ECLIPSE

Version 2015.2

Improving simulation data

Schlumberger

Table of Contents

1

Introduction

Reservoir simulators such as ECLIPSE require significant amounts of complex data and produce equally large amounts of output data. This means that the preparation of this data from information on the reservoir to be modelled is often best done through a graphical user interface such as Petrel, and inspection of the solution output is also best done through such an interface.

Tools like Petrel will guide you on the various kinds of data required and may check that data for consistency and completeness. The alternative, constructing a data file (sometimes referred to as a dataset) using only the instructions in the Technical Description and Reference Manual with a text editor, requires, in addition to the time required to input the volume of data and check its consistency, a systematic knowledge of the simulator requirements, and can pose difficulties for less experienced users. While Petrel simplifies the task of data input, the complexity of the data still requires guidance in using Petrel, and in this document various Petrel workflows are described to achieve the correct data construction of particular features of ECLIPSE simulations.

However, ECLIPSE itself has been designed to run independently of any user interface tool, which is why construction of a deck via text editor remains an available workflow for users. In certain cases some recourse to this method of data input is still required. These cases include very recent features in the simulators that have not yet been migrated into Petrel. They also include complex combinations of ECLIPSE features, again often recently formulated, where the individual components of the workflows are well represented in Petrel but the recommended new combinations have not yet been packaged. In these cases guidance is needed on the appropriate approach to data creation through Petrel.

The overall goal of this is to ensure that the data is consistent. Lack of consistency between the various items of data input is one of the more common reasons for the simulator to fail to converge to a correct solution. The interaction between controlling the advance of time and the various physical models used can also lead to such problems, or reduce the simulator’s efficiency. Tips and tricks for achieving this, particularly for specialized workflows, will also be offered.

Informational output from ECLIPSE commenting on the progress of the simulation (as opposed to results output describing the evolving solution) is again differently packaged in the simulator from its packaging in the graphical user interface. The Petrel approach to viewing informational output about the simulation is to present the individual items of information through various tables and dialogs. The simulator approach is to output all the information to one file, the Print or PRT file. (Other files contain the results output and aspects of the PRT file are mirrored in the run log and the packaged messages file.) The PRT file provides a wealth of information which is designed for you to read directly using a text editor. Certain features of this output are passed to Petrel directly, such as error and warning messages; other parts describe data input and output which are more easily accessed from the Petrel visual displays. Sometimes however it is helpful to

access all this information from the PRT file directly, to see what is going on in the details of the simulation, particularly when there are problems.

This document is designed to help you with these issues. The overall structure of a simulation run and the available output are also described.

Topics covered in this document

This document draws on the experience of many users to highlight the most important and most commonly encountered issues. The document has two main parts, one describing the behavior of the simulator and its output, taking a simulator viewpoint, and the final chapter describing workflows for ECLIPSE data preparation, taking a Petrel viewpoint. The information on simulator behavior covers two related and important issues:

- improving simulation performance – that is, running the simulation in as short a time as possible
- achieving convergence – that is, ensuring that the solution algorithm arrives at the right solution

There is not a hard and fast distinction between these two categories, as in order to achieve convergence the simulator may have to advance through time by using smaller timesteps, which may well increase the simulation time. However, the two are conceptually different.

The document contains the following information:

- Chapter 2 describes the structural sequence of an ECLIPSE simulation, including the ways in which convergence is achieved
- Chapter 3 describes the ways in which convergence can fail
- Chapter 4 describes the reporting protocols for ECLIPSE informational messages and error messages.
- Chapter 5 provides information on to ways in which the simulation can be improved to achieve shorter run times and/or greater robustness. All of this information is generated by ECLIPSE and reported through its own reporting mechanisms. Petrel and the ECLIPSE simulation launcher can harvest these messages and display them to you, and you can gain further understanding of the messages by seeing them in context in the simulation output PRT file.
- Chapter 6 contains various Petrel workflows for the preparation of ECLIPSE input data for several aspects of ECLIPSE functionality. In principle it is possible to create the input decks for these using a text editor, without the need for a user interface. However, the large volume of data involved makes tracking and checking the input a difficult task. Additionally the complexity of the data, and the need to dovetail the various data components together, adds to this difficulty. Petrel and other user interfaces simplify these tasks considerably, but it is still needful to know how to navigate the workflows to achieve this. Various workflows are described in detail to help you with this. Full details are given in the introduction to that part.

2

The structure of an ECLIPSE simulation

This chapter:

- Describes the simulation structure
- Defines convergence
- Defines non-linear convergence and works through a non-linear convergence example
- Discusses convergence criteria

Simulation structure

An ECLIPSE simulation has two stages: an initialization stage, where the data specifying the reservoir and its fluids at an initial date is read in and processed, and an actual simulation stage where time is advanced from the initial time in a set of timesteps, and the reservoir condition at each new time is calculated based on the calculated solution at the previous timestep. In the dataset, the input for controlling the simulation stage (such as timestep and convergence controls, well specifications and controls) is given in the `SCHEDULE` section; the previous sections (`RUNSPEC`, `GRID`, `EDIT`, `PROPS`, `REGIONS`, `SOLUTION` and `SUMMARY`) provide the data for initialization of the solution and general requests for simulation output.

In the simulation stage, the simulator's choice of timestep size is based partly on the difficulty the simulator experiences in trying to calculate the conditions at the next time, and the size of a timestep may vary from hours to months. So that you can monitor the reservoir in a consistent way, timesteps are grouped together into report steps whose size you can determine. Regardless of the size and number of timesteps between two report times, a report is made at the exact time specified, and the previous timestep will be adjusted if necessary to ensure this.

Within a timestep, the calculation of the new conditions at the end of the timestep proceeds iteratively, in a series of non-linear iterations (alternatively referred to as Newtons, as the Newton-Raphson method is used.) Each non-linear iteration is made by solving a linear algebra problem, and this again is done iteratively in a series of linear iterations. These linear iterations are done using a variety of specialist linear solvers.

All this is shown diagrammatically the following figure. More precise definitions of the concepts in this section will be given later in this chapter.

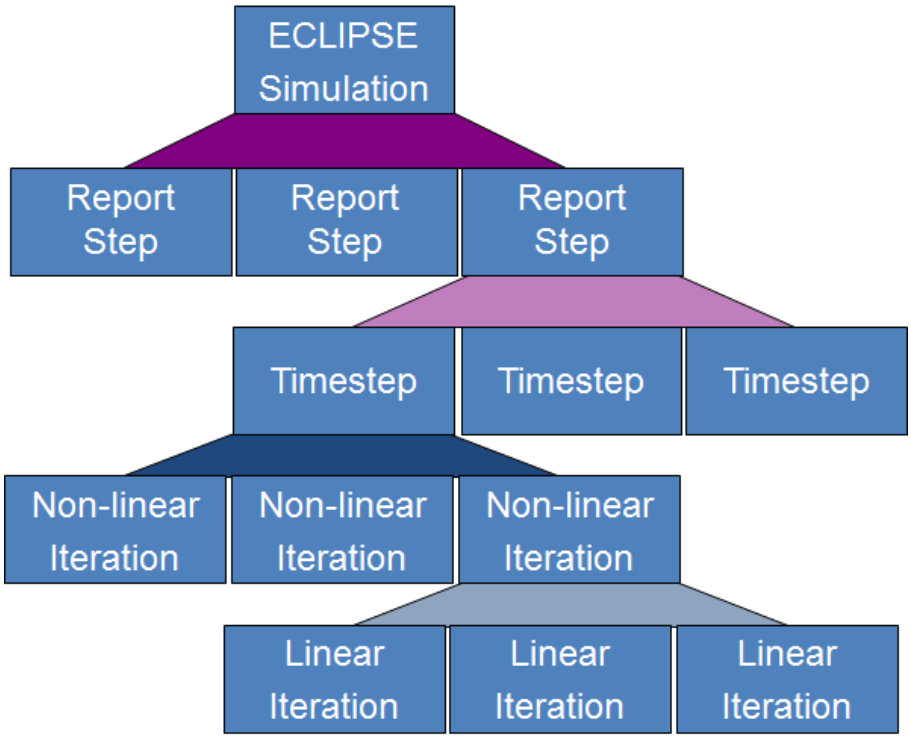


Figure 2.1. The structure of an ECLIPSE simulation

Convergence

Each iteration in a timestep reduces the error in the equations in a systematic way. The process of reducing the error is called convergence. ECLIPSE has default values that control how many timesteps will be used to reach the next report that you have asked for. These default values will work well in most cases, but there are times when you may need to adjust them to take shorter or longer timesteps as appropriate, depending on your analysis of the output data and the convergence issues observed. In the dataset these values are set in the TUNING keywords. Different default values also control how many non-linear iterations will be used to solve each timestep.

These default values should normally be left unchanged. In a few cases, adjustments to the convergence criteria can improve the performance of the simulator. In most cases however the greatest improvements in performance are obtained by identifying the cause of the non-linear problem and changing the data model to reduce the non-linearity.

The first part of this document will explain how to avoid problems of this type, and how to find and fix the problems if they do occur. In general, there is a trade-off between the number of timesteps used and the number of non-linear iterations needed. If the timestep is shorter, the non-linear calculation required to solve the timestep is usually easier and fewer non-linear iterations are required.

If the non-linear iterations do not converge properly, it is possible that the linear iterations themselves have not converged within a previous non-linear (Newton) step. Once this has happened, the solution itself may be corrupted. By the time such problems occur in the linear iterations, it is usually too late to fix them by adjusting the linear convergence control. In this situation the best advice is to avoid such problems by controlling the timestepping and the non-linear iterations. An alternative approach is to revert to an earlier timestep and try a different linear solver, such as CPR.

Non-linear convergence

The equations that the simulators are trying to solve are non-linear – where, for instance, doubling the tubing-head pressure of a water injector will not usually double the amount of water injected, and doubling the oil saturation in a grid block will not usually double the oil mobility in that grid block.

The simulators use an iterative process based on Newton's method to solve these non-linear equations:

1. Linearize the equations
2. Solve the linear equations
3. Check if this linear solution gives a good enough non-linear solution.

   If the solution is good enough then move to the next timestep.

   If the solution is not good enough, go back to step 1 and repeat the process.

This process is illustrated in the following figure.

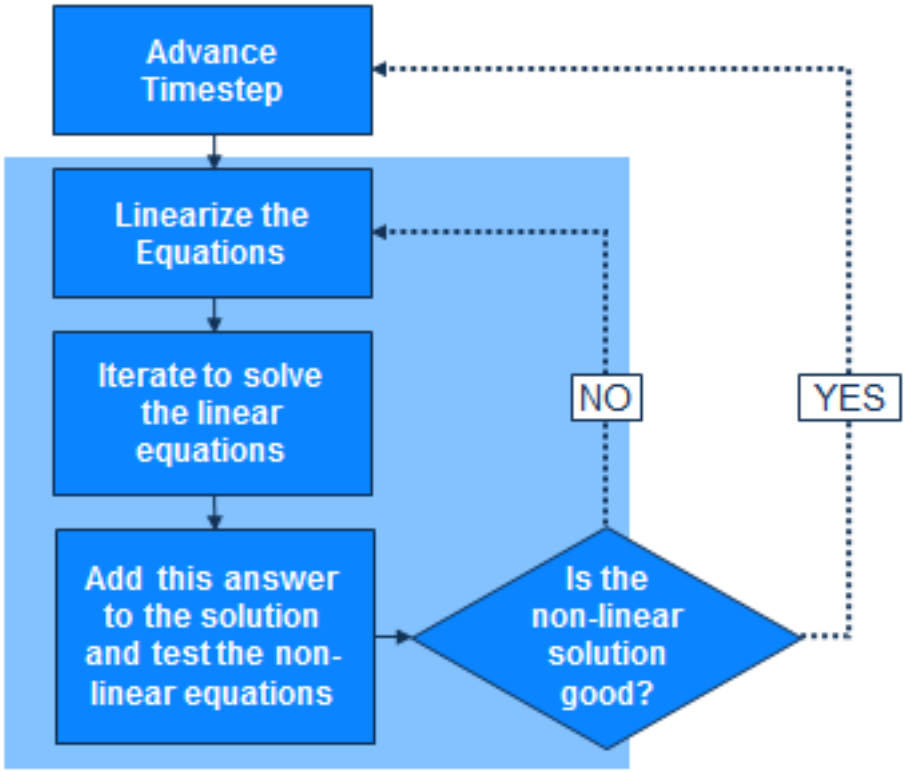


Figure 2.2. Converging the non-linear iterations

The number of passes through this loop — the *number of non-linear iterations* — is a measure of how easy it is for the model to converge. As a rough guide to the efficiency of convergence of a model (which may vary from timestep to timestep), the following ranges of number of non-linear iterations show the degree of difficulty the simulator is having in converging:

1	non-linear iterations per timestep means the step was very easy to converge.
2 to 3	non-linear iterations per timestep, meaning the step was easy to converge.

4 to 9	non-linear iterations per timestep, showing an increasingly difficult problem.
> 10	non-linear iterations per timestep can mean a problem with the model.

This is only an approximate guide, and the addition of more complex physics to the model may cause the simulator to iterate more without indicating increased difficulty of the solution or the presence of problems.

Non-linear convergence example

This example of non-linear convergence is for a continuous function of one variable only. Compared with a reservoir simulation – which even for a small 10 x 10 x 10 grid with oil and water only would need 2000 variables – this is a great simplification, but the same principles apply. For a single variable *x* there is the function *F(x)* and the objective is to find the value of *x* for which $F(x) = 0$. For even greater simplicity the function is assumed to be approximately quadratic.

Starting with an initial value x_0, the first check is to see if $F(x_0) = 0$. Usually this is not the case as shown in the following figure.

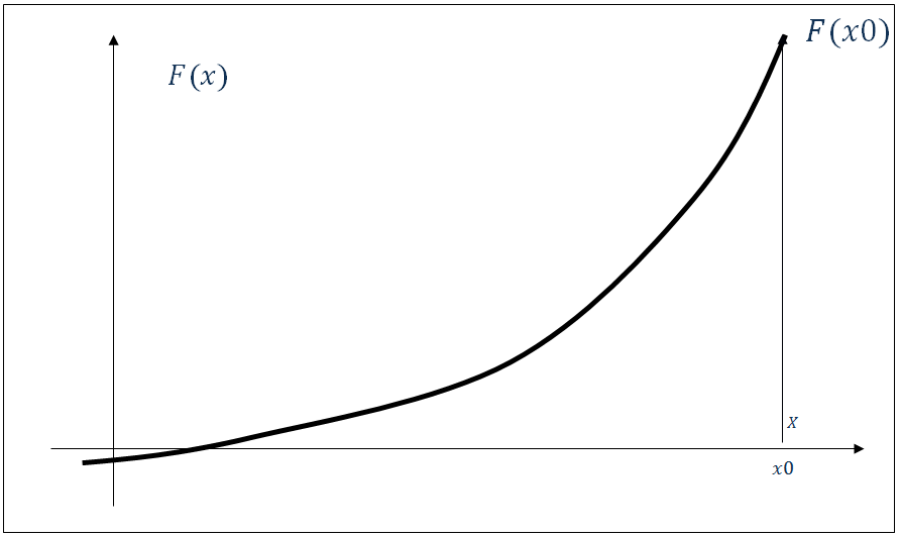


Figure 2.3. The initial guess for the solution does not satisfy the equation

In general $F(x)= 0$ is not easily solved. One reason for this is that the graph of *F(x)* is not a straight line – not linear – and so the first step is to construct the straight line tangent to the point $(x_0, F(x_0))$. This tangent line cuts the X-axis at x_1 (the next figure) and this may be an improved solution compared with x_0.

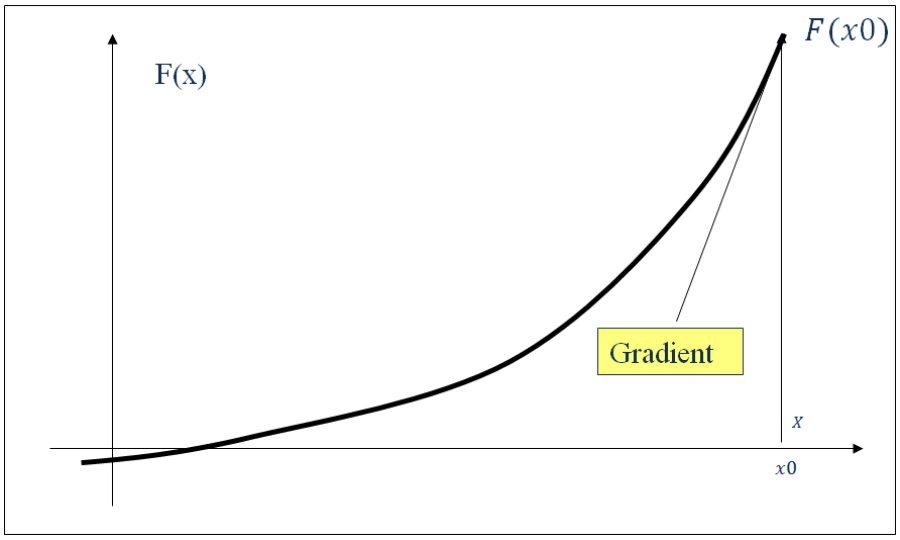


Figure 2.4. The tangent line at x_0 and its solution x_1

The solution is improved if the magnitude of $F(x_1)$ is less than that of $F(x_0)$. This procedure can now be repeated starting with x_1. Figure 2.5 shows the next 2 steps.

