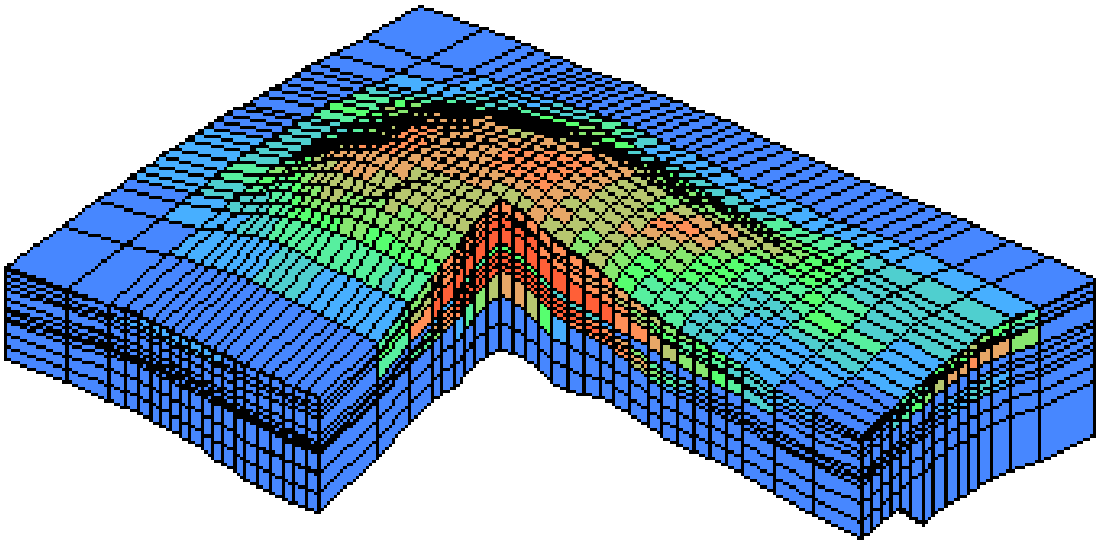
SENSOR Coats Engineering, Inc.
Compositional and Black-Oil Reservoir Simulation

Picture courtesy of ConocoPhillips, from SimView



System for **E**fficient **N**umerical **S**imulation of **O**il **R**ecovery

SENSOR MANUAL

April 1, 2011

Sensor is a numerical reservoir simulation model.

Questions may be addressed to:

Support@CoatsEngineering.com

Website: http://www.CoatsEngineering.com

PREFACE

The following suggestions may save you some time and effort if you have not used Sensor. First read the short sections "Model Description" and "Overview of the Datafile". The provided test cases are described in Table 3. See “Installation” and “How to run the Sensor.exe file” in the “Executing Sensor” section to run an existing case, and see “Data errors and problems” in this same section for some guidance on effective usage. More guidance is given is given in Section 7, “Running Hints” and also throughout the manual on specific features and options. Section 6, “Aid to Viewing Program Printout”, gives a number of search strings that can be used to easily find specific information in the output file. See Section 1.3 and Appendix 1 for discussion of formulation and linear solver selections. You may find the Keyword Index in Section 11 helpful.

TABLE OF CONTENTS

Model Description

Sensor is a generalized 3D model for simulating black oil and compositional problems in single porosity, dual-porosity, and dual permeability reservoirs. The Sensor executable is compiled for use on desktop Windows PC's.

Formulations and Solvers

Sensor includes Impes and Implicit formulations. Its three linear solvers are reduced bandwidth direct (D4), Orthomin preconditioned by Nested Factorization, and Orthomin preconditioned by ILU with red-black and residual constraint options.

Gridding

Any grid type or combination of grid types may be used with Sensor. First is the conventional, seven point orthogonal Cartesian xyz grid. Second is the r-θ-z cylindrical coordinate system. Third is any grid -e.g. corner-point, refined, unstructured, or hybrid - for which input values of pore volume, transmissibilities, and depth are available from a grid package. Since the matrix is represented in unstructured form in Sensor's linear solvers, any unstructured nature of the grid does not adversely affect performance, unlike some other models with structured matrix representations. The nine-point option can be used in xy planes with the Cartesian grid to reduce grid orientation effects. The model handles faults with non-neighbor connections and provides angular closure in the case of cylindrical coordinates when the multiple angular increments sum to within .01 of 360 degrees (by internally creating the appropriate "non-neighbor" connections).

Black Oil pvt

The black oil pvt includes oil in the gas phase (the r_s stb/scf term), and therefore applies to gas condensate and volatile black oil problems. A foamy oil option of black oil treats the first portion of released or injected gas saturation as entrained or emulsified gas which flows with the oleic phase. The black oil pvt table can be re-entered in Recurrent Data at various times to reflect changes in surface processing at the time of the re-entry. Black oil tables can be internally generated from input compositional descriptions.

Compositional pvt

Sensor uses the Peng-Robinson and Soave-Redlich-Kwong equations of state (eos), with optional shift factors and any number of components. Eos parameters may differ for reservoir and surface separation conditions. A number of options are provided for automatic simplification of compositional eos descriptions that can significantly improve run performance

and user productivity. These options make it easy for the user to determine the required degree of complexity in the fluid characterization.

In cases where only depletion is present, with or without water injection, K-values internally generated from the eos can be used to reduce run cpu times. In other cases, when using the IMPES formulation, the tracer option can be used in combination with the K-value option in order to revert to the more rigorous eos when injected gas tracer fractions are detected. Surface separation may be performed using a multi-stage flash or using separator liquid recovery factor tables to simulate both the separator train and a liquids plant.

Compositional descriptions can be automatically converted to black oil tables for improved efficiency in cases in which compositional effects prove not to be significant. The saturation curve can be extended beyond the saturation pressure of the original fluid for improved agreement between compositional and black oil runs.

First contact miscibility options are available for solvent injection into compositional oil reservoirs. An option is provided to automatically pseudoize the oil, while providing for bypassed oil and dispersion control, for maximum accuracy and efficiency.

Multiple Reservoirs and Multiple pvt

Sensor handles the case of multiple reservoirs, e.g. stacked reservoirs, where no transmissibility connects any pair of reservoirs and no well is completed in more than one reservoir. This capability can reduce cpu time by a factor of two or more because different reservoirs do not require the same numbers of Newton or linear solver iterations or pvt treatment.

