

IORSim User Guide

Version 1.0

*Jan Nossen, IFE
Jan Sagen, IFE
Aksel Hiorth, UiS
Børre Antonsen*

November 2021

1. Contents

1. Contents.....	2
1. What is IORSim.....	3
2. Running IORSim in forward mode	4
3. Modifications of Eclipse .DATA files for compatibility with IORSim	4
3.1. Modifications in the RUNSPEC section	5
3.2. Modifications in the GRID section	5
3.3. Modifications in the SOLUTION section	5
3.4. Modifications in the SCHEDULE section	6
4. IORSim input file	7
4.1. Units	7
4.2. Input file structure and general syntax	8
4.3. Description of keywords – General input part1	10
4.3.1. *WELLMODEL	optional 10
4.3.2. *TEMPERATURE	optional 10
4.3.3. *GRIDPLOT_WRITE	optional 10
4.3.4. *REACTING_SYSTEM	optional 11
4.3.5. *TRACER_LGR	mandatory 11
4.3.6. *INTEGRATION	mandatory 12
4.4. Description of keywords for modelling tracer, silicate and geochemistry.....	13
4.4.1. *INTEGRATE_SPECIES	mandatory 13
4.4.2. *MODELTYPE	mandatory 14
4.4.3. *SPECIES	mandatory for Tracer and Silicate, optional repeat 14
4.4.4. *SOLUTION	mandatory for GeoChem, optional repeat 14
4.4.5. *SATNUM	optional 15
4.4.6. *MODELTEMPLATE	mandatory 16
4.4.7. *TINIT	mandatory 16
4.4.8. *COMP	16
4.4.9. *CHEMFILE	17
4.4.10. *KVALWAT *KVALOIL *KVALGAS.....	17
4.4.11. *CO2	17

4.4.12. *MODELINSTANCE	18
4.4.13. *WELLSPECIES	18
4.5. Description of keywords – General input part II	19
4.5.1. *OUTPUT	19
4.5.2. *WELLPLOT INTERVAL	19
4.5.3. *PRINTLEVEL	20
4.5.4. *END	20
5. GeoChem input file	21
5.1.1. add_species	21
5.1.2. complex	23
5.1.3. equilibrium_phases	23
5.1.4. iexchange	24
5.1.5. rate	26
5.1.6. solution	27
6. Output files	28
6.1. Grid concentration files	28
6.1.1. Example of IORSIM_PLOT_*.FUNRST file	28
6.1.2. Information contained in the IORSim_*.FUNRST file	30
7. Example cases	31
7.1. Tracer: CASE.trcinp file	31
7.2. Geochem: CASE.trcinp file and geochem input files	33
7.2.1. CASE.trcimp file	33
7.2.2. geochem1 file (reservoir region 1)	36
7.2.3. geochem2 file (reservoir region 2)	37
7.3. Silicate case	38
8. References	40

1. What is IORSim

IORSim is a simulator that can be used as an add-on to industrial reservoir simulators like Eclipse. The current version focuses on Eclipse. IORSim can be run in two modes:

- Forward mode, or one-way coupling. In this case, IORSim is used as a postprocessor to the reservoir simulator. IORSim reads the Eclipse grid, restart and well rate files and can track the transport and reactions of chemical species, including tracers, ions and diverters through the reservoir. These simulations are normally reasonably fast, and often faster than the Eclipse simulation itself.
- Backward mode, or two-way coupling. In this case, IORSim reads the Eclipse output files as above at each time, runs one step and uses the new solution to generate files containing saturation table numbers (SATNUM) that are read by Eclipse using the Eclipse keyword READDATA. Eclipse then runs one time step using the updated SATNUM values to modify relative permeability curves and generates new output files read by IORSim and so on. The backward mode is activated using either a Python script or a new IORSim launcher GUI in development. The backward coupling is more complex than the forward version and leads to substantially longer simulation times in most cases.

In most cases, existing Eclipse cases can be run with IORSim with only minor modifications, as outlined in the following. Limited testing of IORSim has been done on very large and complex cases.

2. Running IORSim in forward mode

IORSim can be started from the command line as follows:

IORSim CASE

where CASE is the root name of an Eclipse case. Prior to running IORSim, a set of valid Eclipse files must be generated, containing appropriate data for running IORSim. The necessary Eclipse files (formatted or unformatted) are: CASE.INIT, CASE.EGRID, CASE.UNRST and CASE.RFT. A description of the Eclipse keywords necessary to obtain sufficient data to run IORSim is given below in Section 2.

3. Modifications of Eclipse .DATA files for compatibility with IORSim

This section describes the Eclipse keywords necessary to obtain the required data in the Eclipse output files needed by IORSim.

3.1.Modifications in the RUNSPEC section

IORSim is designed to work with three-phase cases. Two-phase oil-water cases can be handled by introducing a dummy gas phase and setting gas saturations to zero (e.g. by putting the gas-oil contact above the top of the reservoir and $K_{rg} = 0$ at $S_g = 0$). For details on how to define a three-phase case in Eclipse, we refer to the Eclipse manual.

IORSim requires unified restart files, i.e. keyword UNIFOUT should be specified in the Eclipse input file. In some cases (e.g. for coupling to the EnKF methodology) it may be convenient to use separate restart files. For these cases an auxiliary program has been made that concatenates two or more separate restart files into a unified restart file that IORSim can use see (Section A.3).

3.2.Modifications in the GRID section

GRIDFILE:

The (minimum) requirement for Eclipse's GRIDFILE option to run IORSim is:

```
GRIDFILE 0 1  
/
```

This is the default from Eclipse, i.e. the default gives enough information for IORSim. Note, however, that if the Eclipse keyword NOGGF is specified in the Eclipse input file, no .GRID or .EGRID files is output, i.e. make sure that NOGGF is not specified in the case.

INIT:

The requirement for Eclipse's INIT option to run IORSim is:

```
INIT
```

This requests output of an INIT file (CASE.INIT).

3.3.Modifications in the SOLUTION section

RPTRST:

If the time zero is to be included in the IORSim simulation, a RPTRST keyword must be present in Eclipse's SOLUTION section and accompanied by an initial WRFTPLT keyword in the SCHEDULE section. If time zero is not wanted in the IORSim simulation, no RPTRST keyword should appear

in this section. The format required for RPTRST and WRFTPLT keywords to enable an IORSim simulation is given below in Section 2.4.

3.4.Modifications in the SCHEDULE section

RPTRST:

The minimal requirement for Eclipse's RPTRST option to run IORSim is:

```
RPTRST
BASIC=2
ALLPROPS=2
FLOWS
RSSAT
/
```

For details on Eclipse's RPTRST keyword we refer to the Eclipse manual, but for convenience a brief summary is given here:

BASIC=2: controls the output frequency, 2 gives a report every report time (specified using TSTEP or DATES).