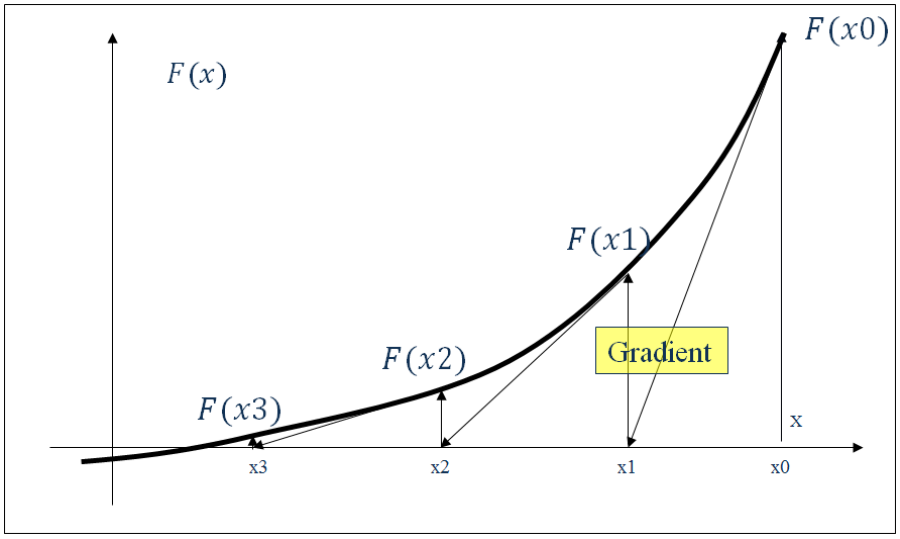


Figure 2.5. Subsequent non-linear iterations

If $F(x_3)$ is sufficiently close to zero then the solution has *converged.*

Convergence criteria

It is important to know when the solution is sufficiently converged – when it "good enough". The size of *F(x)* is called the *residual*; if this is sufficiently small, then the solution has converged. Alternatively the change in solution between two iterations may be used. ECLIPSE 100 uses the residual method, while ECLIPSE 300 uses the solution change method. One reason for this difference is that the calculation of the residual in ECLIPSE 300 involves flash calculations, which can be expensive.

For ECLIPSE 100, the residual for each phase for each grid cell is calculated and the largest of these values is compared with its default value. The total material balance error – the difference between inflow, fluid in place and outflow for each phase – is also considered. The TUNING keyword can reset the target values for these criteria. See below for the additional effect of the TUNINGDP keyword. Note that such changes are not recommended unless the user is familiar with their effects.

For ECLIPSE 300, the solution variables are the pressure and the molar density for each component. The maximum change for each of these variables over all cells is found. The maximum pressure change is compared directly with its target. The component molar density changes are converted to effective saturation changes. This criterion (once conversion to saturations has been performed) is also used by ECLIPSE 100 if the TUNINGDP keyword is used; see the discussion on this below.

Additionally, the ECLIPSE 300 non-linear convergence criteria have 'moving goalposts.' The convergence tolerances are relaxed slightly at each non-linear iteration. Effectively, convergence becomes easier as the number of iterations increases.

The idea is that if you have reached the maximum number of iterations (call that Nmax) and you are close (within a factor of two) to the convergence criteria, then you don't want to chop the timestep and waste all the work you have done so far. So the criterion relaxes by 1/Nmax every Newton iteration.

An example is shown below. The maximum number of non-linear iterations is 12, the units are metric, and the first Newton iteration uses the default convergence criteria (0.1 atm pressure and 0.01 for component specific volume). For later Newton iterations, these criteria were relaxed by 8.33% (=1/12) with each Newton iteration. After 12 Newton iterations, the criteria are doubled to 0.2 atm pressure and 0.02 for component specific volume.

```
Iteration 0 linears req 4 NTOTNL 4076
DX Pressure 0 2.803147 38 16 10 Ind:GLOB  F 1.469590
DX Comp     1 0.000262 34 31  6 Ind:GLOB  T 0.010000
DX Comp     2 0.000042 34 31  6 Ind:GLOB  T 0.010000
..
NLStep= 0 lin= 4 aot= 1.91 Rmax=0.1149E-02 Rsum=0.8487E-05 egain=0.9926E+00
DCHOP2: 1 cells chopped, Try= 1
Iteration 1 linears req 7 NTOTNL 4077
DX Pressure 0 -2.799664 38 16 10 Ind:GLOB  F 1.592056
DX Comp     1 -0.003126 38 16 10 Ind:GLOB  T 0.010833
DX Comp     2 -0.000239 38 16 10 Ind:GLOB  T 0.010833
..
NLStep= 1 lin= 7 aot= 1.91 Rmax=0.3573E-01 Rsum=0.1269E-05 egain=0.3326E+01
DCHOP2: 1 cells chopped, Try= 1
Iteration 2 linears req 4 NTOTNL 4078
DX Pressure 0  2.694699 38 16 10 Ind:GLOB  F 1.714522
DX Comp     1 -0.048358 38 16 10 Ind:GLOB  F 0.011667
DX Comp     2 -0.002969 38 16 10 Ind:GLOB  T 0.011667
```

See chapter 3 on tracking convergence for a more detailed description of the output.

Fig. 2.6 shows the difference between the ECLIPSE 100 and ECLIPSE 300 methods of assessing convergence.

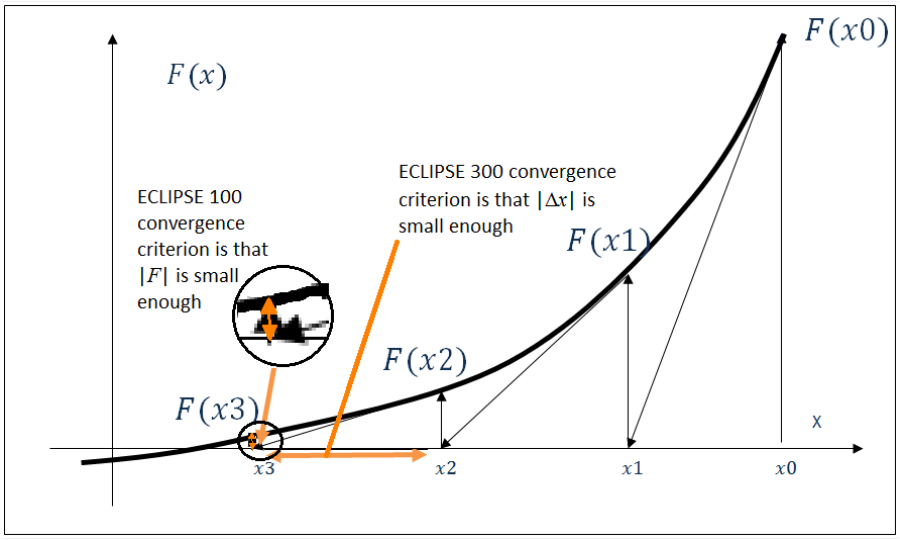


Figure 2.6. The different convergence assessments of the two simulators

If you use the`TUNINGDP` keyword (ECLIPSE 100 only), ECLIPSE 100 solves the linear equations to a tighter tolerance and convergence is reached if either the residual or the solution change criteria is small enough — in other words, the first of the ECLIPSE 100 or ECLIPSE 300 criteria to be met. If you do not use `TUNINGDP`, only the standard ECLIPSE 100 residual criterion is used to test for convergence.

3

Deal with failure to converge

Normally the simulation converges – reaches the required solution. However, unexpected problems with consistency of data, or the conflicting demands of timestepping requirements and changes in the physics, can cause a timestep to finish before the solution has converged. This chapter describes some reasons why this may happen, and how to track down and cure the problem. It looks at the problems inherent in the non-linear nature of the solution, the consequences of not converging the solution, some reasons why this might happen, and techniques for analyzing and tracking the non-convergent behavior. It also describes the different way in which ECLIPSE 300 applies its convergence criteria compared with ECLIPSE 100.

Non-linearity issues

The previous chapter describes how non-linear convergence occurs. The solutions pictured there are non-linear – the graphs are not straight lines – but for the cases illustrated there this did not cause problems. However there are other possible solutions which are more problematic. The solution in the next graph has a so-called S shape:

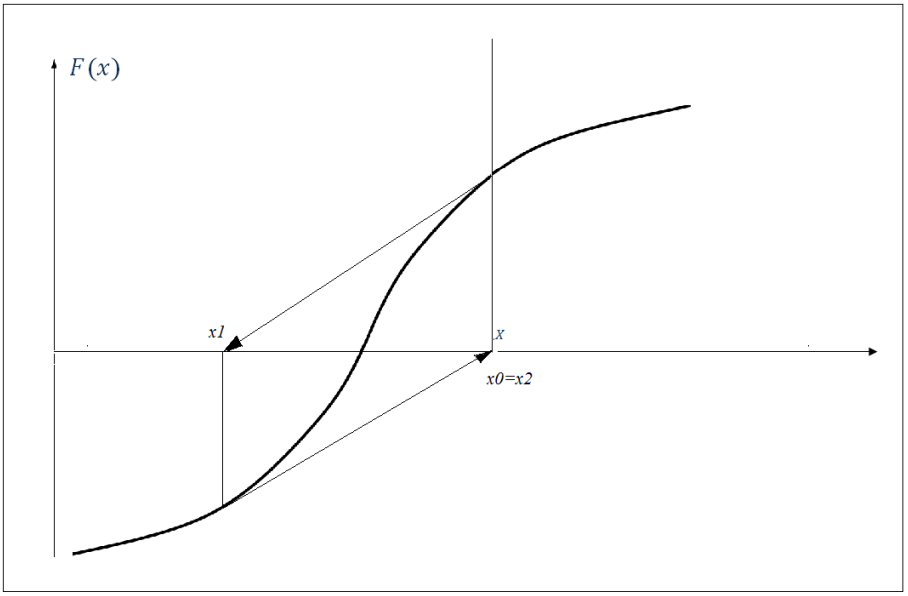


Figure 3.1. Interation without convergence for an S-shaped function F

Here the gradient of the top right segment $(x_0,F(x_0))$ finds a solution at the bottom left $(x_1,F(x_1))$. Because of the shape of the graph, the next Newton step then finds a solution back at the top right segment: $x_2=x_0$. This iteration will continue indefinitely without finding the solution.

For simplicity, the solutions illustrated have been for a function of a single variable. More complicated issues arise in ECLIPSE because the solution is multi-phase. Firstly, as the front between two phases arrives at a cell , the solution changes from single-phase to two-phase. If the solution moves too quickly through a grid block the front may not actually be tracked through that block. To prevent this there are mechanisms to limit the timestep to ensure this does not happen. This is called saturation chopping. It should be distinguished from timestep chopping which is what happens when the solution does not converge within the maximum number of non-linear iterations. In that case the timestep is chopped – its size is reduced tenfold – and the step is repeated to see if convergence can then be obtained.

In addition to phase change behavior, there are effects due to the dissolution of gas in oil, or the vaporisation of oil into the gas. In these cases the solution variable in ECLIPSE 100 itself changes – between gas saturation in the two-phase case (oil and gas – three-phase with water) and dissolved gas or vaporized oil in the single phase case (oil or gas – two-phase with water). The need to change variables imposes its own kind of non-linearity on the solution and can lead to reduced timesteps or possible non-convergence.

Other problems include flow reversals.

The method of solution can cause additional problems. A fully implicit solution solves all the variables simultaneously in the linear solver, but the IMPES (implicit pressure, explicit saturations) and AIM (adaptive implicit) methods may rely on some solution values from the previous timestep. In this case, for the explicit variables or cells, the algorithm can try to extract more fluid from a cell that is present in it. You have to reduce the timestep in these cases to prevent this behavior.

Wells also influence the non-linear behavior, The group control algorithm will sometimes change well rates at every non-linear iteration to reflect the latest calculated conditions in the reservoir. Changing well rates at every non-linear iteration can lead to poor convergence. Also, by default, the simulators only recalculate group control parameters for the first few non-linear iterations, then keep the group controls unchanged for the remaining non-linear iterations. This number of iterations can be changed using `NUPCOL`, but it is best to leave the value unchanged, unless the simulator warns you otherwise. Finally, non-monotonic VFP tables can cause convergence problems during the simulation. VFP tables are always checked for monotonicity in ECLIPSE 300, and the check can be switched on in ECLIPSE 100 by using the `EXTRAPMS` keyword.

Consequences of not converging

When the solution has converged, the expectation is for all variables to have sensible values so that the equations will be well posed. One of the meanings of well-posed is that the equations are suitable for use in the linear solver. If convergence is not achieved, for example because the linear solver reached its maximum number of iterations without converging, it is possible that when the equations are re-calculated with the non-converged solution, they may not be well-posed any more. If the non-convergence is sufficiently severe, then further iterations will not be moving towards the correct solution any more. Additionally they may no longer be linearly independent locally. This will show itself by an error message saying that a determinant is zero. Usually the simulator will then chop the timestep, but if the minimum timestep has been reached then the simulation may stop in an error state. Other signs of non-convergence are the appearance of NaNs (Not a Number – an extended kind of infinity) in the reported solution and in error messages.

Track the source of the problem

Possible causes

A single cell can cause non-convergence. As the number of cells in a simulation increases, the odds increase that a cell will cause non-convergence. If there are only one or two cells in the reservoir model that are causing problems, you can identify these cells and check if there is any engineering or data reason which could explain why they are causing problems. For example the cells may be at, or near, well completions, in which case the well control could be modified, or it could be cells with very small pore volumes in which case the MINPV keyword could be used.

Methods of analysis

Sources of convergence problems can be viewed either by visual inspection of the cell convergence states, or by inspecting the progress of the calculation through the PRT file. Visual inspection is dealt with in the section on the CONV option of the RPTRST keyword below. Convergence reporting in the PRT file is dealt with next. The ECLIPSE 100 and 300 simulators have very different formats and controls and so are described separately.

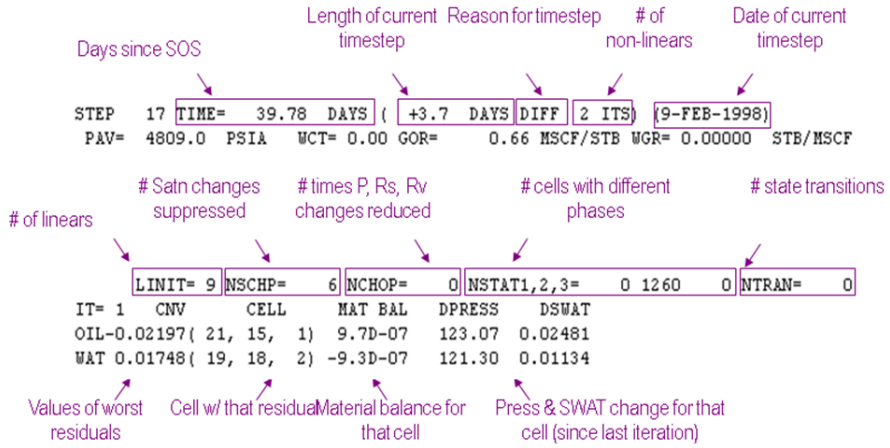
Track the source of the problem in ECLIPSE 100

This section provides inforamtion on:

- what the output looks like
- additional output with the TUNINGDP keyword
- points to look for

What the output looks like

In ECLIPSE 100, the RPTSCHED keyword can be used to select non-linear debugging information by setting the mnemonic NEWTON =2. With this activated, the report on the start of a timestep in the PRT file (and in the console log) looks like this:



The top part of this figure gives the timestepping information, and the lower half reports the convergence behavior of the first non-linear iteration (Newton).