Any number of pvt tables, black oil and/or compositional, may be entered. Each grid block is assigned to one of these tables. All grid blocks in a given Reservoir must be assigned to compositional pvt tables or to black oil pvt tables. That is, multiple pvt tables can be used within a Reservoir but each Reservoir must be uniformly black oil or compositional. There can be any number of Reservoirs in a simulation. All compositional pvt types must use the same number and names of components.

Relative Permeability and Capillary Pressure Treatment

Two-phase water-oil and gas-oil relative permeability and capillary pressure tables may be entered for multiple rock types. These tables are then normalized for use in the model. Relative permeability endpoints are optionally entered by gridblock for use in denormalization. Three-phase oil relative permeability is calculated using Stone's first method as extended by Fayers to treat minimum oil saturation as a function of S_g. Optionally, Stone's second method or Baker's Linear Interpolation may be used. Trapped gas saturation with associated k_{rg} hysteresis is included. Analytical forms of the relative permeabilities and capillary pressures are also available.

In default mode, Sensor uses P_{cwo} and P_{cgo} from the normalized saturation tables, with appropriate denormalization based on gridblock residual saturations. In addition, capillary pressures can be scaled using the Leverett J-function. A vertical equilibrium option can also be applied to the capillary pressure curves.

Compaction

Compaction with hysteresis is represented using input tables giving rock compressibility as a function of stress and porosity. Optionally, the effects of water weakening are accounted for by including water saturation parameters in the compaction tables.

Initialization (Equilibration)

Initial reservoir pressure, saturation, and composition distributions are calculated by capillary-gravitational equilibrium. Any number of grid initialization regions may be specified with different pressure, fluid contacts, and composition vs depth.

Dual Porosity Systems for Fractured Reservoirs

Sensor can model dual porosity and dual permeability systems, and also robustly handles mixed unfractured/dual porosity/dual permeability systems. The number of reservoir layers is doubled, with the first half representing the matrix and the second half representing the fractures. In dual porosity systems (with no dual permeability regions), the matrix equations are eliminated prior to linear solution for maximum efficiency. Sensor easily handles dual systems in unstructured grids, through specification of approximately equivalent rectilinear gridblock dimensions, and through specification of rectilinear matrix block dimensions.

Coal Bed Methane

Sensor can model coal degasification due to depletion in black oil mode. A trace oil phase represents the coal, and the defined porosity represents the cleats and fractures. Black oil pvt data represents the gas properties and adsorption isotherm. Multiple pvt tables and regions allow simulation of depletion in reservoirs with mixed coal and sand layers. Sensor does not have an enhanced coal bed methane model for processes such as carbon dioxide or nitrogen injection.

Implicit Well Treatment

The implicit well treatment includes wellbore crossflow, turbulent (non-Darcy) gas flow effects, and tubing head pressure tables with gaslift for multi-phase flow in the tubing. Special logic is used in compositional cases to achieve specified target rates. This increases efficiency by avoiding additional Newton iterations to converge on specified rate. Options are also available for SWAG (simultaneous water and gas) injection, WAG (alternating water and gas) injection cycle control, regional pressure control, drawdown control, planar sources, and management of new well drilling through drilling schedule logic.

Platforms (Gathering Centers)

Platform or gathering center logic allows assignment of target rates and constraints to groups of wells. Gas can be reinjected, taking into account available produced gas, gas sales, and fuel loss. Produced gas from one platform can be transferred for injection on another platform. Allocation of production targets to the wells can be optimized to maximize instantaneous oil recovery based on simple well penalty factors. On-times can be entered for production, water injection, and gas injection.

Tracers

Sensor can calculate tracer fractions for any number of traced components in Impes mode. This feature is useful in equity situations as well as in tracking injected water and gas streams. Traced components can be any of the fluid components, including water. Tracer calculations increase run cpu times very little.

A similar feature is the Salinity option, allowing prediction of produced water salinity variation resulting from differing salinities of original and injected water.

Regions

Initialization regions may be specified for equilibration purposes, and pvt regions may be specified for variation in fluid characterization. Sensor also provides for specification of output regions for analysis of results and/or for pressure control. Sensor provides a variety of output for analysis, including recurrent printout, end-of-run summaries, and plot file writes, showing rates and cumulatives of production and injection for different (output) regions of the grid. Superregions may be used to group given sets of regions for output or for pressure control.

Stable Step Logic

In default mode, Sensor determines time steps automatically using change-type criteria. In the Impes case, a stable step option determines time steps using stability theory. This option ensures smooth results (e.g. gor and watercut) and eliminates the occasional burden of experimenting with change criteria to reduce oscillatory or unstable results.

Dynamic Dimensioning

Sensor requires no user input data related to dimensioning, either for restart runs or for runs from zero time. The executable scans the data file to determine all dimensions required. In the restart case, it detects dimension changes required and, if necessary, redimensions itself differently from the run which created the restart record.

Active-Block Storage and CPU

The entire model, including the linear solvers, is coded using mapping to require storage and arithmetic only for active blocks. There is no overhead in storage or cpu for blocks missing due to reservoir geometry.

Some of the methodology in Sensor is described elsewhere[1-4].

Overview of the Datafile

All data are entered in free format, using keywords. All keywords are upper case. They do not have to start in column 1, which allows indentation for datafile clarity. The reader reads columns 1-132 on each line. All data are separated by one or more blanks. Integer data should be entered as integers but that is not mandatory. Trailing decimal points are not mandatory in non-integer data. For example, the number 6000. may have its decimal point omitted. Most of the data (keywords) are order independent; exceptions are noted.

Apart from the pvt data, there are almost no differences in the data input for black oil and compositional problems (Equilibration (INITIAL) data formats and specifications for injected gas composition differ). All entered pressures are psia unless noted otherwise. Data strings can be shortened by using n*v. For example, the data string ... 10 10 10 10 7 7 7 3 3 3 3 3 ... may be entered ... 4*10 3*7 5*3 ... Comments may be included on any keyword or data line following an !. Comment lines may appear anywhere in the datafile. A comment line is one where the first character on the line is a "C" or "c", followed by one or more blanks. Blank lines are also permissible.

It is sometimes convenient to use the entries SKIP and SKIPEND to skip large portions of the datafile without the burden of commenting out many lines. The reader skips all lines from SKIP to SKIPEND, inclusive. A given pair of SKIP and SKIPEND must be in the same stack.