ALLPROPS=2: ensures that formation volume factors given in the "old" format (1/FVFOIL, 1/FVFGAS, 1/FVFWAT) are used in the output.

FLOWS: yields output of inter-block flows, including non-neighbour connection flows.

RSSAT: yields output of saturated Rs values (dissolved gas-oil ratio).

NORST=1: exclude arrays required for restarting (but not for IORSim). May reduce Eclipse restart files significantly.

Note:

Beware that RPTSCHED and RPTSOL can also affect the output to restart files. From Eclipse's documentation:

"Output to the RESTART file may also be controlled by using the mnemonic RESTART (switch 7) of the RPTSCHED and RPTSOL keywords. However, RPTRST allows greater flexibility when specifying extra data to be written to the RESTART file. If the first control of RPTRST (or mnemonic BASIC) is set to 1 or 2, subsequent use of RPTSCHED will override the output frequency. However, if the first control (or mnemonic BASIC) is set to 3 or more, then any value of 7th switch of RPTSCHED will be ignored." In other words, the Eclipse input files should be checked for RPTSCHED and RPTSOL, and these be removed (or output frequency set equal to that in RPTRST.

WRFTPLT:

The minimal requirement for Eclipse's WRFTPLT keyword to run IORSim is:

```
WRFTPLT  
'*' NO REPT /  
/
```

For details on Eclipse's WRFTPLT keyword we refer to the Eclipse manual, but for convenience a brief summary is given here:

'*': requests that the .RFT file should include information about all defined wells

NO: has no effect for IORSim, YES works equally well, but adds data not required for an IORSim simulation.

REPT: requests output of the current PLT data (describing the flows through the well connections) for all wells at this time and at all subsequent report times, while they are open or stopped.

4. IORSim input file

4.1.Units

IORSim simulations are performed and reported in SI-units, except for time, which is given and reported in days. IORSim can work with output files reported in Eclipse's lab, field or metric unit system. In all cases, the Eclipse output is converted to SI-units by IORSim. Species concentrations in injection wells (keyword *CONC INJECTION) are given at standard conditions. Production well concentrations given in the .trcconc and .trwlrres files (described in Section 4.1 and Section 4.2) are given at standard conditions. Grid concentrations in the gridplot files requested by the optional keyword *GRIDPLOT_WRITE are given at reservoir conditions (cf. Section 4.4 for an example). Initial concentrations, given by keyword *C INIT, are given at reservoir conditions.

4.2. Input file structure and general syntax

IORSim reads input from an input file CASE.trcinp, where the base name CASE refers to the root name of the Eclipse case.

The syntax of IORSim input files is as follows:

*keyword argument(s)

where *keyword is any of the keywords recognized by IORSim (see below for a description of individual keywords). The argument(s) can be given on the same line as the *keyword, or on separate lines, if there are multiple arguments to a keyword, they can be separated by space(es) or line(s) (i.e. no distinction between lines and spaces). There is no distinction between uppercase and lowercase in keywords or arguments. Comments can be added by preceding them by a #. The effect of # is that the remainder of the line is ignored. Upper or lowercase characters can be used in the input, but are not distinguished, as IORSim converts input from lowercase to uppercase.

The order of the keywords is fixed. Some keywords are optional, among these are:

*WELLMODEL *TEMPERATURE *GRIDPLOT_WRITE *REACTING_SYSTEM *SATNUM
*CO2 *KVAWAT *KVALOIL *KVALGAS *INJECTOR *TIME *PRODUCER

The order of the keywords are as follows, where the optional keywords for the coupled and sequential solver respectively are marked in **green colour** (only in the table below) and the mandatory keywords are marked in **black bold font**:

*WELLMODEL
*TEMPERATURE
*GRIDPLOT_WRITE

*REACTING_SYSTEM
*TRACER_LGR
*INTEGRATION
*INTEGRATE_SPECIES
*MODELTYPE
*SPECIES / *SOLUTION (for Geochem runs)
*SATNUM


```
*MODELTEMPLATE
*TINIT
*COMP
*CHEMFILE                (for Geochem runs)
*CO2
*KVALWAT
*KVALOIL
*KVALGAS
*MODELINSTANCE
*WELLSPECIES
*INJECTOR
*TIME
*OUTPUT
*PRODUCER
*WELLPLOT_INTERVAL
*PRINTLEVEL
*END
```

4.3. Description of keywords – General input part1

In the following sections the input keywords in the IORSim input file CASE.trcinp are described.

4.3.1. *WELLMODEL optional

Syntax: *WELLMODEL

When using the *WELLMODEL keyword, IORSim calculates species production from each perforation zone in the production wells. The mass rates from each zone are added together and give a total mass rate for the production well in question. After this, the water/oil/gas volume rates at standard conditions are applied to calculate the species concentration in the wellstream at the top of the well.

The disadvantage of this method is that zones with negative production rates (injection zones) are ignored. When using the *WELLMODEL keyword, the total mass balances of species inside the well pipe and into and out of each perforation zone are calculated. This method also covers the situations of crossflow that occur when wells are closing.

In IORSim, the *WELLMODEL calculation is done by discretizing the well into pipe blocks. The flow in the well pipe and into and out of the pipe blocks are calculated based on the perforation rates given by Eclipse. The species are then tracked through the ordinary reservoir blocks, and through the well pipe blocks towards the top of the well, where they are produced.

4.3.2. *TEMPERATURE optional

Syntax: *TEMPERATURE

Solve temperature calculation (recommended).

4.3.3. *GRIDPLOT_WRITE optional

The *GRIDPLOT_WRITE keyword triggers writing of a grid concentration file. The name of the restart file (Eclipse format) is IORSIM_PLOT_CASE.FUNRST and is a formatted grid concentration file. CASE in the filename is the name of the Eclipse input data file without the .DATA extension.

This file contains time dependent concentration fields (i.e. concentration values in the grid blocks) for all species defined in a case. The concentrations are written to a file compatible with Eclipse's unified restart file format (.FUNRST). This file can be displayed using software compatible with Eclipse's unified restart file format, e.g. FloViz.

The file contains concentrations recorded for all time steps. An example file is given in Section 4.4.

Units: The concentrations in the grid concentration files are given at reservoir conditions, the unit is amount=m³, where the amount unit is freely chosen by the user (i.e. Bq, mg, g, kg etc.).

4.3.4. *REACTING_SYSTEM

optional

This keywords flags that the case is a Single Well Chemical Tracer case for simulating SWCTT well tests

4.3.5. *TRACER_LGR

mandatory

Syntax:

```
N_lgr
I1 I2 J1 J2 K1 K2 NXref NYref NZref
. . . . .
. . . . .
I1 I2 J1 J2 K1 K2 NXref NYref NZref
```

This defines species grid refinement. The first argument is an integer that defines the number of areas that will be refined. If the first argument is 0, no grid refinement is performed.