Looking in detail at an example of the residuals:

```
IT= 0   CNV       CELL      MAT BAL   DPRESS    DSWAT    DSGAS
OIL 1.00424( 28, 45,  3)  5.3D-03     0.00  0.00000  0.00000
WAT-0.00288(  9,  3,  3) -1.3D-07     0.00  0.00000  0.00000
```

```
GAS********(  5, 45,  1) -1.3D-02     0.00  0.00000  0.00000

    LINIT= 5 NSCHP=    6 NCHOP=    0 NSTAT1,2,3=    50  5400    0 NTRAN=  321
 IT= 1   CNV      CELL      MAT BAL   DPRESS    DSWAT    DSGAS
 OIL-1.99144(  5, 45,  1)  4.3D-02   -24.89  0.00026 -0.20000
 WAT-0.16316(  2,  4,  4) -3.7D-06   -14.02  0.00490  0.00000
 GAS********(  5, 45,  1) -3.1D-02   -24.89  0.00026 -0.20000

    LINIT= 3 NSCHP=  195 NCHOP=    0 NSTAT1,2,3=    50  5370   30 NTRAN=   30
 IT= 2   CNV      CELL      MAT BAL   DPRESS    DSWAT    DSGAS
 OIL-0.62319(  5, 45,  1)  1.7D-02   -21.15  0.00081 -0.01843
 WAT-0.04162( 28,  5,  3) -1.3D-04   -30.47  0.00139  0.04000
 GAS********(  5, 45,  1) -2.2D-02   -21.15  0.00081 -0.01843

    LINIT= 3 NSCHP=   44 NCHOP=    3 NSTAT1,2,3=    50  5367   33 NTRAN=   25
 IT= 3   CNV      CELL      MAT BAL   DPRESS    DSWAT    DSGAS
 OIL-0.30993(  5, 45,  1) -8.9D-05   -26.44  0.00088 -0.21134
 WAT-0.04591( 28,  4,  3) -4.0D-05   -19.80  0.00666  0.11687
 GAS********(  5, 45,  1) -1.4D-02   -26.44  0.00088 -0.21134
```

This shows the first four non-linear iterations (IT=0, IT=1, IT=2, IT=3) in a case that has convergence problems for the gas phase. The first line shows `IT=0`, the first iteration, and column headers for the next three lines; the columns are:

`CNV` The worst residual for the OIL, WATer and GAS phases.

`CELL` The cell that has the worst residual.

`MAT BAL` The material balance for that cell, a measure of mass accuracy.

`DPRESS` The change in pressure in that cell since the last iteration.

`DSWAT` The change in water saturation since the last iteration.

`DSGAS` The change in gas saturation since the last iteration.

The residual for gas in all four iterations is shown as `*******` which means that it is greater than the maximum printable value. It has a very high residual at each iteration for cell (5,45,1), so that is the cell that is causing problems.

After each iteration report above, the line starting `LINIT=` provides more information on what is happening within the model:

`LINIT` Number of iterations required to solve the linearized equations.

`NSCHP` Number of saturation changes that were altered to suppress possible oscillations.

`NCHOP` Number of times the changes in P, Rs, or Rv were reduced to increase stability.

`NSTAT` 1, 2, 3 The number of cells in solution state, either 1, 2, or 3:

- Solution state 1 means no oil is present in the cell.
- Solution state 2 means both oil and gas are present in the cell.
- Solution state 3 means no gas is present in the cell.

`NTRAN` The number of state transitions since the last non-linear iteration.

Any non-zero value of `NSCHP` or `NCHOP` increases material balance errors for the subsequent non-linear iteration and therefore reduces the chances of convergence.

Additional output with the TUNINGDP keyword

The NEWTON switch in RPTSCHED produces extra information in the case of TUNINGDP:

- PTRG is the target pressure change; the default is 1 psi in Field units.
- STRG is the target saturation change; the default is 0 if TUNINGDP is not used and 0.01 if it is used.
- MDDP is the maximum pressure change for convergence.
- MDDS is the maximum saturation change for convergence.

Points to look for

Look for oscillations in the CNV for a phase. If the first iteration has a positive value, the next has a negative value, the next is positive, then negative, and so on, there is perhaps a non-linearity in the system.

These oscillations are sometimes associated with, for example, sudden changes in the slope of the relative permeability curves. If you have access to the SCAL program, you can plot these slopes and look for discontinuities. If you have access to a spreadsheet program then you can numerically calculate and plot the slopes. Remember that ECLIPSE will use all the values of saturation and relative permeability that you give in the table without any smoothing. The same issues arise when speeding up a converged but slow simulation and an example is given in the chapter on “Improving Simulation Performance” (PROPS section.)

Other data may benefit from similar treatment. All of the tabulated data is used to calculate the gradients for the Newton method as described in the last chapter, and well-posed gradients make convergence much easier.

Output of the reason for non-linear failure

If you set DEBUG item 1 in ECLIPSE 100 to be greater than 1, ECLIPSE outputs a number that shows the reason why a non-linear iteration has failed to converge. Some possible values are:

1. Exceeds maximum tolerable pressure change DDPLIM.
2. Exceeds maximum tolerable saturation change DDSLIM.

   Note: DDPLIM and DDSLIM are described in record 3 of the TUNING keyword.

3. Exceeds maximum tolerable material balance RSUM.

   A value of 3 means the sum of normalized residual is greater than the maximum allowed. The allowed maximum is a linear combination of items 3 and 7 of record 2 of the tuning parameter.

   The maximum residual also depends on which Newton iteration you are on.

4. Exceeds tolerable for maximum norm RNMAX.

   A value of 4 means the normalized residual is greater than the maximum allowed. The allowed maximum is a linear combination of items 2 and 6 of record 2 of the tuning parameter.

   The maximum residual also depends on which newton iteration you are on.

5. Uses but fails the TRGDDP / TRGDDS test.

A value of 5 means the change in the solution is more than TRGDDP and TRGDDS. These are the values of DDPLIM and DDSLIM until that point in the simulation. This is used in cases where there is high throughput.

Track the source of the problem in ECLIPSE 300

The way to get non-linear debugging information is by setting item 8 in the DEBUG3 keyword greater than 0. Some of the debug information has been described previously for ECLIPSE 100, but is presented again here in ECLIPSE 300 format for completeness.

Typical output is of the form:

```
Iteration            0 linears req            7
DX Pressure  0     -40.075969 25 32  4   F    1.469590
DX     Comp  1      -0.000375 25 32  1   T    0.010000
DX     Comp  2      -0.000980 25 32  1   T    0.010000
DX     Comp  3      -0.025419 25 32  1   F    0.010000
DX     Comp  4      -0.002318 25 32  1   T    0.010000
DX     Comp  5      -0.000318 25 32  3   T    0.010000
DX     Comp  6       0.009711 25 32  1   T    0.010000
DX     Comp  7       0.008562 25 32  1   T    0.010000
DX     Comp  8       0.004396 25 32  1   T    0.010000
DX     Comp  9       0.001465 25 32  1   T    0.010000
DX     Comp 10       0.000907 25 32  4   T    0.010000
NLStep= 0 lin=  7 aot=   27.27 Rmax=0.8514E+00 Rsum=0.2739E-03 egain=0.2624E-01
 Iteration            1 linears req            5
DX Pressure  0      -0.563835 25 32  1   T    1.592056
DX     Comp  1       0.000035 25 32  1   T    0.010833
DX     Comp  2       0.000063 25 32  1   T    0.010833
DX     Comp  3       0.001887 25 32  1   T    0.010833
DX     Comp  4       0.000242 25 32  1   T    0.010833
DX     Comp  5       0.000111 25 32  1   T    0.010833
DX     Comp  6       0.000270 26 32  1   T    0.010833
DX     Comp  7       0.000238 26 32  1   T    0.010833
DX     Comp  8       0.000127 26 32  1   T    0.010833
DX     Comp  9       0.000044 26 32  1   T    0.010833
DX     Comp 10      -0.000317 25 32  2   T    0.010833
NLStep= 1 lin=  5 aot=    0.38 Rmax=0.1921E-01 Rsum=0.1448E-04 egain=0.3165E-01
 Max changes:pres       40.6      25 32  4 temp       0.00       0  0  0
             oil satn  0.516E-01  17  7  1 gas satn -0.178E-01  17  8  1
             wat satn -0.939E-03  25 32  4 eng dens   0.00       0  0  0
 Throughput ratio:avrg    0.404E-01 max    0.192                26  32   2
 MIF ;  103.0 9.00 6.9094 .01725 8973.3 157.46 62000. 3531.4    0.0 60500. 2 2%
```

This output shows two non-linear iterations leading to a timestep report. The first line:

```
Iteration            0 linears req            7
```

tells us that the iteration 0 (the initial estimate) needed 7 linear iterations to solve the linear problem.

The next 11 lines consist of one line for each of the solution variables showing the largest change in that variable in any cell during this iteration. The solution variables for each grid block are the pressure in the grid block, the molar density of each hydrocarbon component, and a water term. This model has 9 hydrocarbon components. Water is written as component 10.

```
DX Pressure  0     -40.075969 25 32  4   F    1.469590
DX     Comp  1      -0.000375 25 32  1   T    0.010000
DX     Comp  2      -0.000980 25 32  1   T    0.010000
DX     Comp  3      -0.025419 25 32  1   F    0.010000
DX     Comp  4      -0.002318 25 32  1   T    0.010000
DX     Comp  5      -0.000318 25 32  3   T    0.010000
DX     Comp  6       0.009711 25 32  1   T    0.010000
DX     Comp  7       0.008562 25 32  1   T    0.010000
DX     Comp  8       0.004396 25 32  1   T    0.010000
```

```
DX     Comp  9       0.001465 25 32  1   T    0.010000
DX     Comp 10       0.000907 25 32  4   T    0.010000
```

In each of these lines, DX means the solution change. The first DX line is the pressure change. The largest pressure change was an increase of 40.075969 psi in cell (25,32,4), which happen to contain an injecting completion. The 'F' on that line means 'False,' in that the pressure variable has not converged, as the pressure change is greater than 1.469590, which is the maximum pressure change, allowed for convergence for this iteration.

The second DX line shows the largest change in the molar density, expressed as a saturation equivalent, for component 1. The increase of 0.000375 was in cell (25, 32, 1), and is less than the convergence maximum of 0.01, so the component 1 variable is considered to be converged. In fact all the components have converged except for component 3.

The non-linear iteration however has not converged since two of the variables (pressure and component 3) are not yet converged. The next line is a summary of the first iteration (iteration 0).

```
NLStep= 0 lin=  7 aot=   27.27 Rmax=0.8514E+00 Rsum=0.2739E-03 egain=0.2624E-01
```

NLStep=0 means that this in non-linear step 0

lin=7 means that 7 linear iterations were needed to solve it

Rmax= 0.8514E+00 is the worst (maximum) absolute residual at the beginning of this iteration

Rsum= 0.2739E-03 is the sum of all the residuals at the beginning of this iteration

egain= 0.2624E-01 is a gain factor calculated from previous timesteps

The next line

```
Iteration            1 linears req            5
```

is the start of the report on the second non-linear iteration, Iteration 1, which needed 5 linear iterations.

The next 11 lines are the solution changes. They are similar to the reported changes for the first non-linear iteration except that now both the pressure and component 3 have changed by less than the new convergence criteria.

The non-linear iterations have now converged.

```
NLStep= 1 lin=  5 aot=    0.38 Rmax=0.1921E-01 Rsum=0.1448E-04 egain=0.3165E-01
```

is the report of the converged iteration showing that the maximum residual at the beginning of that iteration was down to 0.1921E-01 and the sum of residuals was down to Rsum=0.1448E-04. A report follows with the changes during that timestep.

```
Max changes:pres       40.6      25 32  4 temp       0.00       0  0  0
            oil satn  0.516E-01  17  7  1 gas satn -0.178E-01  17  8  1
            wat satn -0.939E-03  25 32  4 eng dens   0.00       0  0  0
```

The maximum pressure and saturation changes are reported, as well as the cells in which this change occurred. In the case of thermal runs, the maximum temperature and energy density changes are also reported.

The maximum throughput is reported next. Throughput is defined as the volume flowing through a cell divided by the pore volume of the cell. If the throughput is too high (say higher than 0.5) it could cause convergence problems, and the pore volume of the cell that has the highest throughput should be examined.

```
Throughput ratio:avrg    0.404E-01 max    0.192               26  32   2
```

The last of these lines is the report of production, and so forth for the time step.

```
 MIF ;  103.0 9.00 6.9094 .01725 8973.3 157.46 62000. 3531.4    0.0 60500. 2 2%
```

RPTRST CONV

You can use a visual display of the grid blocks causing convergence problems. Adding `CONV` to the `RPTRST` keyword sends two new outputs to the restart files. Each cell will have two new variables that will be used to count the number of times that cell has been one of the most difficult cells to converge.

At the beginning of the simulation, each cell will have its counter set to zero and, at every timestep, the most difficult cells (10 by default for ECLIPSE 300) will have their counter increased by 1. At the end of the run, you can display the cells with the most problems.

The outputs in the restart file for ECLIPSE 100 are:

- CNV_GAS, CNV_OIL and CNV_WAT indicate the worst cells base on fluid saturations.
- CNV_PRE indicates the worst cells base on pressure updates.

The outputs in the restart file for ECLIPSE 300 are:

- CONV_VBR indicates the worst cells base on volume balance residual.
- CONV_PRV indicates the worst cells base on pressure updates.

4

Important information from an ECLIPSE simulation

The progress of a simulation can normally be followed by monitoring the summary vector information requested in the dataset, and viewing the 3D field information (e.g. for pressure and saturations). However if there are issues with the performance or accuracy of the run, it may be useful to inspect its PRT file. This file can be found in the same folder as the input dataset, and will have the same root name (so deck X.DATA will produce the output file X.PRT.)

The PRT file is used to report initialization data in printed form. It will also record current information on overall fluids in place and material balances. Importantly, it also contains information and warning messages from the simulation, and reports of the timestepping and current convergence behavior, as has already been seen in the convergence chapter.

Review reported messages in the PRT file

ECLIPSE outputs a large variety of messages about your dataset – information about the simulation run, errors within the data and errors encountered during simulation. You need to know what should you do about them: which ones are for information only, and do not necessarily need any action, and which ones do need action. You can also ask ECLIPSE to produce reports, of the current state of the data and of the current performance, showing how both linear and non- linear iterations are proceeding and the methods of timestep selection. These messages are recorded permanently in the PRT file and log file, and are echoed to the console screen and made available for inspection by Petrel or the Simulation Launcher. This chapter describes how to inspect the PRT file in a text editor.

The PRT file records the progress of the run and will tell you about timesteps and report steps and the state of the data. It also contains particular alerts which are generally referred to as messages. These come in 6 levels of severity. All the messages start with @ in the files, so you can find all these messages by looking for @. You should start by checking messages (severity 1), comments (severity 2) and warnings (severity3), and then correct any problems (severity 4) and errors (severity 5). There is also a bug, or fatal error level (severity 6) that represents an uncaught error. Examples of all these message types are:

Messages

These give information about the run and should be checked to make sure the content is what you expect. Here the messages tell you that the license check has been performed and that a certain size of memory has been required to process the current data:

```
@--MESSAGE  AT TIME        0.0   DAYS    (19-OCT-1982):
@           CHECKING FOR LICENSES
@--MESSAGE  AT TIME        0.0   DAYS    (19-OCT-1982):
@           THE MEMORY REQUIRED TO PROCESS THE GRID SECTION IS
@           1526 KBYTES
```

Comments

These contain information about potential problems that may affect the run:

```
@--COMMENT  AT TIME        0.0   DAYS    (19-OCT-1982):
@           NO NON-NEIGHBOUR CONNECTIONS FOUND
```

If you did not expect non-neighbor connections, this comment can be ignored. If you know that your reservoir has a lot of fault throws, then you may need to check your model.

Warnings

These are possible data errors and should be checked as there may be an issue to be corrected:

```
@--WARNING  AT TIME        0.0   DAYS    (19-OCT-1982):
@           INCONSISTENT END POINTS IN SATURATION TABLE   1
@           THE MAXIMUM GAS SATURATION (1.0000)
@           PLUS THE CONNATE WATER SATURATION (0.1200)
@           MUST NOT EXCEED 1.0
```

This warning may not be an issue, but the gas and water SCAL tables are inconsistent; using this data means the gas saturation will never be greater than 0.88 as the coonate water saturation is 0.12.

```
@--WARNING  AT TIME        0.0   DAYS    (19-OCT-1982):
@           THE BOTTOM HOLE PRESSURE LIMIT FOR WELL INJECTR1
@           HAS BEEN DEFAULTED. THE DEFAULT VALUE IS  100,000 PSIA
```

This warning could possibly be safe - as long as the injector never goes to BHP control, but setting a more reasonable limit would be preferable.

Problems

Any problems reported mean there is definitely an issue that must be checked and fixed before running the simulation:

```
@--PROBLEM  AT TIME        0.0   DAYS    (19-OCT-1982):
@           OIL   PRESSURE IN EQUILIBRATION CALCULATION BETWEEN DEPTHS
@           8400.0   FEET   AND    8397.7   FEET   HAS NOT CONVERGED.
@           CONVERGENCE ERROR =   0.86508E+33    PSI.
@           THE CURRENT NUMBER ( 100 ) OF DEPTH NODES IN THE
@           EQUILIBRATION CALCULATION IS TOO SMALL
```

In this example, there is a problem in either the specified depth nodes or oil PVT table.