The datafile consists of three sections:

Initial Data
Modification Data
Recurrent Data

The datafile should start with TITLE/alphanumeric lines/ENDTITLE, followed by GRID N_x N_y N_z, and remaining Initial Data. The Initial Data end with the keyword ENDINIT. The Modification Data, if present, start with the keyword MODIFY, following ENDINIT. Recurrent Data normally start with the keyword WELL , and end with the keyword END, which is the last keyword in the datafile. The program ignores any lines of data or comments following END. Most keywords are optional. The few keywords which normally would appear in any dataset are:

```
GRID  MISC  PVTBO or PVTEOS  SWT,SGT or KRANALYTICAL DELX
DELY  THICKNESS  DEPTH  KX,KY,KZ and/or TX,TY,TZ  POROS and/or PV
INITIAL ENDINIT  ! end of Initial Data

WELL  WELLTYPE  BHP or THP  RATE  TIME or DATE END
```

Table 1 shows the layout of a datafile. Table 2 is an actual datafile with voluminous and repetitive portions omitted. Table 3 lists and briefly describes a number of test problems.

Large blocks, or any portions of data desired, may be put in files and called by the main datafile using the INCLUDE keyword.

Example:

```
C           ** Initial Data **

TITLE
   ... title lines ...
ENDTITLE
GRID 63 47 6
INCLUDE
  krdata.inc   ! any desired filename
INCLUDE
  griddata.inc
PVTEOS
... data ...
... other keywords and data ...
ENDINIT    ! end of Initial Data

C          ** Recurrent Data **

WELL
... well data ...
... other Recurrent Data ...
TIME 365
ENDRUN    ! end of execution
... continuing Recurrent Data keywords and data ..
TIME 6350
END       ! datafile errors checked to here
```

When include files containing voluminous gridblock array data are used, i.e. when large arrays are specified with the VALUE or LAYER options, reading the data can be much faster when the array header is contained in the include file, rather than being specified prior to an include file containing its values. For example, it is better to specify:

```
INCLUDE POROS.INC !  where  POROS.INC  contains  the  array  header  and
                  !  following values
```

than to specify:

```
POROS VALUE
INCLUDE POROSVALUES.INC ! where POROSVALUES.INC contains only the array
                        ! values
```

The program checks for data errors down to the entry END. The program executes the datafile down to the entry ENDRUN or END, whichever comes first. The entry ENDRUN is normally not used. If END or ENDRUN is the first Recurrent Data entry, the program will process and check all data down to the END line, initialize the reservoir, and stop.

If no well keywords are entered in Recurrent Data, the program will run with no wells. Except where noted otherwise, all data are remembered - you do not need to reenter data except to change previously entered values.

Executing Sensor

Installation

With respect to installations, the 3 Sensor products are Sensor64, Sensor, and Sensor6k. They are installed by execution of a provided Windows installer (.msi) file. The following environment variables can be used for workflow integrations and manual execution and are set for all product installations:

Environment Variable	Description (points to)
sensor	Sensor product executable
sensorhome	Sensor Product installation folder
sensordata	Folder containing example data sets
sensorplot	SensorPlot.exe
plot2excel	Plot2Excel.xls
sensormap	SensorMap.exe
map2excel	Map2Excel.xls

Multiple products can be installed simultaneously. The above environment variables will point to the files and folders associated with the last installed product. If multiple products are installed and any one is uninstalled, one of the remaining products must be repaired (open Control Panel/Add or Remove Programs and click on Sensor6k, then the Support button to repair) or reinstalled to redefine the above environment variables.

In 64-bit Sensor installations (Sensor64), “sensor” and “sensor64” both point to the installed 64-bit Sensor executable, and “sensor32” points to the additionally installed 32-bit Sensor executable.

In 32-bit Sensor installations (Sensor), “sensor” and “sensor32” both point to the installed 32-bit Sensor executable.

In Sensor6k installations, “sensor” points to the installed 32-bit Sensor6k executable.

The installation-specific environment variables (and executable names, as discussed below) can be used to specify executables when multiple products are installed.

The installer also modifies the system environment variable %path% to include the installation folders containing executables, so the installed executables can be directly referred to by name for command-line execution and integrations. All installations include their corresponding executable named as sensor.exe, which can be globally referred to as sensor.exe or as sensor. When multiple products are installed, the name sensor.exe or sensor will refer to the last

installed product executable. Sensor6k installations include a sensor.exe copy named sensor6k.exe and a sensor.exe copy identified by its full name sensor6k_month_day_year.exe, Sensor (32 bit) installations include a sensor.exe copy identified by its full name sensor_month_day_year.exe, and Sensor64 installations include a sensor.exe copy identified by its full name Sensor64_month_day_year.exe, where month, day, and year additionally indicate the program version. Sensor64 installations also include the 32 bit executable sensor_month_day_year.exe.

How to run Sensor from Command Line

To run Sensor from a script or from another application, or to run Sensor manually from a DOS Command Prompt window (to open click on Start/All Programs/Accessories/Command Prompt), go (cd) to a work directory for execution. Execute the command

```
sensor  datafile  outputfile
```

where "datafile" is the name of the input data file and “outputfile” is any name desired for the printed output file. If no path is given for the input datafile, then the datafile is local and must exist in the work directory. “datafile” and “outputfile” either contain the full paths, or their paths are relative to the work directory. If include files are used, the work directory should be chosen as that containing the data file. If “datafile” or “outputfile” contain any blanks, including the expanded values of any environment variables, then they must be enclosed in quotes for execution. For example, the quotes are required in the command-line execution

```
sensor “%sensordata%\spe1.dat” spe1.out
```

because the path defined by the environment variable %sensordata% has blanks in it.

Binary output files containing all results (fort.*) are created in the work directory for plotting and mapping purposes or for input to other programs. Our examples use the file type conventions .dat for main Sensor data files (.inc for include files) and .out for the printed Sensor output files. Use the text editor (Notepad or Wordpad can be used) or Sensor Launcher or the interface of your choice to view or modify or create data and printed output files.

Multiple jobs may be submitted simultaneously, and this may have an adverse (hardware-related) effect on run cpu time performance. Each simultaneous job must be executed from within a different work directory.

While a job is running, most editors allow you to view the output file to the point where the simulation has reached. If your editor does not allow it, make a copy of the output file, and then edit the copy.