If the first argument (N_lgr) is a positive integer, N_lgr x 9 integers must be given. In these groups of 9 integers, the six first (I1, I2, J1, J2, K1, K2) denote the boundary of the refined block, with lower and upper x-direction grid blocks given first, then lower and upper y-direction grid blocks and lower and upper z-direction grid blocks given at last. The last three numbers in the group of 9 integers denote the degree of refinement in x-direction, y-direction and z- direction, respectively. The last three integers must be odd numbers (NXref, NYref, NZref = 1,3,5,...). A number 1 gives the original grid block size, i.e. no refinement.

Adjacent grid refinement regions are allowed, e.g. cases as illustrated in Figure 1a and b. The adjacent regions can have equal degree of refinement or different degrees of refinement.

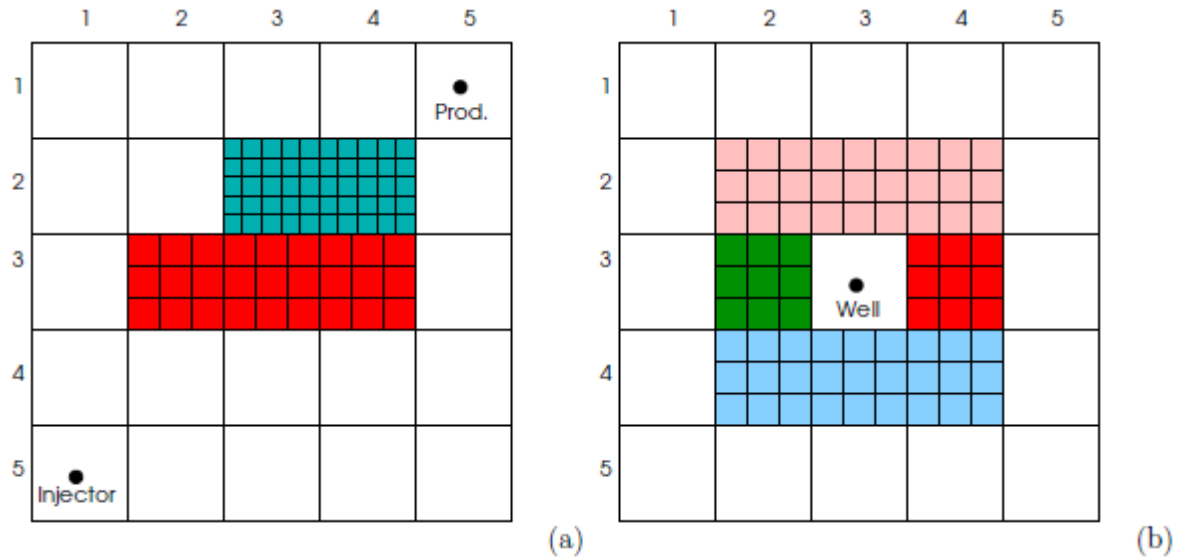


Figure 1: Grid refinement in IORSim. Adjacent regions are allowed, with equal or different degrees of refinement.

Limitations: In the current IORSim version, grid refinement is not allowed for blocks containing wells. Areas around wells can be refined using the method illustrated in Figure 1b. Note, however, that well blocks are dynamically allocated in Eclipse, and there is no way to guarantee that a previously unperforated block will not be perforated at a later time. IORSim does not give any warnings in such cases.

The grid refinement (i.e. if the first argument in the keyword `*TRACER_LGR` is larger than zero) is incompatible with keywords `*GRIDPLOT_WRITE` and `*RESTART_WRITE`. Refined regions cannot overlap. A warning message is issued if this occurs, and IORSim stops.

4.3.6. *INTEGRATION

mandatory

Syntax:

```
*INTEGRATION
Tstart Tstop
Dtmin Dtmax
Dtecl Dteclmax
Metnum
Solver
```

This keyword defines the methods used to integrate the IOR species forward in time. The arguments have the following meaning:

Tstart: Start time of the simulation. If Tstart is set to zero, the integration starts at the first time step found in the Eclipse restart file. Values larger than the first time step found in the

Eclipse restart file can be set if this is convenient, e.g. to avoid calculation of species concentrations before any species have been injected.

Tstop: Stop time of the simulation. Should be set to a value equal or larger than the largest value in the Eclipse restart file, unless the user wants to end the simulation earlier. Setting a very large value (e.g. 1e99) ensures that the IORSim simulation runs until the end of the Eclipse case.

Dtmin: Minimum time step size.

Dtmax: Maximum time step size.

Dtecl: Eclipse time step (for two-way coupling)

Dteclmax: Eclipse maximum time step (for two-way coupling)

Metnum: Choice of integration method. In the present version the argument must have the value 0, which chooses an upwind method.

Solver: Chooses solver. The argument takes one of three possible values; profile chooses a direct Gaussian solver (not suitable for large cases); sparskit2 chooses the iterative sparskit2 solver and linsolver chooses an in-house iterative solver. sparskit2 is the recommended choice for large cases, profile is mostly used for testing on small cases and linsolver is used as an alternative solver for verification purposes.

Units:

Tstart, Tstop, Dtmin, Dtmax: days

4.4. Description of keywords for modelling tracer, silicate and geochemistry

4.4.1. *INTEGRATE_SPECIES

mandatory

Syntax: * INTEGRATE_SPECIES

Activates integration of chemical species in IORSim (tracer, silicate, geochemistry)

4.4.2. *MODELTYPE

mandatory

Syntax: *MODELTYPE <model type>

The model type is either GeoChem, SILICATE or Tracer. GeoChem gives the full GeoChem geochemical model, which can handle most common ions in formation water or smart water. SILICATE gives a simplified geochemical model that is tailored for silicate injection for plugging high permeability zones. Tracer is for tracking tracer species in the reservoir.

Activates integration of chemical species in IORSim (tracer, silicate, geochemistry)

4.4.3. *SPECIES

mandatory for Tracer and Silicate, optional repeat

Syntax:

*SPECIES Species1

*SPECIES Species2

⋮

*SPECIES SpeciesN

4.4.4. *SOLUTION

mandatory for GeoChem, optional repeat

Syntax:

*SOLUTION <name>

<SpecieName1> <Concentration1>

⋮

<SpecieNameN> <ConcentrationN>

The keyword *SOLUTION is repeated for each wanted water solution (composition)

The Solution name can be referred as initial condition (4.4.8 *COMP) or as injected composition in wells.