Errors

This means there is an error in the data and you cannot continue with a simulation run without fixing it. Note that the simulation may not stop immediately, and that other errors may occur, which may or may not be due to the previous error.

```
@-- ERROR  AT TIME        0.0   DAYS    (19-OCT-1982):
@           ERROR IN PVTO TABLE NUMBER   1
@           NOT ENOUGH PRESSURE VALUES (ONLY 1) SPECIFIED
@           FOR HIGHEST RS (=   1.270 MSCF/STB)
```

Errors may also result in other messages. This error is the underlying cause for the problem reported in the previous section. You should always review and fix errors before looking for the solutions to later messages. Previous warning messages may help you to fix the current error.

An error may result in "NaN" (Not a Number) being displayed as part of a message and is usually caused by dividing by zero or something similar.

```
@--WARNING  AT TIME        0.0   DAYS    (19-OCT-1982):
@           GAS IS DENSER THAN OIL   WITHIN THE RESERVOIR
@           IN EQUILIBRATION REGION  1.  CHECK SURFACE
@           DENSITIES, FORMATION VOLUME FACTORS, RS, RV.
@           THE PROBLEM OCCURS AT DEPTH    8725.0000
@           GAS   PRESSURE =   0.4201E-01
@           OIL   PRESSURE =  NaN
@           (RS MAY BE SPECIFIED IN THE WRONG UNITS)
```

If you see "NaN" anywhere in any output, then:

a. Fix any Error or Problem first.

b. If the "NaN" is still there, contact your local support.

Bugs

A bug is an indication of an internal inconsistency in ECLIPSE.

"Bugs" should not occur unless there has previously been an error; you should therefore fix any error or problem before reporting the problem. However, if the bug is still occurs, contact your local support.

5

Ways to improve simulation performance

Improving the simulation performance means making it run as efficiently as possible for a given target accuracy. Before dealing with the data input, you can consider hardware solutions include ensuring that other jobs are not running on the machine, using a faster machine, increasing the memory in the machine, and running in parallel. You can reduce the size of the model, through reduced grid size, and reduce the complexity reduced by using simplified physics models.

For the compositional simulator, the number of components should be considered. It may be possible to use a smaller number of pseudo-components to represent the phase diagram of the reservoir fluids.

Importantly, the input data should cover the expected range of the solution variables, as otherwise data may be extrapolated incorrectly from the data provided. It is important to remember that, while your final solutions may lie within expected bounds, the iterative solution method may explore wider ranges of data in order to reach convergence, and your data should include some allowance for this, so that any extrapolations are physically reasonable. Of course the data should be consistent and of reasonable quality.

This chapter describes what can be done in different sections of the DATA deck (`GRID`, `PROPS`, `SOLUTION` and `SCHEDULE`) to make the simulator run faster and in less memory. The issues discussed here are not exhaustive but are known to cause problems if not considered.

GRID section

A reservoir or field model is divided into grid cells which divide the model into areas of interest, both in terms of structures (such as faults or horizons) and properties. However, these requirements can vary – what is of great interest to a geologist may not be of so much importance to a reservoir engineer. To get the best performance from a simulation run requires identifying the parts of a model that are less important, and reducing the level of detail in those areas within the model. These changes may include:

- Making small volume cells inactive by setting a minimum pore volume that must be exceeded.
- Coarsen areas of the model that are of less interest for simulation.
- Remove isolated cells within a model that do not contribute to the connected (i.e. producible) fluid volume.

Such changes are made in the `GRID` section.

Note: To view and analyse the grid structures and properties, use a visualization program such as Petrel.

Minimum pore volume for active cells

Differences in the volumes of neighboring grid blocks may cause convergence issues where volume change in one cell leads to a disproportionate change in the neighboring cell. Where these smaller cells exist in the model, they are not likely to add to the model and could be removed using the `MINPV` keyword to help speed up the simulation run.

When removing cells, take care to choose a suitable value of `MINPV` that does not significantly change the total pore volumes in each region, and that does not remove high-permeability streaks or thin shale layers within the reservoirs. If `MINPV` is being used to remove pinch-outs, also use `PINCH` to connect across those thin cells that represent pinch-outs. Within these limitations, using `MINPV` should both improve performance and give unchanged production results.

If a full-field model has convergence problems and does NOT have a minimum pore volume, adding `MINPV` is the most likely change that will improve performance.

Coarsen areas of less interest in a model

Where a model contains many grid cells in areas of lesser interest, the model can be modified to reduce the simulation time by reducing the number of cells. Typically you place more cells in areas of importance to capture detail for the simulation. You can a Local Grid Refinement in these areas – this is often done in the neighborhood of a well. Alternatively in areas of less interest, cells can be merged to create fewer and larger cells. This process is called coarsening. Finally, certain kinds of cells can be replaced by other kinds of modelling. For example, if your model contains a large number of water-filled cells, cells could either be coarsened or replaced with an aquifer.

Remove cells that do not contribute to the simulation

A model should include cells that contribute to the overall simulation run. If the model contains isolated cells, such cells are still included in the calculation – that is, they contribute to the total size and run time – and their contribution can distort average reservoir pressure, but they do not contribute to the rest of the solution. Where the model contains isolated cells, you can identify and filter-out these cells visualizing the grid in Petrel to help determine which cells to remove. Fig. 3.1 shows a grid image with clusters of individual or small numbers of isolated cells detached from the main body of the active grid.

Visualizing the grid in Petrel can help determine which cells to remove. For a workflow showing how to remove isolated cells with Petrel, see "Remove unwanted isolated cells".

Figure 5.1. Isolated cell clusters detached from the main active grid

PROPS section

The PROPS section describes the physical properties (densities, viscosities, compressibilities, relative permeabilities and so on) for the fluids and rocks in the simulation. It also checks the data and integrates it with the grid section data to calculate and check extensive fluid and rock quantities.

Total compressibility check

In black oil models, both ECLIPSE 100 and ECLIPSE 300 check for positive compressibility of each single reservoir fluid as the PVT data is read – that is, the formation volume factor must be a monotonically decreasing function of pressure assuming all other variables are held fixed.

ECLIPSE also checks that mixtures of saturated oil and gas have a positive total compressibility even when there is mass transfer between the two phases. For example, increasing the pressure of a cell that contains both oil and gas will:

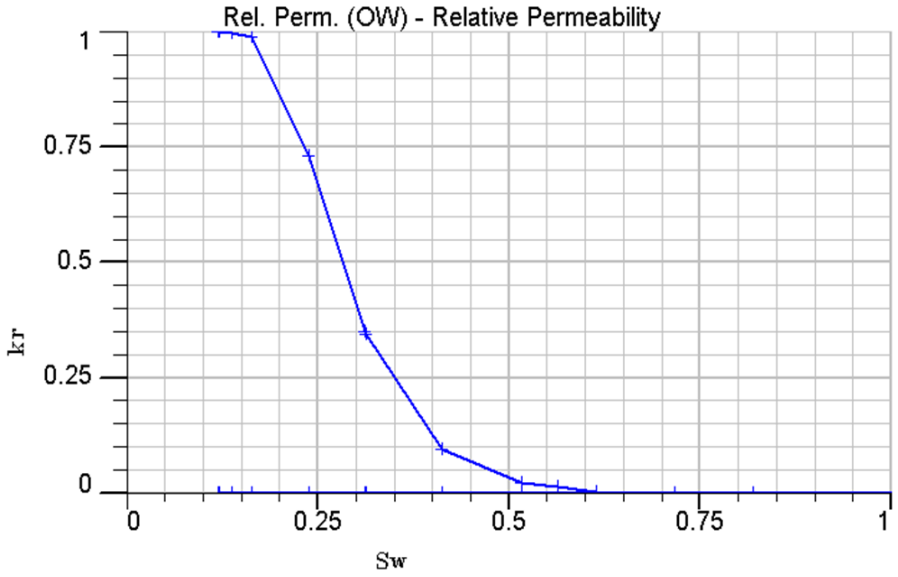
- transfer some gas from the gas phase to the oil phase

- swell the oil due to extra dissolved gas
- decrease the remaining gas volume due to compression

The oil volume will therefore increase with increasing pressure, and the gas volume will decrease with increasing pressure. The total (oil+gas) volume must however decrease. This will only happen if the reduction in the volume of gas is greater than the increase in the volume of oil.

Relative permeability check

Poor convergence may be caused by erratic relative permeability information. For example, a plot of the relative permeability curve for oil against water may appear accurate. A plot of the original curve is shown below:



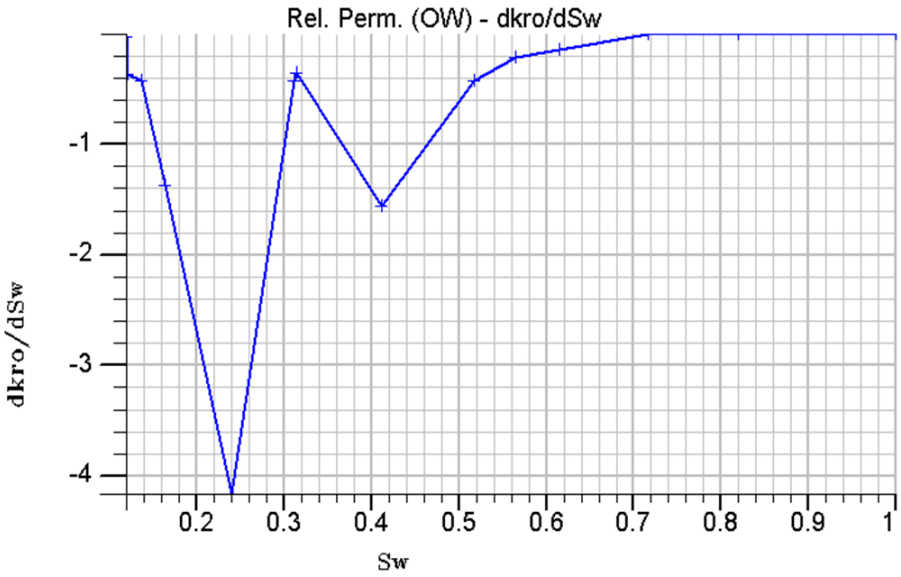
The curve is smooth and looks reasonable. At first, the data in the relative permeability table also looked reasonable. The data is entered using the SOF3 keyword:

```
SOF3
--  SOIL      KROW      KROG
    0.181     0         0
    0.283     0.0001    0.0001
    0.385     0.0015    0.0015
    0.436     0.0124    0.0124
    0.483     0.0217    0.0217
    0.588     0.0939    0.0939
    0.686     0.3499    0.3499
    0.689     0.3501    0.3501
    0.761     0.7323    0.7323
    0.837     0.9887    0.9887
    0.863     0.9978    0.9978
    0.879     1         1
    0.880     1         1    /
```

The S_{oil} and k_{row} column seem to be correct:

- The S_{oil} column is monotonically increasing, and has values between 0 and 1.
- The k_{row} column is also monotonically increasing, and has values between 0 and 1.

When the derivative $d(k_{row})/d(S_{oil})$ is plotted, the expected derivatives should be smoothly varying as k_{row} was smoothly varying. The resultant graph is:

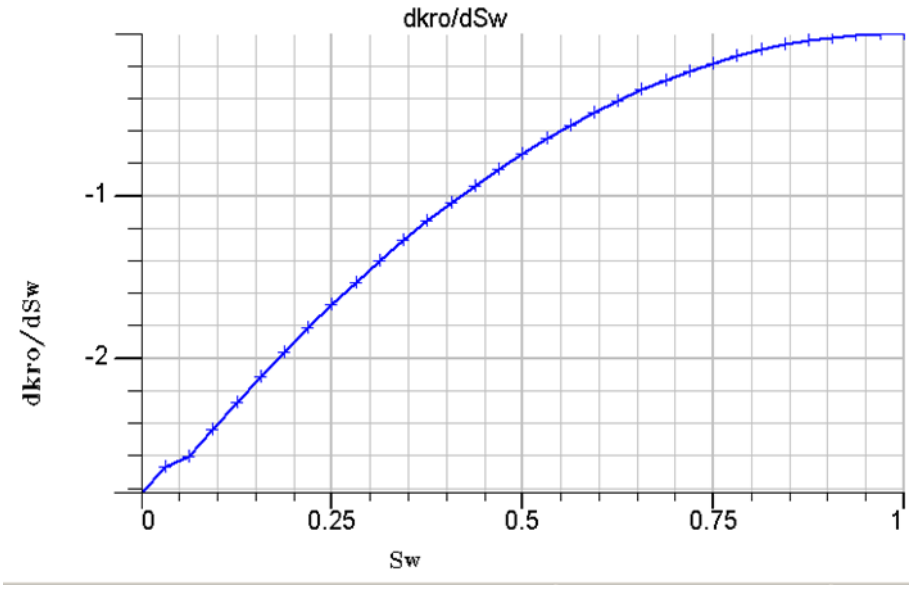


There are sharp changes in slope around $S_w = 24\%$ and around $S_w = 31\%$. These value correspond to S_{oil} of about 68% and 76% respectively. When the SOF3 table is checked at these values; at around 68% oil there are two oil saturations with very close values, 0.686 and 0.689, that have very similar k_{row} values of 0.3499 and 0.3501.

The SOF3 table does not need saturation values that are close together, and in fact very closely spaced values in any table may cause ECLIPSE to do extra computing work as it has to find exactly which two point to interpolate between. In this case, to fix the problem remove either one of the two entries. Removing the row corresponding to the S_{oil} value of 0.686, gives the following table:

```
SOF3
--  SOIL     KROW     KROG
    0.181     0             0
    0.283     0.0001    0.0001
    0.385     0.0015    0.0015
    0.436     0.0124    0.0124
    0.483     0.0217    0.0217
    0.588     0.0939    0.0939
    0.689     0.3501    0.3501
    0.761     0.7323    0.7323
    0.837     0.9887    0.9887
    0.863     0.9978    0.9978
    0.879     1         1
    0.880     1         1    /
```

This change is enough to produce smooth derivatives and to resolve any convergence or performance problem:



Reducing chopping

Any non-zero value of the number of chops NSCHP or NCHOP shown in the convergence reports in the PRT file increases material balance errors for the subsequent non-linear iteration and therefore reduces the chances of convergence. Some saturation chops can be avoided by adjusting relative permeability curves in such a way that the critical saturation is not the same as the lowest saturation value in the table.

For instance, instead of

```
SWFN
    0.2   0       7
    0.3   0.07    4
    0.4   0.15    3
    0.5   0.24    2.5
    0.6   0.33    2
    0.8   0.65    1
    0.9   0.83    0.5
    1     1       0      /
```

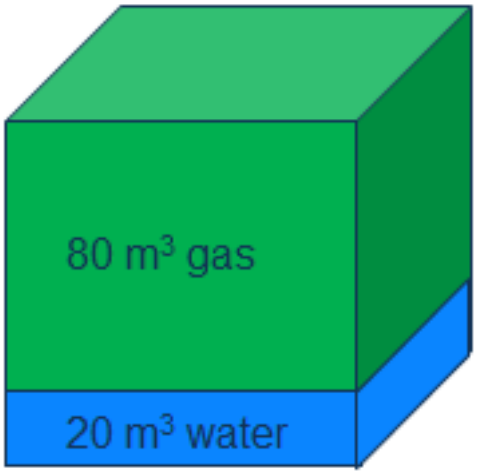
try using

```
SWFN
    0.2   0       7
    0.21  0       1*
    0.3   0.07    4
    0.4   0.15    3
    0.5   0.24    2.5
    0.6   0.33    2
    0.8   0.65    1
    0.9   0.83    0.5
    1     1       0      /
```

The new saturation value at 0.21 may help convergence. It will not affect the initial fluids-in-place but will unfortunately slightly reduce the water mobility for water saturations between 0.2 and 0.3. This may not be important to engineering accuracy.

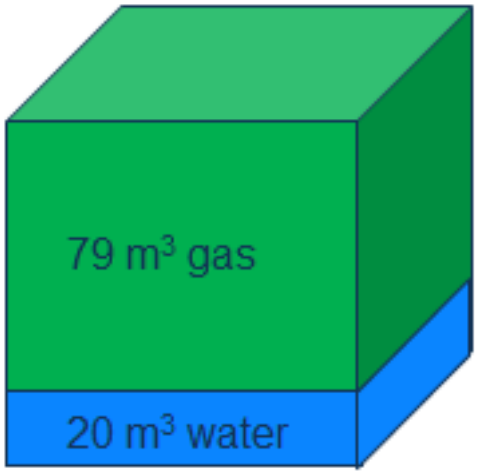
Pressure effects on critical saturation

Consider a grid block with pore volume of 100 m^3. Assume it contains 80 m^3 of gas and 20 m^3 of water. The water saturation is 20%:



In the case where there is no flow in or out of that grid block, and the pore volume decreases by 1% due to reservoir pressure changes:

- The gas will compress but water is relatively incompressible, so there will be approximately 79 m^3 of gas and 20 m^3 of water.
- Water saturation will increase from 20% (20 m^3 of water in 100 m^3 pore volume) to 20.2% (20 m^3 of water in 99 m^3 pore volume)



As you produce/inject, the pressure in every block will change and therefore the pore volume of each grid block will change.