Using Include Files

Specifying the full path to any include files within datafiles eliminates any relative directory considerations at execution time, but it ties datafiles to specific machine path names affecting portability, so relative path specifications are often made. **The specification of any relative path to include files within data files applies relative to the work directory.** For example, assume that datafile case1run3.dat is in C:\sensor\data\studya\case1 and refers to an include file sat1.inc contained in directory C:\sensor\data\studya\includes. If the work directory is chosen to be a level below the data file, for example in C:\sensor\data\studya\case1\run3, then the relative

path reference to sat1.inc in case1.dat must be "..\..\includes\sat1.inc". If the work directory is chosen as the same as that containing the data, then the relative path reference to the include file must be "..\includes\sat1.inc". The simplest approach is to build all data and include files in the same directory, use no path names in execution or in references to include files, and to either work (execute) in the data directory or copy all data to the work directory.

Making Simultaneous and Sequential Runs

Simple batch files and directory structure are provided in the Run Set Folder (click on **Start/Programs/Sensor/Run Set/Main Folder** to view) for making up to 8 simultaneous sets of any number of sequential runs on a single node. They can easily be extended to as many simultaneous sets as desired. Sensor node-locked licenses do not restrict the number of simultaneous runs. You can choose the number of simultaneous sets to run in order to optimize the overall productivity of your system. The optimal number is generally equal to the number of cores or single-core processors and may be limited by available memory.

The name of the batch file that runs Set n is runsetn.bat. It is executed by clicking on **Start/Programs/Sensor/Run Set/n**. It can also be executed from Windows Explorer (by double-clicking) or by command line (enter name). Set n is executed in work directory (folder) runsetn. Set n results for all defined cases are saved in directory setnresults. On multiple executions of the same Run Set, previous results written to the setnresults folders are overwritten. On completion of Set n runs, (as we have written the batch files) folder runsetn will contain no files. You may need to save more results to the setnresults folders than we have moved in the batch files, such as any restart files (we have moved only the output file and the binary map and plot files). The setnresults folders will contain 2 additional files containing summary information for all runs made in the set, sensor.stat and sensorcpu.stat. These files are mainly used in testing.

The batch files runsetn.bat are pre-set to run some of the provided example cases. Set 1 will run the first 3 SPE Comparative Solution Project problems. At the end of the batch files, we have elected to open one of the case output files in Notepad, except for Set 2 (runs cases spe2a, spe5, spe7_1a, and spe9) which opens the set2results folder instead (the command 'opensetnresults' or 'opensetnresults.bat' opens the setnresults folder, where n = 1 to 8). Customize as desired. To run your cases, edit runsetn.bat (click on **Start/Programs/Sensor/Run Set**, right click on **n** (=1,2,…8) and select **Edit**) and change the specifications of our example data and output files to specifications of your data and output files. Add or remove as many cases as desired.

You can change the Run Set numbers in the Run Set menu, **n** = 1 to 8, to the names of your studies or cases or whatever your wish by right clicking on **n** and selecting **Rename**, but you should retain the original set number in the name, i.e. rename Set n from **n** to **name(n)**, to preserve the batch file and directory numbering association. Or, you can right click on the set number **n** and drag it to your Desktop, release, and then right click on the Desktop shortcut to rename it from **n** to **name(n)**. Then, simply double clicking on the Desktop icon(s) displaying your Run Set name(s) will run your cases. Do not change the names of the batch files or the directories in the Run Set Folder.

The directory 'data' is provided only for example and need not be used.

Using Include Files

If you refer to a data file in runsetn.bat that uses include files, then the reference to the include file in the data file must specify either the full path, or the path **relative to the runsetn work directory**, which is contained in the Run Set Folder. An example is provided in runset4.bat, which is pre-set to run spe10_case2.dat, which specifies include file spe10_case2.inc without any path. We have chosen to copy the data and include file to the work directory. The best alternative when using include files is to run Sensor in the directory containing the data file rather than using the runsetn directories. Any data files that you may wish to run sequentially (in different sets) must be in different directories.

Workflow Integrations

Workflow integrations with the Run Set structure to manipulate batch files, data, and results can be accomplished through the use of the environment variable %sensorappdata%, which points to the Run Set Folder and contains a trailing backslash.

Data errors and problems (KEYWORD)

The Sensor executable is dynamically dimensioned. At the beginning of execution, it scans the datafile down to the entry END to determine the necessary dimensions and checks for data errors. You do not need to enter any dimensioning data, either for restart runs or for runs from zero time.

If no errors are found in the following example datafile, execution occurs to 15200 days with restart records written at 7420 days and 15200 days.

```
TITLE       ! start Initial Data
   ...
ENDTITLE
GRID Nx Ny Nz
   ... initial data ..
ENDINIT     ! end of Initial Data

MODIFY PV   ! optional Modification Data
   .. modification data ..

WELL        ! start Recurrent Data
   .. well data ..
   .. other recurrent data at 0 time ...
RESTART
TIME 7420
   .. changes in well, rate, etc data ..
TIME 10150
   .. data changes ..
RESTART
TIME 15200
```

```
END          ! end of the datafile
```

An aid to finding data errors is the keyword KEYWORD. If KEYWORD is entered at any point in the datafile, the program will print each following keyword as it is successfully read. In the event of program read failure not identified in the output file by the internal error checking, the last printed keyword will help you locate the read error. To deactivate a previous entry of KEYWORD, enter KEYWORD OFF. The default status of KEYWORD is OFF. After all reading errors are corrected, remove KEYWORD from the datafile for normal production runs.

Edit your outfile and search for *ERROR and *WARNING. Certain errors can cause later false error messages. Correct the early errors which are clear and rerun before dealing with any later unclear errors.

Any dataset may run poorly if one or more blocks have sufficiently small pore volume and/or sufficiently large transmissibility. The program prints out a table of average and minimum/maximum values of grid block properties such as transmissibilities, thickness, and pore volume. An example of this table is given in Table 4. If some property is sufficiently extreme or non-physical, the problem may run poorly. Edit your outfile and search for "M/M" to view this table. In particular, look for very small gridblock pore volume or thickness, and for very large z-direction transmissibility. For example, if the minimum gridblock pore volume were 1.3 (rb) and the maximum T_z were 312000 (rb-cp/day-psi), then you might enter after the ENDINIT line (assuming a 43x31x7 grid):

```
MODIFY PV
1 43  1 31  1 7  CUT 100  ! zero all gridblock pv's less than 100 rb
MODIFY TZ
1 43  1 31  1 7  < 15000  ! restrict Tz to a maximum of 15000
```

CPU time control (CPULIM)

The optional Initial Data entry

```
CPULIM tcpu
```

results in termination of the run if cpu time exceeds tcpu minutes.