4.4.5. *SATNUM

optional

If the *SATNUM keyword is present, a Satnum file for backward coupling to Eclipse, will be written. If *SATNUM is not present, a forward run is assumed, and no Satnum file is written.

If present, the following input must always be added after the *SATNUM keyword:

```
compName ilow ihigh clow chigh
```

This input line is associated with the backward coupled mode.

compName is component name used for calculating the satnum values transferred to Eclipse

For the GeoChem model we have:

ilow is the lower bound of the range of SATNUM indices

ihigh is the upper bound of the range of SATNUM indices

clow and chigh are the corresponding concentrations of the specie Mg

The SATNUM index is then used to interpolate between relperm curves based on the interpolation

$$irelperm = ilow + (ihigh - ilow) (C_{compName} - clow) / (chigh - clow)$$

For the Silicate model, we have:

Ilow is not in use

Ihigh is the number of rel. perm. curves, which is upper bound of the range of SATNUM

clow must be inputted as a parameter mult between 0 and 1, where $r = \log(K_{mult}/mult)$ in the code becomes the satnum value returned to Eclipse

chigh is not in use

4.4.6. *MODELTEMPLATE

mandatory

Syntax:

```
*MODELTEMPLATE <Template name>
```

The template name is an arbitrary word. It cannot contain spaces.

4.4.7. *TINIT

mandatory

Syntax:

```
*TINIT <Initial reservoir temperature>
```

```
<Water heat capacity> <Oil heat capacity> <Gas heat capacity>  
<Rock heat capacity>  
<Water density> <Oil density> <Gas density> <Rock density>
```

The first argument of TINIT is a uniform initial reservoir temperature.

The heat capacities are (mass/volume?) based.

The densities are in SI units (kg/m³).

4.4.8. *COMP

Syntax:

```
*COMP <Initial composition of species>
```

where <Initial composition of species> is <C1 C2 . . . Cn> for
tracer and silicate cases

and

<Initial composition of species> is <solution_name> for geochem
cases

Note that <solution_name> must be one of the solution names
defined in section 4.4.4 *SOLUTION

Units:

Amount (arbitrary)

4.4.9. *CHEMFILE

Syntax:

```
*COMP <ratefile>
```

where <ratefile> is the filename for the geochemical input file

4.4.10. *KVALWAT *KVALOIL *KVALGAS

Keywords for activating partitioning tracer and species option, and defining the K-values between the water, oil and gas phase. For partitioning species, all three keywords must be given.

If these keywords are omitted, the species will be species confined in the water phase (for instance ideal/passive tracers)

Syntax:

```
*KVALWAT <Kwat>
```

```
*KVALOIL <Koil>
```

```
*KVALGAS <Kgas>
```

The K-values are absolute K-values. For instance Koil/Kwat will be the species K-value between water and oil.

4.4.11. *CO2

Keyword for activating the CO2 option in IORSim.

If used, the *KVALWAT, *KVALOIL, *KVALGAS must be omitted. Internally calculated K-values are then used.

Syntax: *CO2

4.4.12. *MODELINSTANCE

Syntax:

```
*MODELINSTANCE Name IndxLow IndxHigh IndyLow IndyHigh IndzLow  
IndzHigh
```

The Name should be the name given in *MODELTEMPLATE. The 6 indices define a box in xyz space in the ECLIPSE sense.

4.4.13. *WELLSPECIES

Syntax:

```
*WELLSPECIES  
  
*INJECTOR Wellname1  
  
*TIME time Comp1 ... CompN InjectionTemp  
  
:  
  
*INJECTOR WellnameN  
  
*TIME time Comp1 ... CompN InjectionTemp
```

Note that for geochemical cases Comp1 ... CompN must be replaced by a solution name previously defined in section 4.4.4 *SOLUTION

i.e

```
*TIME time solution_name1 InjectionTemp  
*TIME time solution_name2 InjectionTemp
```

Where:

All injection wells defined in Eclipse need to be included, irrespective of whether the IORSim simulation includes nonzero injection of species in all of them.

Wellname = Name of the well. Should be consistent with naming in Eclipse.

time = Injection time

Comp1 ... CompN = injected concentration of all IORSim species (tracer, silicate model)
or
solution_name (geochem model)
InjectionTemp = injection temperature

4.5. Description of keywords – General input part II

4.5.1. *OUTPUT

Syntax:

```
*OUTPUT  
*PRODUCER Well_name_1  
.  
.  
.  
*PRODUCER Well_name_N
```

This keyword defines for which well(s) species concentrations (reported in the CASE.trcconc file) and species production rates (reported in the CASE.trcprd file) should be written.

Well name 1 ... Well name Nprod: Required arguments for keyword *PRODUCER

The well names should correspond to (case sensitive) well names in the Eclipse case to be reported. If well names do not correspond to Eclipse well names, no values are reported in the CASE.trcprd and CASE.trcconc files.

4.5.2. *WELLPLOT INTERVAL

Syntax:

```
*WELLPLOT_INTERVAL  
T1 DTPLOT1  
T2 DTPLOT2  
.  
.
```

This keyword defines the frequency of output to CASE.trcconc and CASE.trcprd files.

T1: Dummy, this argument is ignored (T1 is always 0).

DTPLOT1: Required argument that give the time interval for reporting in the period 0-T2.

T2: Optional argument, first time when reporting interval is changed to DTPLOT2.

DTPLOT2: Required argument if T2 is set, defines the reporting interval from T2.

\: Required argument, marks the end of the *OUTPUT keyword.

Units:

T1, DTPLOT1, T2, DTPLOT2: days

4.5.3. *PRINTLEVEL

This keyword defines the amount of output on the screen, when running IORSim.

Syntax: *PRINTLEVEL n_level where n_level is an integer ≥ 0

When n_level is 0, no screen output is printed on the screen

4.5.4. *END

Syntax: *END

This keyword defines the end of the IORSim input file. It has no arguments.