Assuming the initial water saturation (S_{WL}) in each grid block was 20%, S_w will now be greater than 20% in many blocks. If $S_{WCR}=S_{WL}=20\%$, then all those blocks have mobile water even if S_w is only 20.0001%, and this water WILL flow.

Setting $S_{WCR}=S_{WL}+0.01\%$ will stabilize the model without changing the OOIP.

This can be applied to the critical saturation of any phase; using the original SOF3 table:

```
SOF3
--  SOIL     KROW     KROG
    0.181     0                0
    0.283     0.0001    0.0001
    0.385     0.0015    0.0015
    0.436     0.0124    0.0124
    0.483     0.0217    0.0217
    0.588     0.0939    0.0939
    0.689     0.3501    0.3501
    0.761     0.7323    0.7323
    0.837     0.9887    0.9887
    0.863     0.9978    0.9978
    0.879     1         1
    0.880     1         1    /
```

The oil can be stabilized by changing the 1E-4 values at 18.1% oil saturation to zero. If this is not the preferred solution, you could add a relative permeability of 0 at an oil saturation of 18.2%

```
SOF3
--  SOIL     KROW     KROG
    0.181     0         0
    0.182     0         0
    0.283     0.0001    0.0001
    0.385     0.0015    0.0015
    0.436     0.0124    0.0124
    0.483     0.0217    0.0217
    0.588     0.0939    0.0939
    0.689     0.3501    0.3501
    0.761     0.7323    0.7323
    0.837     0.9887    0.9887
    0.863     0.9978    0.9978
    0.879     1         1
    0.880     1         1    /
```

SOLUTION section

The SOLUTION section of an ECLIPSE dataset contains the data used to define the initial state (pressure, saturations and compositions) of the reservoir. An important part of its function is to calculate the initial fluid in place values (the initial oil, water and gas saturations) in each grid block. The initial state may either be specified for each block (referred to as enumeration); this may not always yield a solution that is stable when the simulation starts. The alternative is to let the simulator work out the equilibrium conditions for the specified fluids – this is referred to as equilibration.

There are three methods that can be used to perform equilibration; they vary in simplicity, stability and accuracy. The method is selected using the ninth argument of the EQUIL keyword that specifies the number of sub-layers (N) used to obtain the initial saturations:

Center-point equilibration (EQUIL item 9 = 0)

The simulator sets the fluid saturations in each grid block according to the conditions at its center. This method gives a stable initial state since the phase pressure differences in the simulation are also taken between cell centers. It is however the least accurate method, particularly in cases where the fluid contact passes through large grid blocks.

Horizontal fine grid equilibration (EQUIL item 9 < 0)

The top and bottom halves of each grid block are each divided into -N layers of equal thickness, and the saturations are determined locally in each layer. The phase saturations for the block are the average of the saturations in each layer. This option provides a more accurate calculation of

the fluids in place, but may yield a solution that is not completely stable in the initial state. The size of EQUIL item 9 determines N the number of layers, and a negative value is used to signal this option.

Tilted block fine grid equilibration (EQUIL item 9 > 0)

This option is similar to horizontal fine grid equilibration, but it takes into account the slope of each grid block. The top and bottom faces of the blocks are treated as planes that are tilted about their central points. The top and bottom halves of the grid block are each divided into N horizontal layers of equal thickness, but the thickness of the layers in the top half is generally different from the thickness of the layers in the bottom half. This is because the distance from the block center line to the highest corner is not the same as the distance from the block center line to the lowest corner, if the upper and lower faces have different tilts. The phase saturations in each block are calculated as a weighted average of the saturations in the layers, weighted according to the area of each layer that is enclosed within the block multiplied by the layer thickness. This option provides the most accurate calculation of fluids in place, but may yield a solution which is not completely stable in the initial state. The size of EQUIL item 9 determines N the number of layers, and a positive value is used to signal this option.

With either of the fine grid equilibration methods there is a redistribution of fluids between grid blocks near the contacts when the simulation begins, which occurs independently of any external driving force (such as wells). The reason is that a steady state solution on the fine equilibration grid (in which each block is subdivided into several layers) is not necessarily a steady state solution on the coarser simulation grid.

SCHEDULE section

The SCHEDULE section specifies the operations to be simulated as time advances from the initial reservoir state. In addition to the convergence issues discussed in earlier chapters of this document, two important functions of the SCHEDULE section are:

- To control the overall timestepping and to match this to the reporting requirements
- To place, control and constrain the wells. Guidance on general well modelling is given in the *"ECLIPSE Technical Description"* document. This chapter discusses particular issues related to multisegment wells.

Timestepping and iterations

This section describes:

- report steps and performance
- how to extract information from the PRT file for ECLIPSE 100 and ECLIPSE 300
- timestepping selection options
- timesteps and non-linear iteration contrl
- the gain option in ECLIPSE 300

Report steps and performance

Usually the timing of report steps can be made on the basis of what your project needs, and does not affect the overall performance of the simulator. The number of report steps and the time between report steps will depend on the type of model that you are simulating:

- For a prediction or forecasting run lasting for instance 30 years, you may ask for monthly reports for the first year, quarterly reports for the next five years, and yearly reports for the rest of the simulation.
- For history matching you may ask for weekly reports for the first year and for monthly reports for the remainder of the history match, to test the validity of your model on a finer time scale.
- For slim-tube experiments, reporting intervals are likely to be minutes and hours.

However, if the number of reports being asked for is large, or if the amount of computational time used to reach a report time is small, then the overhead of many reports may be too great. Computational time may be reduced by changing the requested reports if:

- You are asking for more reports than you actually need.
- Each report step is being reached in just a single timestep. Savings could also be made if the report step is taking 2 or 3 iterations.

For example, if you have created a file `BASE.STEPS100` (see "ECLIPSE 100" below) and find that it mainly contains lines of the form:

```
STEP 40 TIME= 400.00 DAYS ( +10.0 DAYS REPT 1 ITS) (4-FEB-2003)
STEP 41 TIME= 410.00 DAYS ( +10.0 DAYS REPT 1 ITS) (14-FEB-2003)
STEP 42 TIME= 420.00 DAYS ( +10.0 DAYS REPT 1 ITS) (24-FEB-2003)
STEP 43 TIME= 430.00 DAYS ( +10.0 DAYS REPT 1 ITS) (6-MAR-2003)
```

then each timestep is a report step (because after the timestep size +10 DAYS, the reason for this choice of size is given as REPT – reaching the report time), and each timestep is solved in a single iteration (given after the report step reason as 1 ITS), so the run may go two or three times faster if you allow the simulator to produce report steps once a month instead of once every 10 days.

The example above was for an ECLIPSE 100 run. The same applies for an ECLIPSE 300 run with:

```
Rep ; 400.0 10.0 8.7838 .19498 1.4E05 32884. 1.2E06 4843.6 .00000  1.3E06 1
Rep ; 410.0 10.0 8.7838 .19498 1.4E05 32884. 1.2E06 4843.5 .00000  1.3E06 1
Rep ; 420.0 10.0 8.7838 .19498 1.4E05 32884. 1.2E06 4843.4 .00000  1.3E06 1
Rep ; 430.0 10.0 8.7838 .19498 1.4E05 32884. 1.2E06 4843.3 .00000  1.3E06 1
```

where Rep gives the timestep selection reason and the final 1 gives the number of iterations.

The next sections show how to extract detailed information from the PRT file for the different formats of the two simulators.

Extract information from the ECLIPSE 100 PRT file

From the log file extract the step information into a separate file that contains one line for each timestep. For example, on a Linux system, with an ECLIPSE 100 log file called `BASE.LOG`, use the following command to create a file `BASE.STEPS100`:

```
grep STEP BASE.LOG > BASE.STEPS100
```

The file `BASE.STEPS100` contains one line for each timestep of the form:

```
STEP   15  TIME=400.00 DAYS ( +30.0 DAYS REPT  3 ITS)  (4-FEB-2003)
```

Where

"STEP 15"	means this is the 15th timestep.
TIME= 400.00 DAYS	means there have been 400 simulated days since the beginning of the simulation.
+30.0 DAYS	shows that the latest timestep was of 30 days.
REPT	is a mnemonic explaining why 30 days were chosen, and means that a report step has been reached.
3 ITS	mean 3 non-linear iterations were needed to solve the 30 day timestep.
(4-FEB-2003)	is the current simulation date.

Extract information from the ECLIPSE 300 PRT file

From the log file extract the step information into a separate file that contains one line for each timestep. For example, on a Linux system, with an ECLIPSE 100 log file called BASE.LOG, use the following command to create a file BASE.STEPS300:

```
grep ";" BASE.LOG > BASE.STEPS300
```

The file BASE.STEPS100 contains one line for each timestep of the form:

```
Rep ;  400.0 30.0 8.7838 .19498 1.4E05 32884. 1.2E06 4843.6 .00000 1.3E06 3
```

Where

Rep is the mnemonic that shows that a report step has been reached,

30.0 shows that the latest timestep was of 30 days,

the next 8 numbers show the GOR; water cut; oil, gas, and water production rates; average field pressure; gas and water injection rates,

3 at the end of the line indicates that three non-linear iterations were required to solve the 30-day timestep.

If the AIM option is used then the line will have an extra number at the end, for example:

```
Rep ;  400.0 30.0 8.7838 .19498 1.4E05 32884. 1.2E06 4843.6 .00000 1.3E06 3 2%
```

2% shows the percentage of the cells that was solved fully implicitly.

Timestep selection options

E100	E300	Explanation
INIT	Init	First timestep
MAXF	MIF	Maximum increase factor
REPT	Rep	Report step

E100	E300	Explanation
HREP	Hrep	Half step to report
CHOP	Redu	Timestep chopped
DIFF		Follows CHOP
TRNC	TTE	TTE limit
	SCT	Solution change
	TPT	Throughput limit

A more complete list is available in the "convergence" section of the *ECLIPSE Technical Description*.

Timestep sizes and non-linear iteration control

The control parameters for choosing timestep sizes and numbers of non-linear iterations per timestep are given sensible defaults, and usually it is preferable to leave these unchanged. In a few cases adjustments can be made to these parameters to improve performance. It may be possible to take somewhat longer timesteps without compromising the desired accuracy of the solution or creating convergence problems.

The *Gain* option in ECLIPSE 300

Gain is an option, not used by default, which can significantly improve performance. The idea is to speed the code up by not taking an extra nonlinear iteration if that iteration is likely to generate a very small solution change. ECLIPSE can predict the behavior of the next iteration based on the history of previous time steps.

Note: This option should be used with care, as there may be cases in which this option can lead to dramatic changes in the solution. Always check against a run that does not use this option.

ECLIPSE calculates a gain factor from previous time steps and, rather than using the AOT, uses

```
EAOT=MIN(2.0*AOT*EGAIN,AOT)
```

Where:

`EAOT` Effective AOT.

`AOT` The AOT as calculated above.

`EGAIN` A measure of the expected improvement at the next step.

The gain value is printed out with the non-linear debug, as in the preceding example.

The gain option can be activated using switch 68 of the `OPTIONS3` keyword. If you set the switch 19 of the `OPTIONS3` keyword to 1, the gain will not be used if any cell has changed state during the iteration. Should this occur, you might expect the previous history to be less valid.

Production controls through multisegment wells

As the whole system being modelled is highly coupled between the surface system, the wells and the reservoir, convergence issues with datasets including multisegment wells are normally a manifestation of issues elsewhere in the dataset. This means that although you see a problem in your well, it might be caused by a non-linearity in your grid or your fluid data. Therefore, you should first QC the whole dataset before narrowing down your search to find specific problems in the multisegment well itself.

To QC your model:

1. QC your grid and your fluid data (PVT or relative permeability values) as the system is highly coupled and although you might see a problem in your well, it might be caused by a non-linearity in your grid or your fluid data.
2. Ensure your multisegment well is correctly defined.

   Note: Defining multisegment wells by hand is extremely error-prone and so it is recommended to use Petrel instead, using the **Define well segmentation** process.

   If you have defined the multisegment wells by hand, you can visually check the branches and numbering in the **Well section window** in Petrel. Branches and numbering can be switched on by right clicking on the segmentation set in the **Input** pane, and checking on the **Show segment info** option in the **Style** tab.

Looped flow paths

Check your multisegment well for looped flow paths as these can often cause problems; see Loop flow paths in the *ECLIPSE Technical Description* for more details.

In Petrel, using the Suppress annular segments option in the Define well segmentation process can remove looped flow paths, particularly for completions using inflow control devices.

If looped flow paths are necessary for your model, you can use the `WSEGSOLV` keyword to set the parameters affecting the multisegment well iterative linear solver. Items 2 and 4 have more influence on the solution convergence than the other items and increasing these by a factor of 5 is reasonable depending on the complexity of your model and the computation time you wish to spend. Items 7 and 8 in `WSEGSOLV` might also help to control any spurious high flows that can be generated during the non-linear solving of looped multisegment wells.

Ways to modify multisegment well information

If you have checked your grid, your fluid data and your segmentation and you are still experiencing convergence issues, the next thing to try is to reduce the non-linearity in your model to give you a clue as to what might be causing the convergence problem. The non-linearity can be reduced by making the following changes, (these are in order of what should make most effect and so should be tried first):

- Turn off drift flux in any of the segments and use the homogeneous well model instead (item 7 in `WELSEGS`, or clear the **Phase slip** check box in the Petrel **Define well segmentation** process). If the model gives significantly different results with and without phase slip (for any time period of the simulation which does not have convergence issues) then a VFP table could be used instead to derive the pressure drop along the segments.

- Reduce the number of segments in each well. In the Petrel Define well segmentation process this can be done by increasing the minimum segment length.
- Use a conventional well model for the vertical section of the well. In the Petrel Define well segmentation process this can be done by unchecking the Segment up to (SSTVD) option. If this is unchecked, by default, the well will be segmented to the top connection/device in the well. Alternatively, as stated above, a VFP table could be used to model the flow to surface.
- Turn off the friction and acceleration pressure drop calculations (item 6 in WELSEGS, or clear **Friction** and **Acceleration** in the Petrel **Define well segmentation** process).
- Check in the RFT log if any of the completion devices have a particularly high pressure drop across them. If they do, try reducing the strength of these devices or removing them altogether.
- Turn off cross flow (item 10 in WELSPECS). The default in ECLIPSE is on, set to NO to turn off cross-flow.

Note that simplifying your model in these ways should only be done as an experiment to help you determine the root cause of your convergence problem. Your model should still include all the physics necessary for the problem, but trying some of these changes might help you pinpoint the problem and help you determine how much of the physics is necessary to include without overcomplicating the problem.

Other keywords that could help are:

- WSEGITER – this is a more expensive, but more robust iteration sequence, to give multisegment wells a better opportunity to converge. It involves relaxing the predicted pressure over an iteration by averaging it with the pressure from the last converged value. The relaxation becomes successively more severe if the well continues to not converge. When convergence is achieved, the relaxation is reduced again.

  Note: If you are using any tuning keyword, you must add WSEGITER after the tuning keyword.

- MXWSIT – By default, if the WSEGITER keyword is not entered, each well is allowed up to MXWSIT iterations. Increasing the value of MXWSIT in record 3 of the TUNING keyword may give your model a better chance of converging.
- WVFPEXP – if using VFP tables this can help for tubing head pressure (THP) controlled wells, especially those with fluctuating gas production.
- EXTRAPMS – if using VFP tables, use the EXTRAPMS keyword which gives a warning when extrapolating outside of the VFP table and could be giving physically unrealistic results. You can check your VFP table by opening the **VFP manager** in Petrel and using the quality control tools found there. More information on these tools can be found in the VFP quality control section of the Petrel help.
- WELSEGS item 8 in record 2 – check if the friction factor is too high. Lowering this may help convergence. You should aim to have default roughness for an open hole completion as the upper limit for roughness values for well equipment.

6

Petrel Workflows

The workflows in this section show how to perform some of the tasks outlined earlier in this guide for improving your simulation data.

Use an aquifer to simulate water cells in a model

If you have a large amount of water in your model, you can speed up a simulation run by replacing the water cells with an aquifer.

An aquifer can be used to simulate large amounts of water in a model. Adding an aquifer to your model allows you to focus on an area of interest in your model, exclude the water cells, and receive simulation results much faster.

You can add easily visualize and define the area of interest and then add an aquifer to your model using Petrel.

To use an aquifer in a simulation case you need to:

- Create a polygon that defines the area of interest in the model using the `Polygon editing` tools in Petrel.
- Create an aquifer using Petrel's `Make aquifer` process.
- Open the simulation case in the `Define simulation case` process and add the aquifer to the inputs on the `Grid` tab.
- Save and run the simulation case.