Run Completion Notification and Exit Codes

For a Sensor execution that successfully runs to the final time or date specified in the data file, Sensor will print “RUN COMPLETED” both at the end of the printed output file and to standard output (this is the computer terminal if manually executed), and the corresponding exit code of 0 is set in the Windows environment variable %errorlevel%. To see its value, execute the command

```
echo %errorlevel%
```

For a Sensor execution that encounters a data error or a program error resulting in normal termination, Sensor will print “RUN FAILED” both at the end of the printed output file and to standard output, and the corresponding exit code of -1 is set in the %errorlevel% environment variable. For a Sensor execution that encounters a program error resulting in abnormal

termination, no message will be printed to the output file or to standard output, but the exit code of -1 indicating run failure is set in the %errorlevel% environment variable.

1 Initial Data

The Initial Data are described here in an order in which they might normally be entered.

1.1 Run Title (TITLE, ENDTITLE)

The first datafile entries are normally TITLE, followed by ENDTITLE.

```
TITLE           !   optional
  ... any number of lines ...
ENDTITLE        !   if TITLE was entered
```

1.2 Grid Dimensions (GRID)

```
GRID  Nx Ny Nz  !   three integers
```

The model grid can be corner-point (xyz), Cartesian (xyz), or radial (cyclindrical, rθz). In the latter case, Nx is the number of blocks in the radial direction and Ny is the number in the angular direction. All grids are block-centered, i.e. the grid points are in the centers of their grid blocks. In the anisotropic xyz case, e.g. unequal k_x and k_y, the grid axes are assumed to be the principal axes.

Radial (cylindrical) grids (RADIAL)

The RADIAL option allows for internal calculation of radial and angular grid increments. If used, do not enter the DX and DY arrays. Radial grids are normally used only for single-well coning problems. The fully implicit formulation (not Impes) should be used for coning problems. Do not use turbulence with radial problems.

Definitions:

n = N_x, the number of blocks in the radial direction
r_w = wellbore radius, ft
r_e = exterior radius, ft
r_{bi} = inner block boundary radius, block i
r_{ci} = block center radius, block i
r_{b1} = rw

There are five options for setting values of block-boundary and block-center radii. The examples here are for n = N_x = 10 and N_y = 1. In all options below, the last line entered is N_y values of Δθ, degrees.

Option 1:

Only r_w and r_e are entered. The program calculates geometrically-spaced r_{ci} such that r_w is the logmean of r_{c0} and r_{c1} and r_e is the logmean of r_{cn} and r_{cn+1}. r_{bi} values are the logmean of r_{ci-1} and r_{ci}.

Example:

```
RADIAL
  1          ! option #
.33  2000    ! rw  re
 360         ! Ny Δθ values, degrees
```

Option 2:

Enter the first block-center radius r_{c1}, along with r_w and r_e. The program calculates geometrically spaced r_{ci}, i=2,n, such that r_e is the logmean of r_{cn} and r_{cn+1}. The value of r_{bi} is calculated as the logmean of r_{ci-1} and r_{ci}, except for $r_{b1} = r_w$.

Example:

```
RADIAL
  2          !  option #
 .33  2000   !  rw re
  2.5        !  rc1
 360         !  Ny values of Δθ, degrees
```

Option 3:

Enter block boundary radii r_{bi} (n values). The program calculates r_{ci} as the volume-average or volume centroid radius using r_{bi} and r_{bi+1}.

Example:

```
RADIAL
  3          ! option #
.25  2050.   ! rw re
.25 2. 4.32 9.33 20.17 43.56 94.11 203.32 439.24 948.92 ! n rbi values
 360.        ! Ny Δθ values, degrees
```

Option 4:

Enter block-center radii r_{ci}. The program calculates r_{bi} values as the logmean of r_{ci} and r_{ci-1}, except for $r_{b1} = r_w$.

Example:

```
RADIAL
  4          ! option #
 .5  1200    ! rw re
 2  4 8 16 32 64 128 256 512 1024 !   n rci values, ft
  360        ! Ny Δθ values, degrees
```

Option 5:

Enter the three radii r_w, r*, and r_e. The program calculates r_{bi} values such that: (a) if r*=r_e, there are N_x equal-volume blocks between r_w and r_e, (b) if r* < r_e, there are N_x-1 equal-volume blocks

between r_w and r* plus one block between r* and r_e. The block-center radii r_{ci} are then calculated as in option 3. N_x must be 3 or larger for this option.

Example for a lab experiment with an annulus:

```
RADIAL
   5                  ! option #
   0.   .0811   .115   ! rw r* re     core radius = .0811 ft
  360                 ! Ny Δθ values, degrees
```

1.3 Formulation and Solver Selection

Implicit formulation selection (IMPLICIT)

Impes is the default formulation. Enter

```
IMPLICIT   ! use the Implicit formulation
```

in Initial Data to select the fully Implicit formulation. For most problems, storage requirement and cpu time are significantly larger for the Implicit formulation. However, Implicit is generally faster than Impes for single-well coning type problems, lab-scale problems, foamy black oil problems, and dual porosity / dual permeability cases. The IMPLICIT option is automatically invoked for the latter.

In the general single porosity case, when multiple runs are to be made during a study, we recommend that the user make test runs to determine the most efficient formulation. The test runs should be long enough so that effects of early run performance on the timings are negligible.

Solver selection (NF, ILU, D4)

The default linear solver is Orthomin[5] preconditioned by RBILU(0) - red-black ILU(0) - which is usually the most efficient of the ILU variants. Appendix 1 describes other ILU variants which can be specified using the keyword ILU. Enter

```
NF
```

in Initial Data to select the Nested Factorization[6] solver. The NF solver is faster than the default ILU solver for some problems but the ILU solver is more reliable. Enter D4 to select reduced band width direct solution[7]. D4 is applicable only for small problems; it requires excessive cpu and storage for typical field-scale problems. For one-dimensional (1D) problems, do not enter NF or D4 or any other solver data.

In the general case, when multiple runs are to be made during a study, we recommend that the user make test runs (for the default ILU and optional NF) to determine the most efficient solver.

Appendix 1 describes ILU and NF solver parameters and gives their default values. These default values have been determined from many problems and we recommend that you do not change them, unless necessary.