5. GeoChem input file

The geochemistry is entered in a separate file geocheminp defined in keyword ... in .trcinp file. Below all keywords are listed

Keyword	Short description
add_species	opens an include file allowing additional definitions of species and reactions
complex	surface complexes
equilibrium_phases	specification of minerals that will be in equilibrium with solution 0
ixchange	define ion exchange capacity
rate	Defines minerals and dissolution kinetics
solution(omitted)	This is moved to section 4.4.1 *SOLUTION in .trcinp file

5.1.1. add_species

Use: In many cases one would like to add new reactions or change the existing logK values. This is done by specifying them in a separate file, usually entered in the top of the CASE.geocheminp file

Description: Imports new basis species, surface complexes, secondary species and ion exchange species into the database. If some of the species already exists, they will be overwritten. Example file shown below.

```
add_species "ions.txt"
```

In IORSim it is possible to define new interactions that are not initially present, this is done by including a file 'ions.txt' in the run catalogue. If the keyword HKF is given, the geochemical solver uses the HKF EOS to calculate the log K values. The units and the symbols are the same as those given in the appendix of (Johnson, Oelkers et al. 1992). If the keyword ANA is given, the geochemical solver calculates the log K directly based on the formula

$$\log_{10} K = A_1 + A_2 T + \frac{A_3}{T} + A_4 \log_{10} T + \frac{A_5}{T^2}, \quad (5.1)$$

$A_{1...4}$ is given as input, if too few is given the rest is assumed to be zero. The input is quite flexible as illustrated below. The species are read in before the simulation is started and added to the data base, ions that are not used in the simulation are simply removed from the database.

```

#BASIS_SPECIES
#Name a0 Mw      DeltaG      DeltaH      S      a1      a2      a3      a4      c1
      c2      omega
#name a0 mol_weight (if ANA is used)
#ACETATE
ACE-  5 59.04 /HKF -88270      -116160      20.6 7.7525      8.6996
      7.5825      -3.1385      26.3 -3.86 1.3182      /
/end
SECONDARY_SPECIES
# DeltaG      DeltaH      S      a1      a2      a3      a4      c1      c2      omega
# ANA = a0 + a1 * T + a2/T + a3 * logT + a4/T^2
NaX = X- + Na+ / ANA 0. /
CaX2 = 2X- + Ca+2 / ANA -0.8 /
MgX2 = 2X- + Mg+2 / ANA -0.6 /
KX = X- + K+ / ANA -0.7 /
BaX2 = 2X- + Ba+2 / ANA -0.91 /
SrX2 = 2X- + Sr+2 / ANA -0.91 /
BgCl- = Bg+2 + 2.34Cl- / HKF -17450      -18270      34.1 5.2088
      4.9399      3.8015      -2.9832      9.8168      -1.6698      -0.03 /
GCO3Bg+ = GCO3- + Bg+2 / ANA .2 /
/end
EXCHANGE_SPECIES
#ion exchange species
#name
Z+
/end
SECONDARY_SPECIES
#dissociation of acetic acid
ACEH = ACE- + H+ /HKF -94760      -116100      42.7 11.6198      5.218
      2.5088      -2.9946      42.076      -1.5417      -0.15 /
#ion exchange with sulphate and chloride - not standard
SO4Z2 = 2Z+ + SO4-2 / ANA -0.8 /
ZCl = Z+ + Cl- /ANA -0.7/
/end
MINERAL_PHASES
# HKF = mol_volume      DeltaG      DeltaH      S      a1      a2      a3 a7
# ANA = mol_volume a0 + a1 * T + a2/T + a3 * logT + a4/T^2
WITHERITE = Ba+2 + HCO3- - H+ /HKF 45.81 -278400 -297500      26.8 21.5
      11.06 -3.91      /
#K-FELDPAR = Al+3 + 2H2O - 4H+ + K+ + 3SiO2 /HKF 108.87 -895374 -
949188      51.13 76.617      4.311 -29.945
/end

```

5.1.2. complex

Line	complex Parameter number	Parameter description	Symbol	Unit	Example value
1	1	Numerical method	Text	No unit	method
1	2	Which method used	integer	No unit	1 or 2
2	1	Specific surface area	Text	No unit	s_area
2	2	Value of surface area	SA	mol/kg rock	1000
2	3 (optional)	Size of diffusive layer	D	nm	10
3	1	Name of surface complex	Text	No unit	GCa
3	2	amount of complex	C	Sites/nm ²	2

Example

```
complex
# method = 0 surface potential is set to zero
# method = 1 (default) solves Grahame equation
method 1
#specific surface area m^2/L pore volume
#if explicit diffusive layer calculation the size of diffusive layer
#in nm must be given
# s_area size of diffusion layer
# 1000 10
s_area 1000
#Name sites/nm^2
GCa 2
GCO3 2
/ end
```

5.1.3. equilibrium_phases

Use: Defines mineral or other buffer phases.

Description: Reads a list of buffer phases and their concentration (weight fraction), each buffer on a new line. The keyword is ended with a slash '\' in the first column on a new line.

Up to 4 items are read from each line:

Item	Name	Description	default
1	<i>buffer_name_</i>	Name of buffer	-
2	<i>c_buffer_</i>	Concentration of buffer, mol/kg w	0
3	<i>log_af_</i>	Log activity	0
4	<i>phase_name</i>	Valid input: 'oil' or 'gas', or else water is assumed.	water

Example:

```
--
equilibrium_phases
# name      mol/kg w log_a  phase
magnesite   10          0.
calcite     24          0.
CO2 (G)     .6         -3.5
/ ends keyword
```

5.1.4. iexchange

Use: Defines ion exchange capacity.

Description: Reads two items which must be on the same line. The keyword is ended with a slash '/' in the first column on a new line.

Two items are read from each line:

Item	Name	Description	default
1	<i>exchange_name</i>	Name representing the exchange sites. The name must be defined before it is used here. A predefined name is X, representing cation exchange.	-
2	<i>capacity</i>	Exchange capacity, Q_v , in units [Eq./L PV] = [meq/ml PV]	0

Cation exchange capacity (*CEC*) is typically given in meq/100 g solid material. Typical range for sandstone could be 0.1-1.0 meq/100 g. Converting to meq/ml PV :

$$Q_v = CEC \cdot \frac{\rho_m(1-\phi)}{100\phi}, \quad (5.2)$$

where ρ_m is the matrix density in g/ml. Example: with *CEC* = 0.5 meq/100 g, porosity 0.2 and matrix density 2.7 g/ml, Q_v = 0.054 meq/ml PV.

The ion exchange calculations in the geochemical model has been extended to also allow anion exchange at positively charges surfaces. The anion exchange capacity, *AEC*, in meq/100 g solid, is converted to Eq/L pore volume by

$$Z^+ = \frac{AEC \cdot \rho_m (1 - \varphi)}{100\varphi} \quad (5.3)$$

Where ρ_m is the matrix (solid) density in g/ml and Z^+ (Eq/L) represents the anion exchange capacity input. The exchange of sorbed Cl^- with SO_4^{2-} can be written:

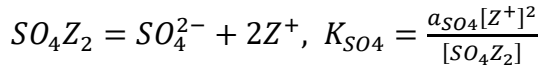


The ion exchange implemented in the geochemical model in IORSim follows the Gaines & Thomas convention and can be expressed with:

$$K_{SO_4}^{Cl} = \frac{\beta_{Cl}^2}{\beta_{SO_4}} \cdot \frac{a_{SO_4}}{a_{Cl}^2} = \frac{\beta_{Cl}^2}{\beta_{SO_4}} \cdot \frac{[SO_4^{2-}]\gamma_{SO_4}}{[Cl^-]^2\gamma_{Cl}^2}, \beta_{SO_4} = 2 \cdot \frac{[SO_4Z_2]}{Z^+}, \beta_{Cl} = \frac{[ClZ]}{Z^+} \quad (5.5)$$

Where β_i denotes the equivalent fraction occupied by species i , and a_i denotes the activity of species i .