Note: Ensure that you have set up the simulation case in Petrel to use the maximum number of available processes. The case can be submitted to a queue on your local machine or cluster if one has been defined in the `System settings` dialog.

Define the area of interest

Begin by creating polygon that defines the area to be influenced by the aquifer. This is used as an input to the `Make aquifers` process.

1. Display the grid in a `2D window`.

2. In the `Models` pane, select the oil-water contact.

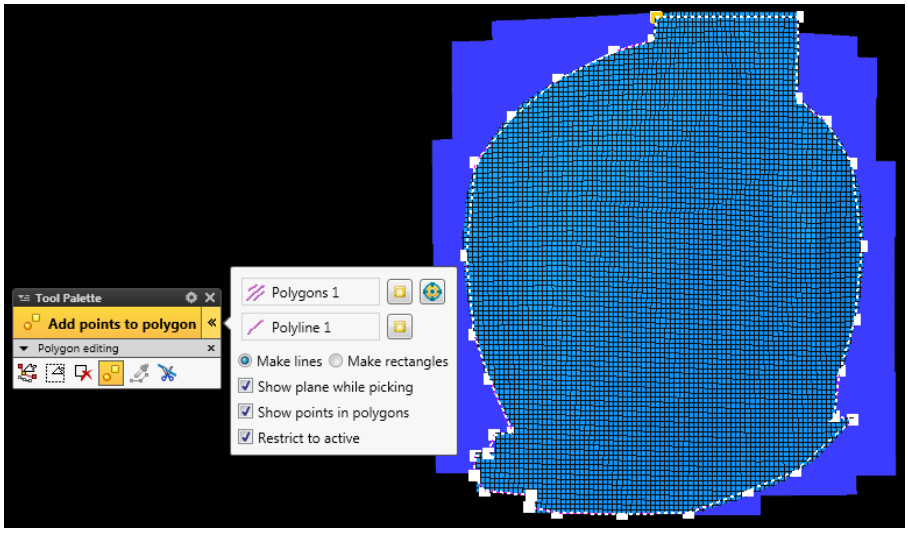
   Use the contact as a reference when creating the polygon defining the area of interest.

3. On the `Reservoir Engineering` tab, in the `Utilities` group, click **Polygon editing** to open the `Polygon editing Tool Palette`.

4. On the `Tool Palette`, click **Add points to polygon**.

   If there are other polygons in the model, click » to expand the `Tool Palette` and ensure that 'New' is displayed.

5. Digitize the points on the polygon. Double-click on the first point to close the polygon.



6. In the `Input` pane, right-click the polygon that you have just created, and then click **Settings**.

7. In the `Settings` dialog, click the `Info` tab, provide a more descriptive name for the polygon and click **OK**.

Create an aquifer to simulate the water in your model

In this step of the workflow, create an aquifer to include in the simulation.

Before running the `Make aquifer` process, ensure that you create a polygon that defines the area of interest using the `Polygon editing` tools in the `Utilities` group on the `Reservoir engineering` tab.

1. In the `Models` pane, select the grid.

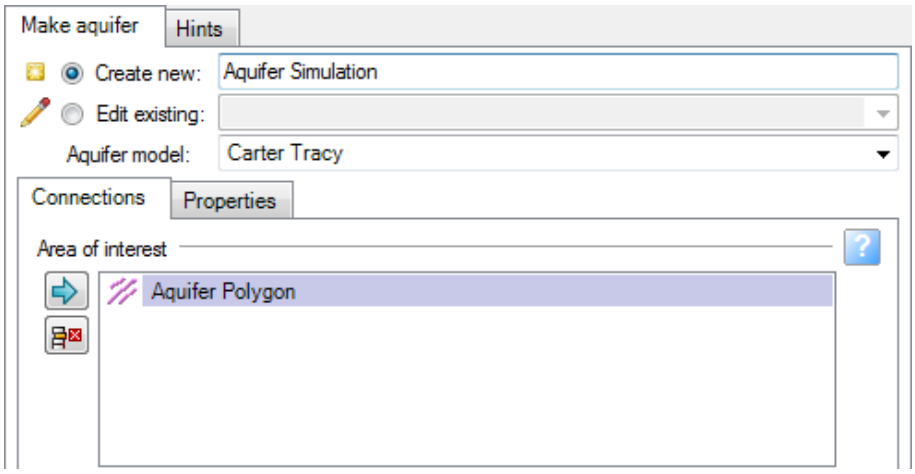
2. On the `Reservoir Engineering` tab, in the `Boundaries` group, click **Aquifers**.

3. In the `Make aquifer` dialog, click **Create new**.

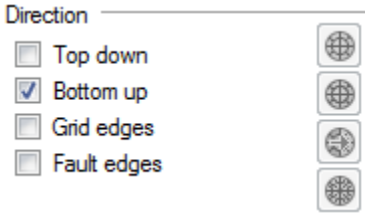
   Change the name of the aquifer, if required.

4. Choose the `Aquifer model`.

5. On the `Connections` tab, insert the polygon defining the aquifer area of interest.

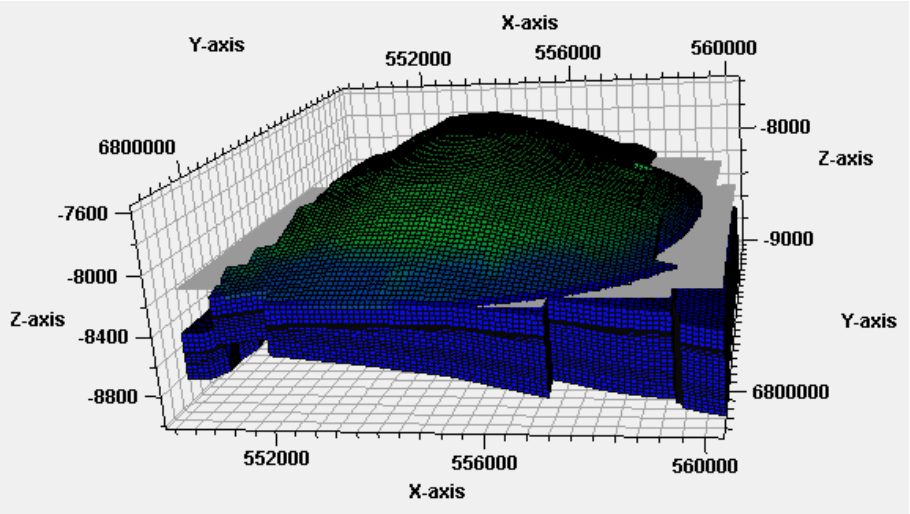


6. In the `Direction` panel, select 'Bottom up'. This option specifies that the aquifer is connected to the bottom edge at the bottom of the reservoir for all cells within the area of interest.



7. Set the `Vertical extent`.
   a. Select **Top limit**.
   b. Choose 'Fixed depth' and enter the depth.

   You may find it useful to display the axis in the `3D window` when setting the depth. To do this, on the `Window 3D` tab, in the `Display elements` group, click **Axis**. Display the oil-water contact as an additional reference.



8. Click **OK** to create the aquifer.

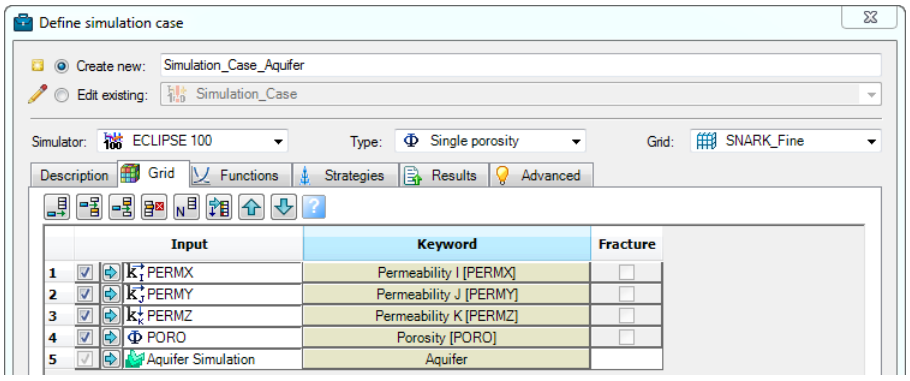
To use the aquifer in a simulation, open the `Define simulation case` process and select your simulation case. On the `Grid` tab, insert the aquifer from the `Aquifers` folder on the `Models` pane into the grid inputs table. Click **Apply** and then run the case.

Define and run the simulation case using the aquifer

In this step of the workflow, define a simulation case and include the aquifer as an input on the `Grid` tab of the `Define simulation case` process.

1. On the `Simulation` tab, in the `Simulation` group, click **Define case**.
2. Click **Create new** and enter a name for the new case.
3. Click the `Grid` tab.
   a. Click **Append** to add a new row to the table.
   b. Select the aquifer in the `Models` pane and insert into the `Input` field.

   The Keyword is automatically set to 'Aquifer'.
4. Click **Apply**.



5. Click **Run** to export the data files and run the case.

Remove unwanted isolated cells

You can find isolated cells in your model and, based on engineering decisions, remove those that do not contribute to your strategies for hydrocarbon production modeling. This helps to reduce the computing resources during reservoir/field simulation.

Isolated cells/regions require processing during simulation but, depending on where they are, they may not contribute to your modeling and production strategies. By removing those that do not contribute, you could reduce the simulation time. For example, there may be small isolated regions that you will never target for production, because they are too small or are in parts of the reservoir that you do not plan to develop.

There are a number of stages in this process:

1. If you haven't already done so, create a property that removes inactive cells and any other cells with porosity, permeability or pore volume below a threshold value. This property has the value zero (0) for cells that you want to remove from the simulation and the value one (1) for all other cells.
2. Use the geometric modeling process to show the connected volumes in this property.
3. View the statistics to determine the number of connected volumes in the new property.
4. Depending on the number of connected volumes:
   - Filter the property to investigate the isolated cells.

     or
   - Create a continuous property and then filter it to investigate the isolated cells.

   The connected volumes property is a discrete property. If you have a large number of connected volumes, more than 10 or 20, you should convert the property to a continuous one as Petrel processes this more efficiently.
5. Review the hydrocarbon volume in the cells you have selected for removal from your model.
6. If you have already defined well trajectories in your model, find selected cell groups that are penetrated by wells and ensure that you retain these if they are potential targets.
7. Remove the isolated cells from your model:
   - Remove all of the cells displayed with your filtering.

     or
   - Remove single clusters.
8. Define a simulation case which uses the isolated cells property.

Note: In the descriptions of these stages, the names used for properties and filters, and the example images used, may be different from the ones that you use and see in your model.

Create a property to exclude cells with zero pore volume

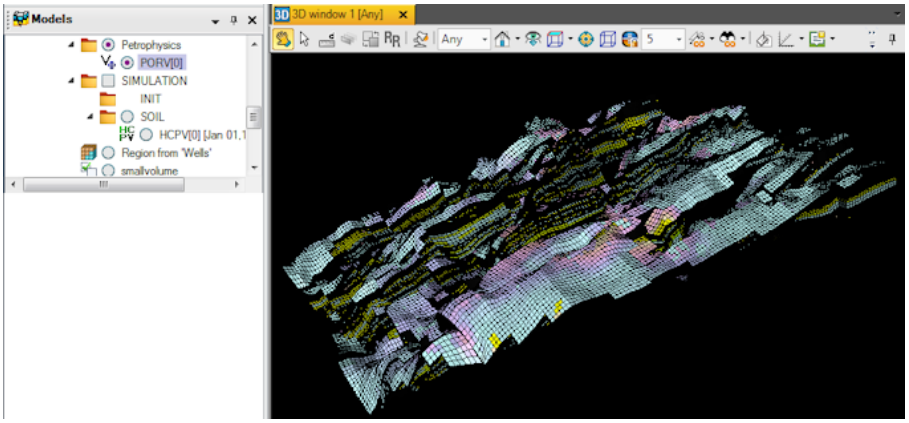
Create a property that excludes cells with zero pore volume which you do not want to simulate.

In this part of the workflow, you create a property using a Boolean calculation. The example sets all cells with a zero pore volume to zero (0) and all other cells to a value of one (1). The cells set to 0 are the ones that you want to ignore. The cells with a value of 1 are those that you will investigate further.

Note: This workflow uses zero pore volume to differentiate between cells, but you could use a different pore volume cut-off, or another property, for example porosity or permeability.

1. Open a 3D window and display a pore volume model (you may have more than one defined).

   To do this, on the `Home` tab, in the `Insert` group, click **Window** and then click **3D window**. Then select a pore volume property from the `Properties` folder in the `Models` pane.



2. Analyze the property using filters, statistics and settings to determine the pore volume cut-off. To do this:

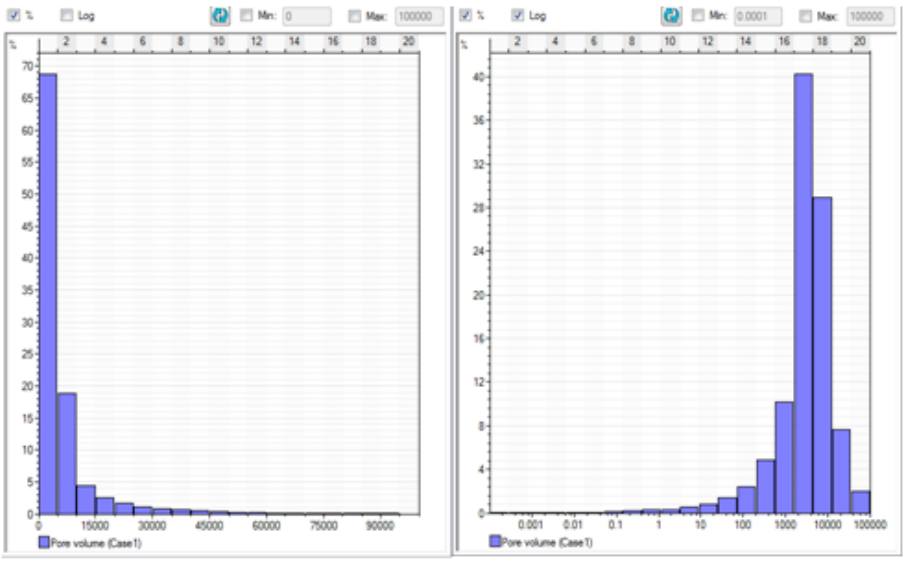
   a. Right-click the property (in the `Properties` folder in the `Models` pane) and select **Create 1D filter**.

   b. Click **Both limits defined**.

      The `Min` value shows you whether your model contains cells with zero pore volume, or a small non-zero volume.

   c. To analyze the property further, display a histogram by right-clicking the property and selecting **Settings**. In the `Settings` dialog, click the **Histogram** tab.

   d. You can toggle the **Log** check box to switch between linear or log scaling of the x axis to view the distribution of the pore volumes, for example:



Using the information from the filter limits and your analysis of the histograms, determine the pore volume cut-off that you want to use.

e. Click on **Cancel** in the filter and histogram windows to clear them without saving your settings.

3. On the `Reservoir Engineering` tab in the `3D and fault properties` group, click **3D properties**.

4. Set the calculator to calculate the property.

   For example, to calculate a property called `actnum` calculated from a property called `PORV[0]` in the `Petrophysics` folder in the `Properties` folder, the calculation is:

   actnum=If(Petrophysics\PORV[0]=0, 0, 1)

   This Boolean logic sets cell values to zero if they have pore volume=0 and to one otherwise. To create this in the calculator:

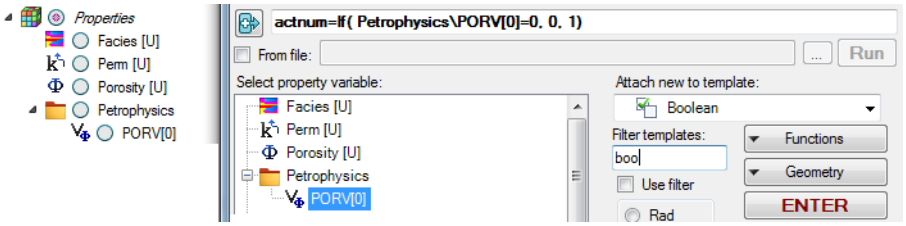
   a. Type **actnum=** into the calculator.

   b. Click **If**.

   c. In the `Select property variable` pane, click the `PoreVolume` model to add it to the calculator.

   d. Complete the calculation by adding **=0, 0, 1)**.

   e. Select the `Boolean` template in the `Attach new to template` list. A quick way to do this is to type **boo** into the `Filter templates` box.