Gas percolation control for Impes (PERC)

Some Impes problems require control of vertical gas percolation[8]. That is, they run more smoothly, with fewer steps, iterations, and lower cpu time, with little if any difference in results. In default mode, Sensor does not use percolation control. To activate gas percolation control, enter

```
PERC (no data)
```

in Initial Data. Also, it can be activated (enter PERC) or deactivated (enter NOPERC) in Recurrent Data. Any PERC entry is ignored in Implicit problems.

Nine-point difference scheme (NINEPOINT)

Sensor uses the conventional five-point difference scheme in the xy planes and seven-point scheme in 3D. The optional nine-point difference scheme in the xy plane reduces grid orientation effects in adverse-mobility ratio pattern floods. This is discussed in Appendices 2 and 10 in connection with datafiles test12.dat and test13.dat. The nine-point scheme is not needed in full field studies where gravity and heterogeneity effects dominate grid orientation effects. To activate nine-point differencing, enter

```
NINEPOINT
```

in Initial Data. The option can not be used: for 1-d problems, if either NX or NY is 1, if KX and KY are not both entered, or if any of the transmissibilities TX, TY, or TZ are entered.

Explicit well treatment (EXPLICIT)

For stability, Sensor treats all well terms implicitly in both Impes and Implicit formulations. For certain 1-D research type problems, it may be desirable to treat well terms explicitly. Never enter the following for any field-scale or 2D or 3D runs. It can be entered for 1D runs where timestep is tightly controlled to keep volumetric throughput ratios at wells < 1.0. The resulting explicit treatment of production well terms gives somewhat greater accuracy in such 1D runs. In field-scale or 2D or 3D runs this entry will give instability and high cpu:

```
EXPLICIT WELL
```

1.4 Fluid and Rock Properties

Constant data (MISC)

```
MISC Bwi cw Denw visw cf pref   ! water properties, rock
                                ! compressibility
```

B_{wi} = initial water formation volume factor, (vol at pref and Tres)/
(vol at 14.7 psia, 60 deg F), rb/stb
c_w = water compressibility at Tres, 1/psi
Denw = stock tank water density lbs/cu ft (or sp gr)

visw = water viscosity, cp
c_f = rock pore volume compressibility for grid blocks not assigned to a compaction table, 1/psi
p_{ref} = reference pressure for water volume factor, psia

At reservoir temperature and any p,

B_w = $B_{wi}(1 - c_w(p - p_{ref}))$ rb/stb
Dw = $(Denw / B_{wi})(1 + c_w(p - p_{ref}))$ lbs/cu ft
visw = constant water viscosity, cp

If MISC is omitted, Sensor sets defaults:

B_{wi} = 1 rb/stb
c_w = 3 x 10-6 1/psi
Denw = 62.4 lbs/cu ft
visw = .35 cp
c_f = 4 x 10-6 1/psi
p_{ref} = 4000 psia

For grid blocks not assigned to a compaction table, grid block pore volume PV and porosity POROS vary with pressure and formation (rock) compressibility c_f according to

PV = PVbase (1. + c_f (p – pbase))
POROS = POROSbase (1. + c_f (p – pbase))

The POROSBASE keyword is used to enter a global value of pbase. If POROSBASE is not entered, then pbase for each gridblock is equal to its initial pressure, and entered gridblock porosities or pore volumes, and PVbase, are defined as values at initial pressure. If POROSBASE is entered to specify a global value for pbase, then all entered gridblock porosities or pore volumes, and PVbase, are values at pbase.

The entered/printed/mapped arrays POROS and PV are POROSbase and PVbase in the above equations. The printed/mapped array PVC is PV in the above equation.

Reference pressure for porosity (POROSBASE)

```
POROSBASE  pbase   ! psia   optional

      pbase =     pressure at which entered porosities (or pore volumes) were
                  measured, psia
```

Do not enter POROSBASE if entered datafile porosities or pore volumes are values at initial grid block pressures. Enter POROSBASE only if datafile porosities or pore volumes were measured or apply at some pressure pbase.

Water viscosity by PVT type (VISW)

If water viscosity is to vary with pvt type, then use the entry

VISW

```
   ipvt  visw   ! as many lines as desired
```

where ipvt = pvt type (integer) and visw = water viscosity, cp. The visw value entered under VISW will override the visw value entered on the MISC dataline.

Standard conditions (PSTD, TSTD)

```
PSTD pstd    ! standard pressure, psia      optional, default=14.7
TSTD Tstd    ! standard temperature, deg F  optional, default=60
```

If all pvt types are black oil, these data are not needed or used.

Black oil pvt data (PVTBO)

Secondary keywords – ZGAS, PRESSURES, DENSITY, PSAT, RS, SRS, COIL, CVOIL, IFT, BO, VISO, BG, ZG, VISG

Enter all PVTBO data before entering any INITIAL data.

PVTBO tables may also be re-entered as often as desired in the Recurrent Data. The re-entry alters no grid block saturation pressures or saturations, but a step change in GOR will occur. It may be important to update any well tubinghead pressure tables in use at the time of PVTBO re-entry. Example files for a small, radial, single well, gas condensate problem are test16.dat, test16.out, test16.sp (SensorPlot data file), and test16.xls (Plot2Excel plots).

Following PVTBO, the keyword entries DENSITY and PRESSURES define surface densities and the pressure vector $\{p_i\}$:

```
PVTBO n
(ZGAS tres)
PRESSURES  nsat ntot (PSIG)
p1 p2 ... pnsat ... pntot       ! ntot values
DENSITY    deno  deng  coil cvoil
```