The input exchange constants is in form of half reactions:



From experiments, we obtain the relative association of ions with the rock surface:

$$\frac{K_{SO_4}}{K_{Cl}^2} = \frac{a_{SO_4}[Z^+]^2}{[SO_4Z_2]} \cdot \frac{[ClZ]^2}{a_{Cl}^2[Z^+]^2} = \frac{a_{SO_4}}{a_{Cl}^2} \cdot \frac{2\beta_{Cl}^2}{\beta_{SO_4}} = 2K_{SO_4}^{Cl} \quad (5.7)$$

We use Cl^- as a reference ion and set the constant for this ion to unity, $K_{Cl}=1$, and use $K_{SO_4}^{Cl} = 2K_{SO_4}^{Cl}$ as input, or for other ions, $K_{A_i} = z_i K_{A_i}^{Cl}$.

Similarly for cations, one typically uses Na^+ as a reference ion ($K_{Na}=1$) and $K_{M_i} = z_i K_{M_i}^{Na}$.

Example defining cation + anion exchange capacities:

```
#include file with additional species, in this case anion exchange (Z+) is defined
add_species "ions.txt"

iexchange
X 0.045
Z+ 0.06
/ end
```

5.1.5. rate

Use: Defines minerals and dissolution/precipitation kinetics.

Description: Reads mineral name and kinetic parameters for dissolution/precipitation from a single line. Several minerals can be specified, each on a new line. The keyword is ended with a slash '/' in the first column on a new line.

For each mineral in the reservoir a name must be given, it is important that the species in the mineral is also in the solution keyword. Mineral dissolution and precipitation are described by the following rate law:

$$\frac{\partial c_i(t)}{\partial t} = \sum_j \xi_{ij} I_j, \quad (5.8)$$

where ξ_{ij} is a stoichiometric coefficient for the exchange of specie i between the aqueous phase and mineral j. The flux I_j is given by:

$$I_j = \left(k_1 e^{-\frac{E_1}{R} \left(\frac{1}{T} - \frac{1}{298.15} \right)} + k_2 e^{-\frac{E_2}{R} \left(\frac{1}{T} - \frac{1}{298.15} \right)} a_H \right) |1 - \Omega_j^m|^n. \quad (5.9)$$

The rate parameters are listed below. The specific surface area is not currently used, but must be given a value in the input file. The original idea was to include a specific surface area for each mineral and use this to calculate changes in permeability, in the same way as for the silicate model.

Rate Parameter number	Parameter description	Symbol	Unit	Example value
1	Name of mineral	Text	No unit	Calcite
2	Amount present in rock		mol/kg rock	38
3	Specific surface area	SA	m ² /liter rock	1
4	Log activity of mineral	log a	Dimensionless	0
5	Activation energy	E_1	J/mol	37.8e+3
6	Neutral rate constant	k_1	mol/s	3.43e-2
7	Activation energy	E_2	J/mol	8.4e+3
8	Acidic rate constant	k_2	mol/s	1.11E+03
9	Potens in rate constant	m	Dimensionless	1
10	Potens in rate constant	n	Dimensionless	1

Example defining kinetics for calcite, magnesite, and anhydrite. Note that when we give the values 0 for the amount, it means that magnesite and anhydrite can be formed during the simulation

```
--
rate
# rate = (k_1*exp(-Ea/Rg) (1/T-1/298.15)+k_2*exp(-Ea/Rg) (1/T-1/298.15) *(1-SI^n)^m
#mineral      mol/kg   Sg   log_af   logEa_1      k_1      logEa_2      k_2
n             m
calcite       38.454 1     0     37.8e+3      3.43E-02   8.4e+3 1.11E+03    1
1
magnesite     0.0    1     0     60e+3 .7E-08      0.0    0.0          1    1
anhydrite     0.0    1     0     60e+3 7E-08      0.0    0.0          1    1
barite        0.0    2     0     30.8  1.11E-03    0.0    1.11E-03     1    1
/end
```

5.1.6. solution

Use: Defines brine ion composition.

Description: The keyword and a number on the same line, followed by a list species names and concentration starting on the next line, defines a solution. The species list is ended with a slash '/' in the first column on a new line (which also terminates the keyword). The first solution number should be zero, and other solutions should be defined in increasing order

Example using the geochemical model:

```
#add include file 'ions.txt' with additional definitions
add_species "ions2.txt"
solution 0
-- formation water. ions are given in mol/l
    pH 7.5
    Na 0.050
    K  0.01
    Mg 0.045
    Ca 0.013
    HCO3 0.002
    Cl 0.125
    SO4 0.024
/ end solution 0
This solution input must be defined in CASE.trcinp file, *SOLUTION keyword, see section
4.4.4 in this manual
```

6. Output files

6.1. Grid concentration files

IORSim can write time dependent concentration fields (i.e. concentration values in the grid blocks) for all species defined in a case. The concentrations are written to a file compatible with Eclipse's unified restart file format (.UNRST or .FUNRST) described in detail in Chapter 6 of ECLIPSE File Formats Reference Manual. This file can be displayed using software compatible with Eclipse's unified restart file format, e.g. FloViz. The format of the file is the same as the restart file described above, but contains concentrations recorded for all time steps. By default the file is unformatted, but a formatted grid concentration file format may be chosen by the user (see Section 3.3.4.)

Naming convention:

The IORSim concentration file is named following the convention from Eclipse, where the case name forms the base of the name, and the file name extension is .FUNRST. To avoid possible a naming clash with Eclipse's restart file, the IORSim file is given a prefix IORSIM_PLOT . If the case is named CASE.trcinp, the IORSim concentration file is thus named IORSIM_PLOT_CASE.FUNRST.