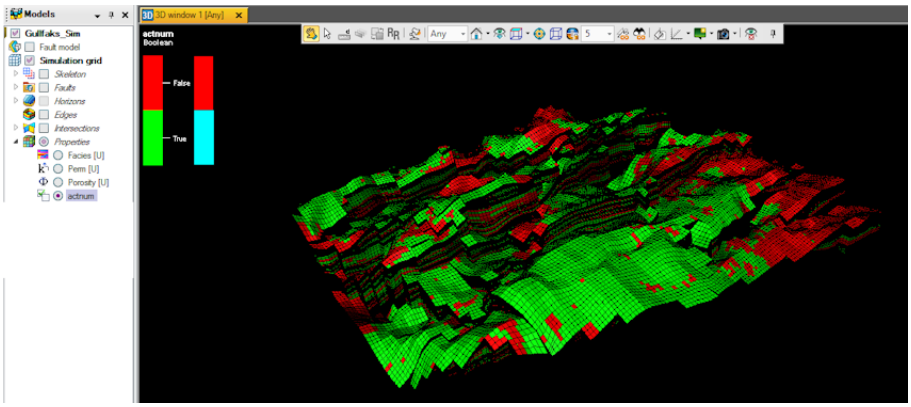


If you choose to use a non-zero pore volume as the cut-off limit, for example, less than or equal to 0.0001, you enter this value into the calculator instead:

actnum=If(Petrophysics\PORV[0]<=0.0001, 0, 1)

5. Click **ENTER** and close the `Property calculator` window.

This adds the actnum property to the `Properties` folder in the `Models` pane and you can display it in the 3D window.



The legend (toggle on/off in the `3D window tools palette` by clicking **Axis** and then **Auto legend**) shows the true (value 1) and false (value 0) colors on the left and a histogram of the relative amount of true/false cells on the right.

Show the connected volumes in the new property

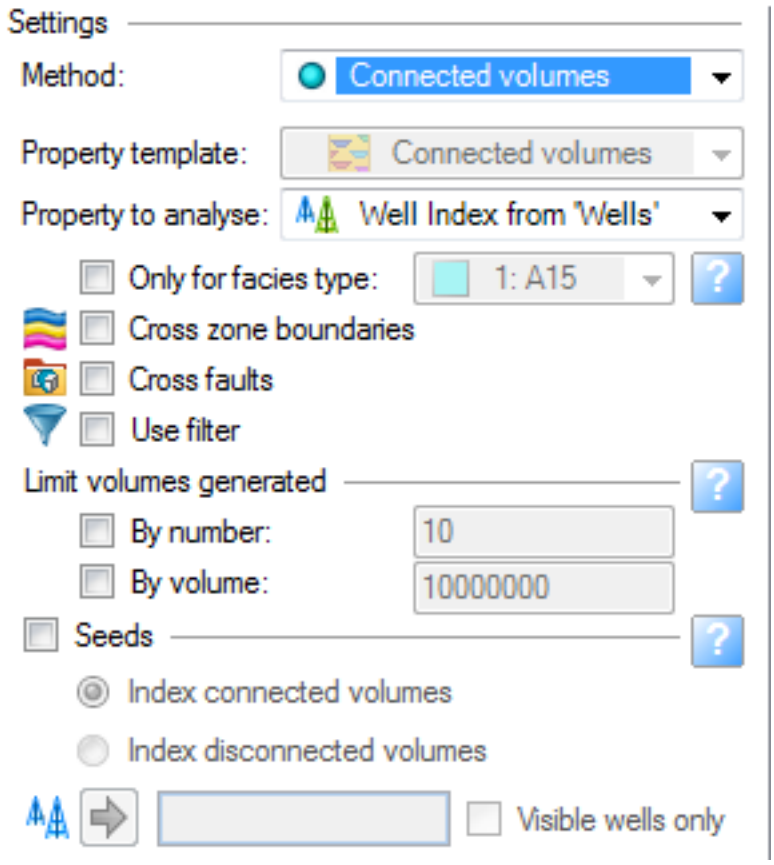
After creating the property to differentiate between zero and non-zero cells (the actnum property from the previous stage), you create a property showing the connected volumes in the non-zero regions.

1. On the `Reservoir Engineering` tab, in the `3D and fault properties group`, click **Geometrical modeling**

   This displays the `Geometrical modeling` dialog.

2. From the **Method** list, select **Connected volumes**.

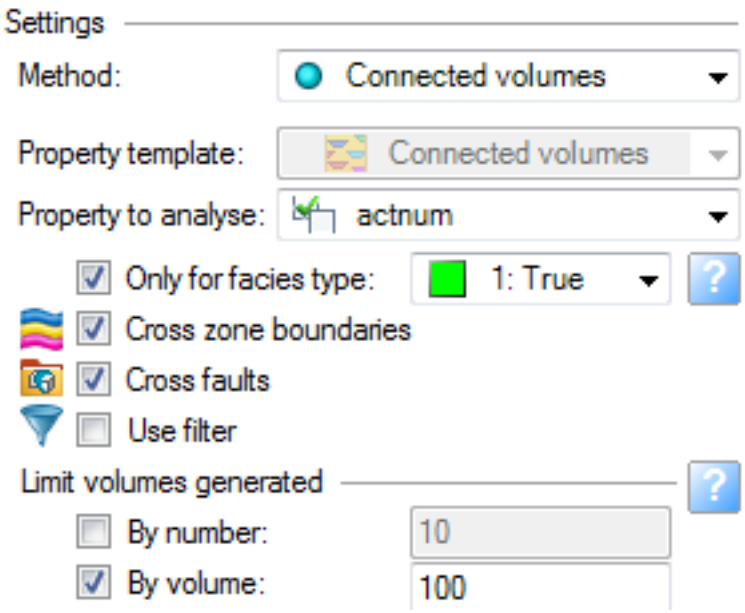
   The `Geometrical modeling` window displays further settings for the connected volumes property.



3. From the **Property to analyse** list, select the `actnum` property.
4. Select the **Only for facies type** check box.

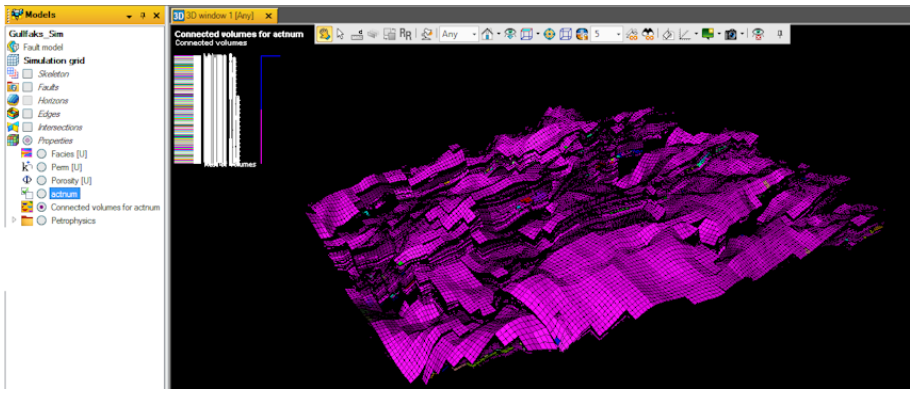
   By default this is set to `1: True`. If this is not displayed, select it from the `Only for facies type` box.
5. Also select **Cross zone boundaries** and **Cross faults**.
6. Limit the number of isolated volumes generated by selecting **By volume** and entering a small volume, for example **100**.

   This volume is 100 of the Petrel project's units, for example $100m^3$ for metric units.



7. Click **OK**.

The new property is in the `Models` pane in the `Properties` folder and you can display it in the 3D window.

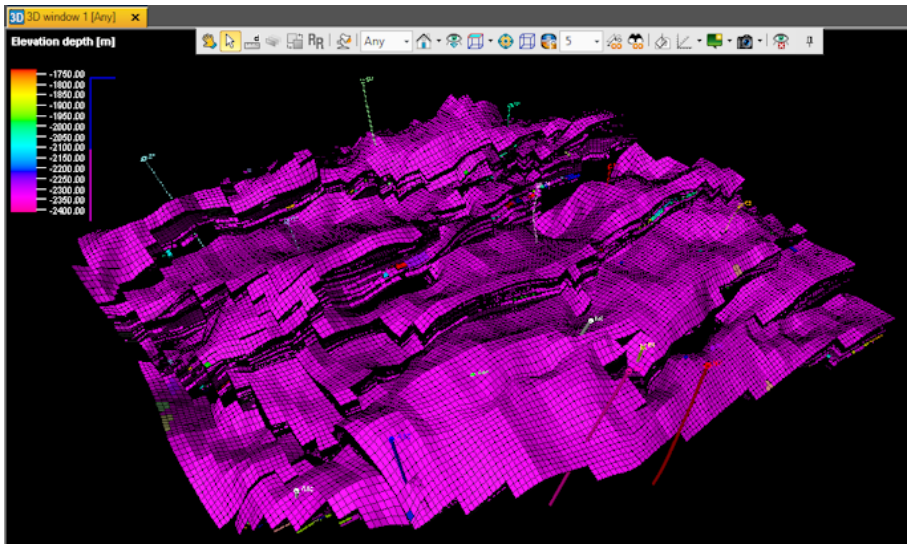


If you want a different name for the property:

1. Right-click the property in the `Models` pane, and then click **Settings**.
2. Click the **Info** tab and enter the new name in the `Name` box.
3. Click **OK**.

You can use the same method to change the names of other properties. For example you may have defined a number of different connected volume properties, using different methods and settings, and want to use a naming convention that allows you to choose between them.

If you have wells defined in your project, you can display them by clicking the **Wells** folder in the `Input` pane. The wells are displayed along with the current property in the 3D window.



Display the statistics for the connected volume property

Use the statistics to find the number of connected volumes in the property.

In this step, you use the statistics to find the number of connected volumes in the connected volume property. Depending on the number:

- If you have a small number of connected volumes, you can start filtering the property to investigate the isolated volumes to decide which to keep or remove.
- If you have a large number of connected volumes, first create a continuous property from the connected volume property as Petrel processes continuous properties more efficiently. You can then filter the continuous property to investigate the connected volumes.

1. Right-click the connected volume property in the `Properties` folder in the `Models` pane and select **Settings**.
2. Click the **Statistics** tab.
3. Note the number of connected volumes and click **OK** to close the statistics window.

Having viewed the statistics, you should filter the property to investigate the isolated cells, or create a continuous property and then filter it.

Determine which cells to remove from the model

Filter the connected volume property to view the isolated cells and determine which to remove.

In this step, you use a 1D filter to view different collections of isolated cells to determine which cells to remove from the model. This is an iterative process in which you apply different filters to display different connected volumes. If your filtering displays all of the volumes that you want to remove using the filter, you can remove them in one step. If not, you will have to investigate and remove individual volumes in turn.

To filter the cells:

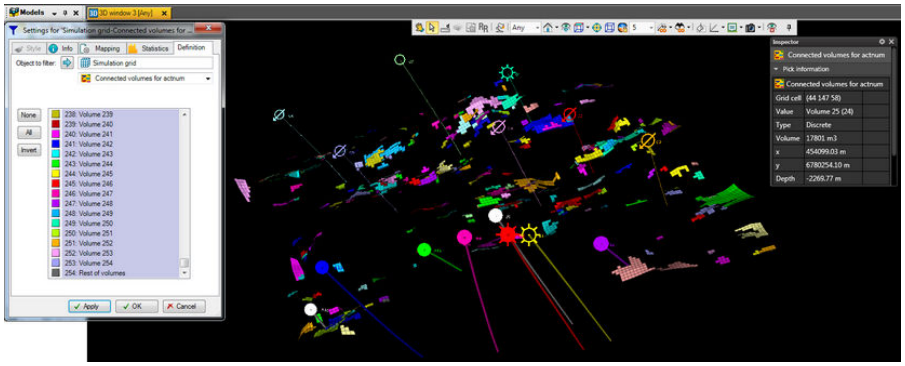
1. Display the connected volume property in a 3D window.

If you don't have the property displayed from the previous process, in the quick access toolbar click **Window** and select **3D window**. Then click the connected volume model in the `Models` pane.

2. Right-click the connected volume property in the `Properties` folder in the `Models` pane and select **Create 1D filter**.
3. Use the filter settings to create a set of volumes to view and click **Apply**.

   `0: Volume 1` is the largest volume and `254: Rest of volumes` is the smallest and may contain more than one isolated volume.

4. To inspect the properties of any of the filtered cells, on the `Home` tab in the `View` group, click **Inspector**. With the **Pick tool** active (click **Select[P]** in the `3D window tools palette`), click on any cells in the 3D window to inspect the properties, including volume.
5. Repeat the filtering and inspecting to display the cell clusters that you want to remove from the model. You can toggle the well display on and off, by selecting **Wells** from the `Input` pane.



   This example has all volumes displayed apart from `0: Volume 1`, but you could have any set of volumes here depending on your filtering and the settings in your model.

6. Close the filter and inspector windows.

At the end of the process, you have a number of filtered connected volumes displayed.

Having filtered the cells, you can perform further checks before inactivating the small connected volumes:

- Check the volume of hydrocarbon in the filtered cells against the total volume in the model/grid. Depending on the volume contained, you may wish to re-filter to reduce the potential hydrocarbon volume that you are going to remove from your model.
- If you have wells in your model, you can should the isolated cells to see if any of them are penetrated by wells.

After these checks you can remove unwanted clusters of cells from the model.

Determine which cells to remove from the model using a continuous property

Where you have a larger number of connected volumes, create a continuous property and then filter it to view the isolated cells and determine which to remove.

In this step, you create a continuous property from your connected volumes property. You then use a 1D filter to view different collections of isolated cells so that you can determine which ones to remove from the model. This is an iterative process in which you apply different filters to display different connected volumes. If your filtering displays all of the volumes that you want to remove using the filter, you can remove them in one step. If not, you will have to investigate and remove individual volumes.

To create the continuous property and filter the cells in it:

1. Display the connected volume property in a 3D window.

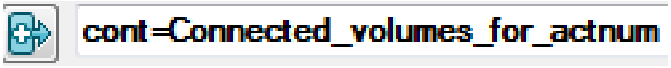
   If you don't have the property displayed from the previous process, in the quick access toolbar click **Window** and select **3D window**. Then click the connected volume model in the `Models` pane.

2. On the `Reservoir Engineering` tab in the `3D and fault properties` group, click **3D properties**.

3. Use the calculator to create the property. For example, to create a property called `cont`.

   a. Type **cont =**

   b. Insert the connected volume property into the calculation

   c. Select **General** from the `Attach new to template` menu.

   d. Click **ENTER**

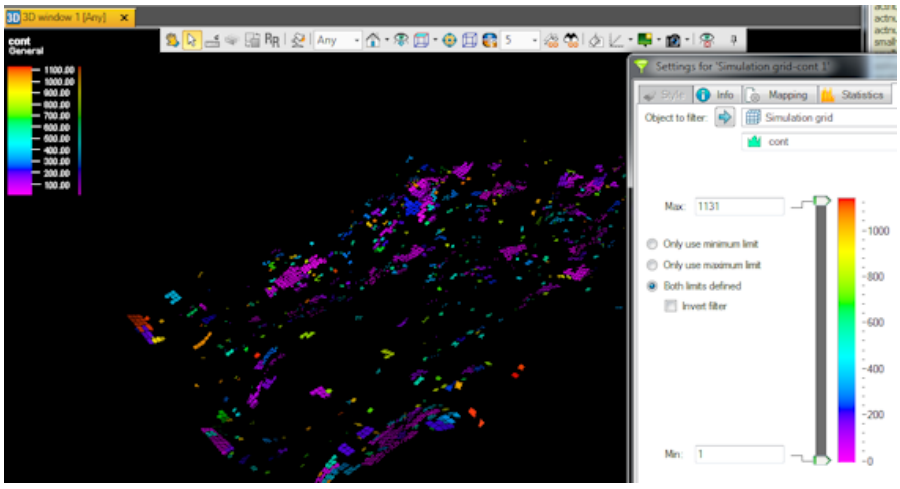
4. Select the continuous property from the `Properties` folder in the `Models` pane to display it in the `3D window`.

5. Right click on the property in the `Properties` folder in the `Models` pane and select **Create 1D filter**.

6. Set `Min` to **1** and click **Apply**.

   This removes the largest volume, Volume 0, in the filtered display. The lower numbers are for the bigger connected volumes, with the `Max` limit being for the smallest connected volume.

7. On the `Grid Property Tools` tab, in the `Color table` group, click **Adjust color table**. Click **Yes** in the next window.



8. If the initial filter setting (removing volume 0) does not remove enough connected volumes, use the filter settings to create a different set of volumes to view and click **Apply**.

   This is an iterative step and you can filter and re-filter until you have the set of connected volumes that you want to deactivate.

9. To inspect the properties of any of the filtered cells, on the `Home` tab in the `View` group, click **Inspector**. With the **Pick tool** active (click **Select[P]** in the `3D window tools palette`), click any cells in the 3D window to inspect their properties, including volume.

10. Close the filter and inspector windows.

At the end of the process, you have a number of filtered connected volumes displayed.