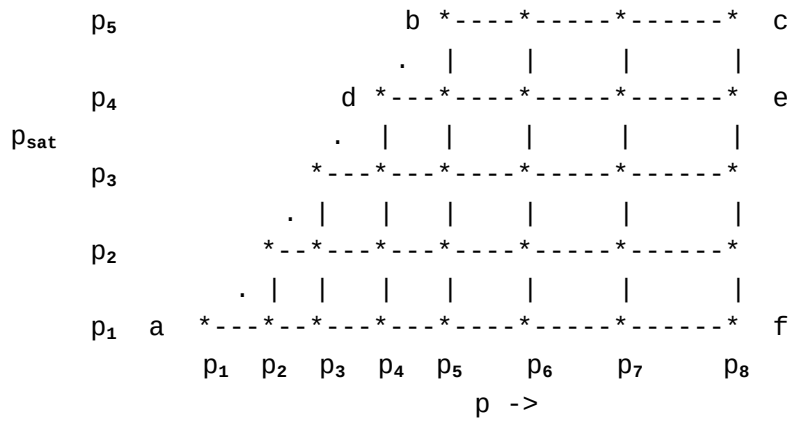
where

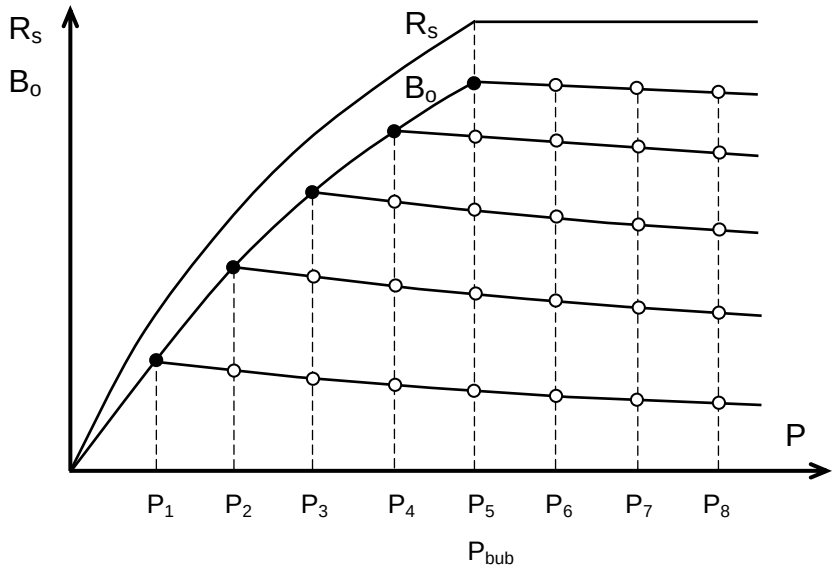
n	= pvt type (internally set to 1 if omitted)
ZGAS	= optional label indicating that Zg rather than Bg values will be specified
tres	= reservoir temperature, required if Zg values are specified
n_{sat}	= number of saturated pressures
n_{tot}	= total number of pressures, > or equal to nsat
$p_1,p_2,..,p_{ntot}$	= ntot monotonically increasing pressures, psia
deno	= oil surface density, lb/cu ft or specific gravity (water = 1.0)
deng	= gas surface density, lb/cu ft or specific gravity (air = 1.0)
coil	= oil compressibility, used for oil densities in undersaturated region unless undersaturated Bo data are specified, 1/psi
cvoil	= oil viscosity coefficient, used for oil viscosities in undersaturated region unless undersaturated oil viscosities are specified, 1/psi

The remaining black oil pvt data are specified in tabular form and include:

R_s	dissolved gas in oil phase, scf/stb
r_s	vaporized oil in gas phase, stb/mmcf
ift	gas-oil interfacial tension, dynes/cm
B_o	oil phase formation volume factor, rb/stb
B_g or Zg	gas phase formation volume factor, rb/scf, or compressibility factor
viso	oil phase viscosity, cp
visg	gas phase viscosity, cp

Data are required at the points noted on the following grid:





In the above grid, $n_{sat} = 5$ and $n_{tot} = 8$. The remainder of the pvt data consists of three tables, each preceded by a header line specifying the columns of data included. These three tables are the Saturated Table, the Undersaturated Table, and the B_g Table. The Saturated Table specifies data along the saturated envelope ab and has the following eligible header symbols:

PSAT RS SRS COIL CVOIL IFT BO VISO BG (or ZG) VISG

The Undersaturated Table enters data in the undersaturated region abcf and has the eligible header symbols:

PSAT P BO VISO BG (or ZG) VISG

If B_g (Z_g) and visg are represented as single-valued functions of p, then the B_g (Z_g) Table can be used with header symbols P BG (ZG) VISG. Whenever r_s (oil vaporized in gas phase) is constant (e.g. 0), B_g (Z_g) and visg are single-valued functions of p. The default value of r_s is 0 if r_s is not entered using SRS in the Saturated Table.

If B_o and viso are not entered in the Undersaturated Table, then they are calculated at the points noted by asterisks in the region abcf of the above grid, using c_{oil} and c_{voil} and B_o,viso values entered in the Saturated Table:

$B_o(p,p_{sat}) = B_o(p_{sat},p_{sat})\ e^{\{-coil*(p-psat)\}}$

$viso(p,p_{sat}) = viso(p_{sat},p_{sat})\ e^{\{cvoil*(p-psat)\}}$

There are numerous options to simplify entry of black oil pvt data. Major simplification in data entry is possible for:

two-phase water-oil problems $n_{sat}=1$

two-phase gas-water problems $n_{sat}=1$

three-phase problems with approximate representation of undersaturated Bo, B_g, viso, visg.

two- or three-phase research-type problems with incompressible gas, oil, and water.

The input data are illustrated by example cases. You can save time by skipping to the example below which fits your problem:

Example 1 Two-phase water-oil
Example 2 Two-phase gas-water
Example 3 Three-phase with constant r_s
Example 4 Three-phase with variable r_s
Example 5 Three-phase with variable r_s and rigorous undersaturated treatment
Example 6 Research-type problems with incompressible water, oil, and gas

Some example data files for PVTBO are spe1.dat, spe2.dat, spe7_4a.dat, spe9.dat, spe10_case1.dat, spe10_case2.dat.

Notes:

1. For two-phase problems, n_{sat} must be 1.
2. If entered pressures are psig, append the word PSIG after ntot on the PRESSURES dataline.
3. Reservoir pressures must lie between p_1 and p_{ntot}.
4. The model uses bilinear interpolation in the above grid abcf.
5. Extrapolation is used at pressures exceeding p_{ntot}.
6. The order of pressure entries in the three tables is immaterial.
7. The pressures $\{p_i\}$ do not need to be equally spaced.
8. If undersaturated B_o and viso are entered in the Undersaturated Table, as in Example 5 below, c_{oil} and c_{voil} are not used.
9. In the multiple pvt case, enter one block of PVTBO data for each black oil pvt type.
10. The program internally assigns component names of OIL and GAS to components 1 and 2, and WATR to water (component 3).

Arbitrary units (UNITS)

Arbitrary sets of units can be used in the PVTBO data through specification of conversion factors. Entry of

```
UNITS  c1 c2 c3 c4 c5 c6 c7 c8 c9
```

immediately following the PVTBO data line converts entered units to standard units as follows:

p psia = c1 * p(entered) + c2
Rs scf/stb = c3 * Rs(entered)
Bo rb/stb = c4 * Bo(entered)
rs stb/mmcf = c5 * rs(entered)
Bg rb/scf = c6 * Bg(entered)
Viscosity = c7 * viscosity(entered)
coil 1/psi = c8 * coil(entered)
cvoil 1/psi = c8 * cvoil(entered)
deno lb/ft3 = c9 * deno(entered)
deng lb/ft3 = c9 * deng(entered)

The data for Example 5 given in Table 15 provide an example of the use of UNITS.