6.1.1. Example of IORSIM_PLOT_*.FUNRST file

The following is an example of an IORSim grid concentration file in the .FUNRST format in a 50x1 grid. The case has two species (HTO and PART):

```
'SEQNUM' 1 'INTE'  
2  
'INTEHEAD' 95 'INTE'  
001000  
00501150  
001000  
001000  
001000  
9700300  
0019000  
000000  
000000  
000000  
000000  
0000177  
178900000  
000000  
000000  
000000  
0000-1  
'LOGIHEAD' 15 'LOGI'  
FFFFFFFFFFFFFFFF  
'DOUBHEAD' 1 'DOUB'  
3.50000000000000E+00  
'HTO' 50 'REAL'  
0.00000000E+00 0.00000000E+00 0.00000000E+00 0.00000000E+00  
0.00000000E+00 0.00000000E+00 0.00000000E+00 0.00000000E+00  
0.00000000E+00 0.00000000E+00 0.00000000E+00 0.00000000E+00
```

[illegible]

```

2.32934245E-12 -1.88954429E-12 -4.21049176E-12 -3.34269374E-12
-4.52695363E-13 2.45646407E-12 3.95531958E-12 3.67812768E-12
2.08192101E-12 -5.30878848E-14 -2.03264012E-12 -3.44255353E-12
-4.15918125E-12 -4.26308815E-12 -3.93254526E-12 -3.35992323E-12
-2.70380528E-12 -2.07190016E-12 -1.52355396E-12 -1.08125303E-12
-7.43867207E-13 -4.97837557E-13 -3.25047308E-13 -2.07540785E-13
-1.29845719E-13 -7.97373032E-14 -4.81333757E-14 -2.85983192E-14
-1.67430699E-14 -9.66861767E-15 -5.51207792E-15 -3.10480477E-15
-1.72915201E-15 -9.52758753E-16
'SEQNUM ' 1 'INTE'
4
.

```

6.1.2. Information contained in the IORSim_*.FUNRST file

Here we describe some of the information in the IORSim_*.FUNRST file. For details on the format and contents of the tables given in the IORSim_*.FUNRST file, we refer to Chapter 6 in the ECLIPSE File Formats Reference Manual.

SEQNUM: Contains one integer, the sequence number, which is set equal to the current report step number read from the Eclipse restart file used by IORSim.

INTEHEAD: Contains 95 integers. The following contain useful information for us:

Items 9,10,11 - grid dimensions NX, NY and NZ

Item 12 - NACTIV = number of active cells

Item 65 - IDAY = calendar day at this report time

Item 66 - IMON = calendar month at this report time

Item 67 - IYEAR = calendar year at this report time

Item 95 - negative number implies that the file is produced by other simulators than Eclipse.

LOGIHEAD: Contains 15 logicals, not relevant for the current version of IORSim.

DOUBHEAD: Contains 1 real. This is the time in days at this report step.

The remainder of the data in the IORSim_*.FUNRST file, until the next SEQNUM, contains one table per species. These tables ('PART ' and 'HTO ' in the example in Section 4.4.1 over) contain real numbers representing the species concentration in the active grid blocks.

7. Example cases

7.1. Tracer: CASE.trcinp file

```
*TEMPERATURE
*GRIDPLOT_WRITE
*GRIDPLOT_FILE FORMATTED
*REACTING_SYSTEM /
*TRACER_LGR
# n_lgr
    0
*INTEGRATION
# tstart tstop
    0.0 1.e99
# dtmin dtmax
    0.0 1.e99
# dtmin_ecl dtmax_ecl
    0.0 1.e99
# 0.0 0.5
# metnum
    0
solver sparskit2
*integrate_species
*MODELTYPE Tracer
*SPECIES SpecieNo1
*SPECIES SpecieNo2
#
# Only one model template included here
# -----
*MODELTEMPLATE Comp1
*tinit 130.0
# Relative vale for water, Jan Ludvig
# Relative vale for oil, Jan Ludvig
# Relative vale for gas, Jan Ludvig
# Relative vale for rock, Jan Ludvig
# cpWat cpOil cp Gas cpRock
    4.0 2.0 1.0 1.0
# Water density, test value
# Oil density, test value
# Gas density, test value
# Rock density, test value
# roWat roOil ro Gas roRock
```

```

    1000.0 800.0 400.0 2500.0
*COMP 0.0 0.0
*KVALWAT 1.0 1.0
*KVALOIL 0.0 1.0
*KVALGAS 0.0 0.0
#           Name  IndxLow IndxHigh
*MODELINSTANCE  Comp1  1          10  1  1  1  1
*WELLSPECIES
*INJECTOR INJ1
#      Time      Comp          Tempinj
*TIME  3.5      2.314e-3 2.314e-3 25.0
*TIME  4.0      0.0      0.0      25.0
*OUTPUT
*PRODUCER PRO1
*wellplot_interval 0.0 0.01 /
*PRINTLEVEL 1
*END

```


7.2. Geochem: CASE.trcinp file and geochem input files

7.2.1. CASE.trcimp file

```
*TEMPERATURE
*GRIDPLOT_WRITE
*GRIDPLOT_FILE FORMATTED
*REACTING_SYSTEM /
*TRACER_LGR
# n_lgr
    0
*INTEGRATION
# tstart tstop
    0.0 1.e99
# 0.0 2.0
# dtmin dtmax
    0.0 1.e99
# dteclmin dteclmax
    10 10
# metnum
    0
solver sparskit2
*integrate_species
*MODELTYPE GeoChem
*solution solution0
    pH 9
    Ca 0.15
    Mg 0.021885
    Cl 2.5
    HCO3 1e-8
    Na 1.1428
    SO4 1e-8
    K 0.0073
    Ba 0.00184
    Sr 0.0085
*solution solution1
    pH 1.
    Ca 0.013
    Mg 0.0445
    Cl 0.525
    HCO3 0.002
    Na 0.45
```

SO4	0.024
K	0.01
Ba	0
Sr	0

*solution solution2

pH	1.
Ca	0.013
Mg	0.0445
Cl	0.525
HCO3	0.002
Na	0.45
SO4	0.024
K	0.01
Ba	0
Sr	0

*solution solution3

pH	0.5
Ca	0.02
Mg	0.06
Cl	0.525
HCO3	0.002
Na	0.45
SO4	0.024
K	0.01
Ba	0
Sr	0

*SATNUM Mg 1 12 1.e-4 1.e-3

*MODELTEMPLATE Comp1

*TINIT 130.0

Relative value for water, Jan Ludvig

Relative value for oil, Jan Ludvig

Relative value for gas, Jan Ludvig

Relative value for rock, Jan Ludvig

#	cpWat	cpOil	cp Gas	cpRock
	4.0	2.0	1.0	1.0

Water density, test value

Oil density, test value

Gas density, test value

Rock density, test value

#	roWat	roOil	ro Gas	roRock
	1000.0	800.0	400.0	2500.0

*COMP solution0

*CHEMFILE geochem1

```

*MODELTEMPLATE Comp2
*TINIT 130.0
# Relative vale for water, Jan Ludvig
# Relative vale for oil, Jan Ludvig
# Relative vale for gas, Jan Ludvig
# Relative vale for rock, Jan Ludvig
# cpWat cpOil cp Gas cpRock
    4.0  2.0    1.0    1.0
# Water density, test value
# Oil density, test value
# Gas density, test value
# Rock density, test value
# roWat roOil ro Gas roRock
    1000.0 800.0 400.0 2500.0
*COMP solution0
*CHEMFILE geochem2
#           Name      IndxLow      IndxHigh
*MODELINSTANCE Comp1      1 1 1 1      1 25
*MODELINSTANCE Comp2      1 1 1 1     26 50
*WELLSPECIES
*INJECTOR INJ1
#   Time  Comp      Temp
#   0    solution1  25.0
#  30    solution2  25.0
*TIME  0  solution1  25.0
*TIME  30 solution2  25.0
*TIME  31 solution3  25.0
*OUTPUT
*PRODUCER PRO1
*wellplot_interval  0.0 0.01
*PRINTLEVEL 1
*END

