parameters for domain.
136 I-IJKRES J-IJKRES K-IJKRES /
137
138 RPTSRC We report these parameters for the
139 SMIR#1 SMIR#2 SMIT#1 SMIT#2 /
140
141 EQUALS We specify FIPNUM=1 in the domain.
142 FIPNUM 1 /
143 /

```



```
192 RPTPOST
193     NOTHING TIME MASS#1 MASS#2 /
194 POSTFPCE                                     Here, we export consolidated file
195     1 /                                         for FIPNUM=1 region.
196 /
197
198 END      #####
```

Results (scenario 12)



Thank you!

References

1. MUFITS reservoir simulator website. <http://www.mufits.imec.msu.ru>.
2. ParaView website. <http://www.paraview.org>.