Having filtered the cells, you can perform further checks before inactivating the small connected volumes:

- Check the volume of hydrocarbon in the filtered cells against the total volume in the model/grid. Depending on the volume contained, you may wish to re-filter to reduce the potential hydrocarbon volume that you are going to remove from your model.
- If you already have wells in your model, you should review the isolated cells to see if any of them are penetrated by wells.

After these checks you can remove unwanted clusters of cells from the model.

Check the hydrocarbon pore volume in the filtered cells

Check the amount of hydrocarbon in the connected volumes that you have selected for deletion.

In this step, you check the volume of hydrocarbon in the filtered cells against that in the complete model. Depending on the relative percentages, you can deactivate the isolated cells you have identified, or you can repeat your filtering to create a different group of cells for deletion. For example, if you have 0.5% or 1% hydrocarbon in your selected cells, you may decide to deactivate them. If you have 3%, 5% or 10%, you may decide to re-filter to create a set of isolated cells that contains less hydrocarbon.

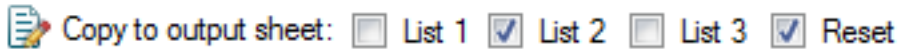
To check the hydrocarbon volume:

1. Double-click the label of the hydrocarbon pore volume (possibly shortened to HCPV) in the `Models` pane.

For example 

This displays the `Settings` dialog.

2. Click the **Statistics** tab.
3. Create a report for the hydrocarbon volume in the filtered cells from the `Settings` window:
   a. At the top of the window, ensure that **Show statistics for the filtered cells only** is on. Note the orange background color to the filter icon.
   b. At the bottom of the window, ensure that the `List 2` and `Reset` check boxes are selected.

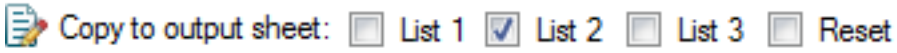


   c. Click **Copy to output sheet** .

   This displays the `Output sheet` for the HCPV property for the filtered cells.

Statistics for HCPV[0] [Jan 0	
Description	**Value**
Unit:	rm3
Time step:	Jan 01,1995
Is upscaled (U	No
Realization in	0
Total number	3367
Grid cells (nl x	117 x 147 x 5
Total number	997542
Property resol	Per cell with v
Compressed	72.11% of cell
----------	----------
Type of data:	Continuous
Min:	0
Max:	5648
Delta:	5648
Number of def	3367
Mean:	306
Std. dev.	713
Variance:	508381
Sum:	1028626

4. Create a second report for the hydrocarbon volume in the unfiltered cells from the `Settings` window:
   a. At the top of the window, ensure that **Show statistics for the filtered cells** is off.
   b. At the bottom of the window, ensure that only the `List 2` check box is selected.



c. Click **Copy to output sheet**.

This adds the property for the unfiltered cells to the `Output sheet` for the HCPV property, for example:

Statistics for HCPV[0] [Jan 0	
Description	**Value**
Unit:	rm3
Time step:	Jan 01,1995
Is upscaled (U	No
Realization in	0
Total number	368252
Grid cells (nl x	117 x 147 x 5
Total number	997542
Property resol	Per cell with v
Compressed	72.11% of cell
----------	----------
Type of data:	Continuous
Min:	0
Max:	13200
Delta:	13200
Number of def	368252
Mean:	601
Std. dev.	1035
Variance:	1072125
Sum:	221234396

5. Compare the sum for the filtered (1028626 in the example) and unfiltered (221234396) volumes and use the percentage difference to decide how to proceed.

   In the example, the filtered cells that have been identified for deactivation account for less that 0.5% of the hydrocarbon volume and this is well below any threshold for reconsidering the filtering.

6. Click **OK** to close the `Settings` window and close the `Output sheet`.

At the end of the process, you can re-filter the cells if the hydrocarbon volume in the current set is too high. You can also check to see how any wells in your model penetrate the filtered cell set.

Check the well trajectories through the filtered cells

Check the intersections of the wells with the filtered cells to determine whether to keep some of the filtered volumes in the model.

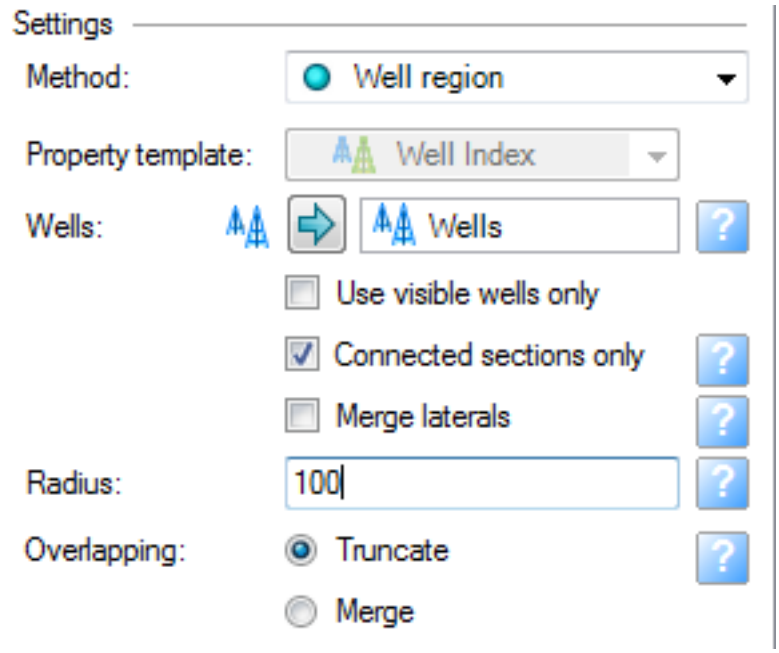
In this step, you see which cells are penetrated by wells, if you have them in your model. If you identify volumes which are targets, or potential targets, you can re-filter your connected volumes property to remove them from the filtered set and so from the cells that you are going to deactivate.

1. On the `Reservoir Engineering` tab, in the `3D and fault properties` group, click **Geometrical modeling**.

This displays the

2. Ensure that `Create New` is selected (it is by default) and then from the `Method` menu, select **Well region**.
3. Select **Connected sections only**.
4. Set the radius from each well that you want to consider.

   The units are based on your project settings. For example, metric units with a setting of 100 define a radius of 100m.



5. Click **OK**.

   This creates a new property with the default name `Region from 'Wells'`.

6. Select the property to display it in the `3D window` and then move around the display to look at any areas of interest.

   To toggle the wells on/off in the display, select **Wells** from the `Input` pane. For example, here is a small volume that can be ignored, even though it lies close to a well:

At the end of the process, you can either re-filter the cells if significant volumes are intersected by, or lie close to wells in your model and are targets for production. If you don't need to re-filter the cells, you can now deactivate them.

Deactivate the filtered cells

Deactivate the cells that you have identified as being unnecessary for your model.

In this step, you deactivate those isolated cells that you want to remove from the model. In this process, the filtered cells contain all of the isolated cells that you want to deactivate.

To remove all of the filtered cells:

1. Select the connected volume property to re-display it in the 3D window.

   As your filter is still set on, this shows the filtered cells in the connected volume property. If the filter is not on, you can re-select it from the `Filters folder` in the `Input` pane.

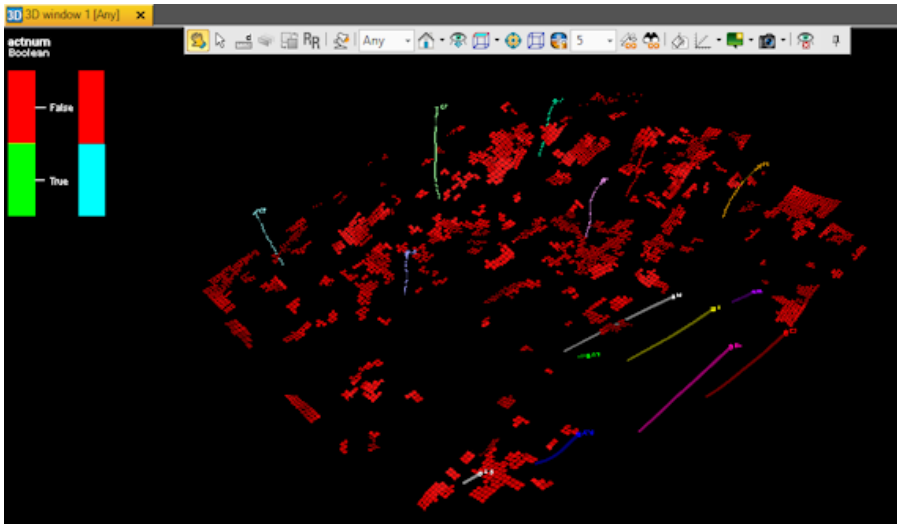
2. On the `Reservoir Engineering` tab, in the `3D and fault properties` group, select **3D properties**.

3. Type **actnum=0** for the calculated expression (actnum=0).

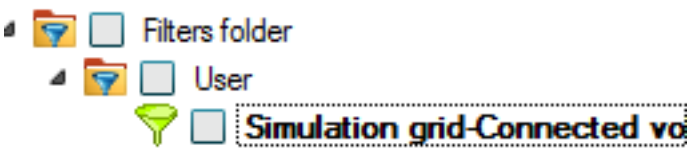
4. Select **Use filter** (Use filter).

5. Click **ENTER** and close the `Property calculator`.

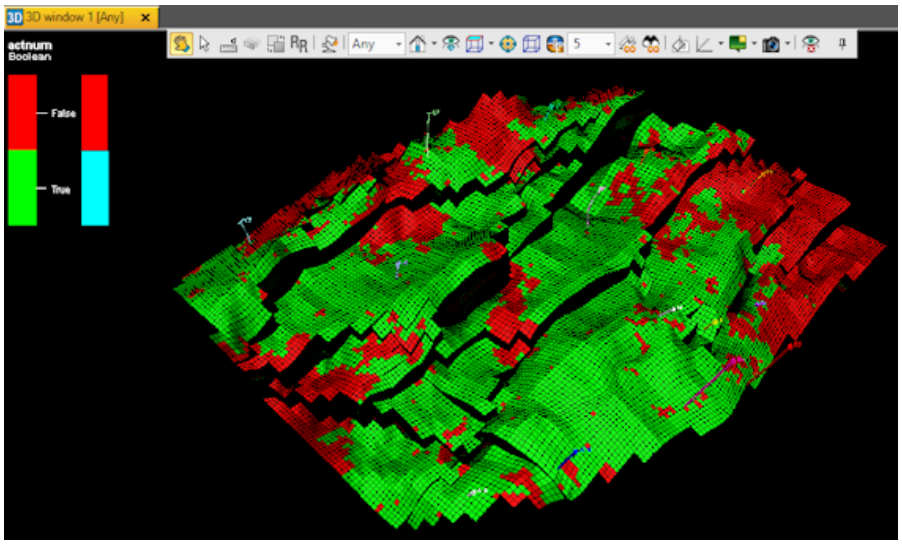
6. Select the **actnum** property to display it in the `3D window`.



7. In the `Input` pane, click the filter in the `Filter folder` to switch it off.



At the end of the process, you have deactivated groups of isolated cells and the simulator will not consider them in its calculations. The actnum property includes all of the cells that you deactivated.



Deactivate groups of cells individually

Analyze and deactivate individual groups of isolated cells.

In this step, you deactivate individual groups of cells to remove those selections from your model. To do this you analyze the filtered display of connected volumes. To isolate a group of cells you use the calculator to set them to actnum=0, while leaving the other cells with an actnum property value of 1.

1. Display the filtered connected volume property in the `3D window`.

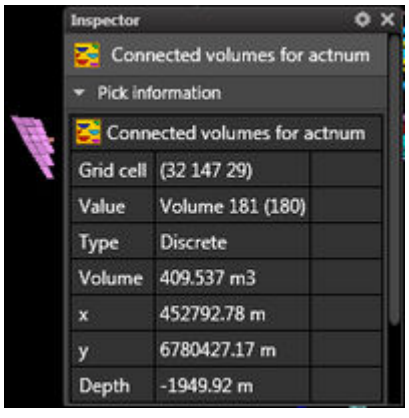
   To do this, select the connected volume property from the `Models` pane and the filter from the `Filters Folder` in the `Input` pane.

2. Switch to pick mode ( ).

   Toggle between pick and view mode by pressing Esc, or select the mode from the window Tool Palette. In this way, you can move around the grid in view mode and inspect properties in pick mode.

3. On the `Home` tab, in the `View` group, click **Inspector**.

4. Click individual groups of cells in pick mode to identify small isolated volumes. For example:



Note: Because of limits in the color ranges used for the display, it may be that cells of different volumes are displayed in the same color. You can check this by creating a 1D filter (right click on the cells and select **Create 1D filter**) and filtering on the volume number.

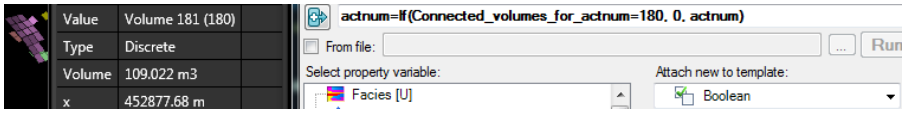
5. On the `Reservoir Engineering` tab in the `3D and fault properties group`, click **3D properties**.
6. Enter the calculation to give the connected volume an actnum value of 0.

   For example, if the Boolean property is called actnum and you want to give connected volume 180 an actnum value of 0, the calculator expression is:

   actnum=If(Connected_volumes_for_actnum=180, 0, actnum)

   To enter this into the calculator:

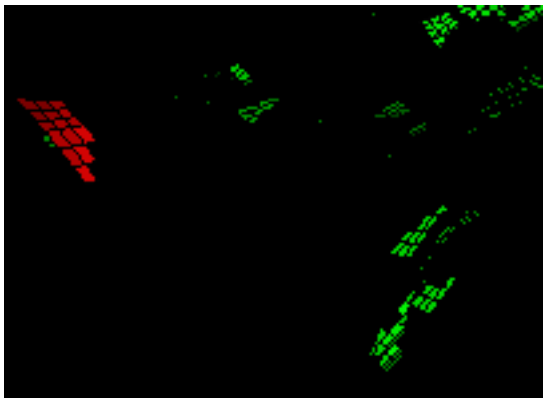
   a. Type **actnum=**
   b. Click **If**
   c. Select the connected volume property from the `Properties` folder in the `Models` pane and insert it into the calculator
   d. Type = and the volume number before the first comma (`=180`) and **0** before the second
   e. Click before the closing bracket, select the actnum property from the `Properties` folder in the `Models` pane and insert it into the calculator.
   f. Select the `Boolean` template in the `Attach new to template` box.



Note: If you created a 1D filter in step 4, you can use it to filter several connected volumes at the same time. In this case, you can set the actnum values on these cells in one go. To do this, enter the calculator expression **actnum=0** and select **Use filter**.

7. Click **ENTER**.
8. Select the connected volume property from the `Properties` folder in the `Models` pane to display it in the `3D window`.

   The group of cells indexed by volume number 180 is now set to 0.

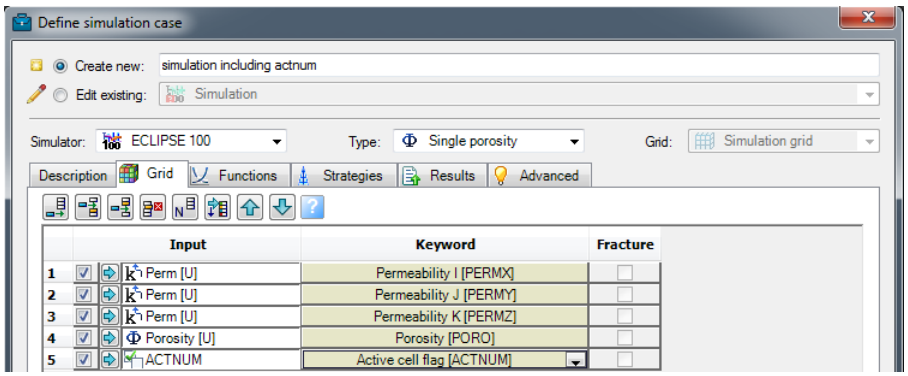


At the end of the process, you have deactivated a particular group or groups of cells by adding them to the Boolean property actnum (created in the first step of the process).

Define a simulation case which includes the isolated cells property

In this step, you define a simulation case and include the actnum property as an input on the `Grid` tab of the `Define simulation case` process.

1. On the `Simulation` tab, in the `Simulation` group, click **Define case**.
2. Click **Create new** and enter a name for your new case.
3. Click the `Grid` tab.
4. Click **Append** to add a new row to the table.
5. Select the `actnum` property in the `Models` pane and insert into the `Input` field.
6. Choose **Active cell flag [ACTNUM]** from the `Keyword` list.
7. Click **Apply**.



8. Click **Run** to export the data files and run the case.

You can compare the simulation results with the isolated cells defined against other simulations which do not include the property.