```

7.2.2. geochem1 file (reservoir region 1)

```

rate
# rate = (k_1*exp(-Ea/Rg) (1/T-1/298.15)+k_2*exp(-Ea/Rg) (1/T-1/298.15) * (1-
SI^n)^m
#mineral    wt-fraction  Sg    log_af    logEa_1    k_1          logEa_2    k_2
n          m
calcite      38.454      1     0     37.8  3.43E-02    8.4  1.11E+03    1
1
magnesite    0.0      1     0     60    .7E-08      0.0  0.0        1
1
anhydrite    0.0      1     0     60    7E-08      0.0  0.0        1
1
barite       0.0      2     0     30.8  1.11E-03    0.0  1.11E-03    1
1
#talc        0.01      2     0     42    8.84E-09    0    0.00E+00    1
1
#strontianite 0.01      1     0     30.8  1.11E-10    30.8  0.00E+00    1
1
#muscovite   0.01      1     0     22    8.84E-15    30.8  0.00E+00    1    1
#kaolinite   0.01      1     0     22.2  1.47E-09    0     9.26E-08    1    1
#witherrite  0.01      1     0     30.8  1.11E-04    48    1.11E-03    1    1
#paragonite  0.01      1     0     0     8.84E-17    23.6  8.84E+03    1    1
#quartz      0.01      1     0     0     2.80E-07    0     0.00E+00
1      1
/ end
iexchange
X 0.01
/ end
#method 0 set surface charge and potential to 0 in calculations
#method 1 include surface charge and potential in calculations (FAST IF IT
CONVERGES)
#method 2 same as 1, but more stable (SLOW)
#complex
#method 0
#GCa 1e-2
#GCO3 1e-2
#/ end
#row 1 to 6 /* Set injector wells (NB! counting starts on 1 and excludes
the producers) */
#row 1
/end

```

7.2.3. geochem2 file (reservoir region 2)

```

rate
# rate = (k_1*exp(-Ea/Rg) (1/T-1/298.15)+k_2*exp(-Ea/Rg) (1/T-1/298.15) * (1-
SI^n)^m
#mineral    wt-fraction  Sg    log_af    logEa_1    k_1          logEa_2    k_2
n          m
calcite      40.454      1     0     39.0  3.43E-02    8.4  1.11E+03    1
    1
magnesite    0.0      1     0     65    .7E-08      0.0  0.0        1
    1
anhydrite    0.0      1     0     60    7E-08      0.0  0.0        1
barite       0.0      2     0     30.8  1.11E-03    0.0  1.11E-03    1
    1
#talc        0.01      2     0     42    8.84E-09    0    0.00E+00    1
    1
#strontianite 0.01      1     0     30.8  1.11E-10    30.8  0.00E+00    1
    1
#muscovite   0.01      1     0     22    8.84E-15    30.8  0.00E+00    1    1
#kaolinite   0.01      1     0     22.2  1.47E-09    0     9.26E-08    1    1
#witherrite  0.01      1     0     30.8  1.11E-04    48    1.11E-03    1    1
#paragonite  0.01      1     0     0     8.84E-17    23.6  8.84E+03    1    1
#quartz      0.01      1     0     0     2.80E-07    0     0.00E+00
    1    1
/ end
iexchange
X 0.01
/ end
#method 0 set surface charge and potential to 0 in calculations
#method 1 include surface charge and potential in calculations (FAST IF IT
CONVERGES)
#method 2 same as 1, but more stable (SLOW)
#complex
#method 0
#GCa 1e-2
#GCO3 1e-2
#/ end
#row 1 to 6 /* Set injector wells (NB! counting starts on 1 and excludes
the producers) */
#row 1
/end

```

7.3. Silicate case

```
*TEMPERATURE
*GRIDPLOT_WRITE
*GRIDPLOT_FILE FORMATTED
*REACTING_SYSTEM /
*TRACER_LGR
# n_lgr
    0
*INTEGRATION
# tstart tstop
    40 1.e99
# dtmin dtmax
    0.0 1.e99
# dtecl dteclmax
    10 20
# metnum
    0
solver sparskit2
*integrate_species
*MODELTYPE SILICATE
*SPECIES Silica
*SPECIES MobGel
*SPECIES HCl
*SPECIES Ca
*SPECIES ImMobGel
#      compName  ilo ihi  clo  chi
*SATNUM  SpecNo1    0  12  0.5  0.0

#
# Only one Model template here
# -----
*MODELTEMPLATE Compl
*TINIT 95.0
# Relative vale for water, Jan Ludvig
# Relative vale for oil, Jan Ludvig
# Relative vale for gas, Jan Ludvig
# Relative vale for rock, Jan Ludvig
# cpWat cpOil  cp Gas  cpRock
    3.9  2.1    2.1    1.0
# Water density, test value
# Oil density, test value
# Gas density, test value
```

```

# Rock density, test value
# roWat roOil ro Gas roRock
  999.9 800.0 400.0 2350.0
#*COMP 0. 0. 0. 0.4
#
# Name IndxLow IndxHigh IndyLow IndyHigh IndzLow IndzHigh
*MODELINSTANCE Comp1 1 20 1 20 1 10
*WELLSPECIES
*INJECTOR I1
# Time Comp
*TIME 0.0 0 0 0 0.04089 0 20
*TIME 300 5 0 5.0 0. 0 20
*TIME 330 0 0 0 0.04089 0 20
*OUTPUT
# Output wells
*PRODUCER P1
*PRODUCER P2
*wellplot_interval 0.0 0.01
*PRINTLEVEL 1
*END

```

8. References

Johnson, J. W., E. H. Oelkers and H. C. Helgeson (1992). "SUPCRT92: A software package for calculating the standard molal thermodynamic properties of minerals, gases, aqueous species, and reactions from 1 to 5000 bar and 0 to 1000 C." Comp. and Geo. Sci. **18**(7): 899-947.