Example 1. Water-oil problem:

A water-oil problem is one where no free gas saturation exists. The reservoir contains water and undersaturated oil. The value of n_{sat} must be 1. No B_g, visg, ift, or r_s data are required.

```
PVTBO
DENSITY  .836  .066  .00001428   .00006875  ! deno deng coil cvoil
PRESSURES 1 14
2040 2500 3000 3500 4000 4500 5000 5500 6000 6500 7000 8000 9000 10000

C     SATURATED TABLE
PSAT       BO        RS          VISO
2040.0    1.4207     502.0         .38
```

Undersaturated B_o,viso values could be entered in an Undersaturated Table if desired:

```
C  UNDERSATURATED TABLE (coil,cvoil are not used)
PSAT    P      BO    VISO
2040   2040  1.4027   .38
       2500    --     --
       3000    --     --
       ...
      10000    --     --
```

Example 2. Gas-water problem:

A gas-water problem is one where no oil saturation exists. The reservoir contains water and undersaturated gas. The value of n_{sat} must be 1. No B_o, viso, c_{oil}, c_{voil}, ift, or R_s data are required. The r_s value is normally 0 but may be > 0. The c_{oil} and c_{voil} values are not used. B_g and visg are single-valued functions of p and must be entered in the B_g Table.

```
PVTBO
      DENSITY  .721 0.605  0.  0.
      PRESSURES 1 11
      14.7 500 1000 1500 2000 2500 3000 3500 4000 4500 5000

C  SATURATED TABLE
      PSAT  SRS
      14.7  3.25

C  BG TABLE
      P      BG       VISG
      14.7  0.221143  0.01269
       500  0.00626    .01313
      1000  0.003032  0.01401
```

```
1500  0.001975   .01486
2000  0.001464  0.01585
2500  0.001172  0.01711
3000  0.000987  0.01847
3500  0.000863  0.01987
4000  0.000775  0.02128
4500  0.00071   0.02268
5000  0.00066   0.02409
```

Example 3. Three-phase with constant r_s:

B_g and visg are single-valued functions of p because r_s is constant (0). Undersaturated B_o and viso values are generated from constant c_{oil} and c_{voil} values. The Undersaturated Table is not required. Data are from datafile spe1.dat.

```
PVTBO
      DENSITY  49.1   .792  1.37E-5  9.02E-5   ! deno deng  coil cvoil
      PRESSURES  11 11 PSIG
      0 500 1000 2000 3000 4000 5000 6000 7000 8000 9000

C  SATURATED TABLE
       PSAT       BO        BG          RS       VISO   VISG
      9000.00   2.35700  0.00038600   2984.00  0.2030  0.0470
      8000.00   2.22450  0.00045175   2642.50  0.2645  0.0430
      7000.00   2.09200  0.00051750   2301.00  0.3260  0.0389
      6000.00   1.95950  0.00058325   1959.50  0.3875  0.0349
      5000.00   1.82700  0.00064900   1618.00  0.4490  0.0309
      4000.00   1.69500  0.00081100   1270.00  0.5100  0.0268
      3000.00   1.56500  0.00108000    930.00  0.5940  0.0228
      2000.00   1.43500  0.00161400    636.00  0.6950  0.0189
      1000.00   1.29500  0.00319700    371.00  0.8300  0.0140
       500.00   1.20700  0.00627400    180.00  0.9100  0.0112
         0.00   1.06200  0.16666600      0.00  1.0400  0.0080
```

If variable c_{oil} and c_{voil} values are desired then (enter c_{oil} and c_{voil} zero on the DENSITY dataline):

```
PVTBO
DENSITY  49.1   .792  0.   0.   ! deno deng  coil cvoil
PRESSURES  11 11 PSIG
0 500 1000 2000 3000 4000 5000 6000 7000 8000 9000

C  SATURATED TABLE
PSAT    RS      COIL    CVOIL      BO        BG      VISO   VISG
9000   2984    1.37E-5  9E-5     2.357   .000386    .2030  .0470
8000   2642.5  1.35E-5  8.8E-5   2.2245  .00045175  .2645  .0430
```

```
7000   2301    1.33E-5  8.6E-5   2.092   .0005175   .3260  .0389
6000   1959.5  1.31E-5  8.4E-5   1.9595  .00058325  .3875  .0349
5000   1618    1.29E-5  8.2E-5   1.82    .000649    .4490  .0309
4000   1270    1.27E-5  8E-5     1.695   .000811    .5100  .0268
3000    930    1.25E-5  7.7E-5   1.565   .00108     .5940  .0228
2000    636    1.23E-5  7.4E-5   1.435   .001614    .6950  .0189
1000    371    1.2E-5   7E-5     1.295   .003197    .8300  .0140
 500    180    1.1E-5   6.5E-5   1.207   .006274    .9100  .0112
  0      0     1E-5     6E-5     1.062   .166666   1.0400  .0080
```

If no gas is to go into solution above the initial bubble point pressure of 4000 psig,

```
PVTBO
      DENSITY  49.1   .792  1.37E-5  9.02E-5   ! deno deng coil cvoil
      PRESSURES  6 11 PSIG
      0 500 1000 2000 3000 4000 5000 6000 7000 8000 9000

      C  SATURATED TABLE
      PSAT       BO        BG          RS       VISO   VISG
      4000.00   1.69500  0.00081100   1270.00  0.5100  0.0268
      3000.00   1.56500  0.00108000    930.00  0.5940  0.0228
      2000.00   1.43500  0.00161400    636.00  0.6950  0.0189
      1000.00   1.29500  0.00319700    371.00  0.8300  0.0140
       500.00   1.20700  0.00627400    180.00  0.9100  0.0112
         0.00   1.06200  0.16666600      0.00  1.0400  0.0080

C     The above Saturated Table does not specify Bg and visg values
C     at pressures above 4000 psig.  Therefore the Bg Table is
C     required:

C   BG TABLE
         P          BG        VISG
      9000.00   0.00038600   0.0470
      8000.00   0.00045175   0.0430
      7000.00   0.00051750   0.0389
      6000.00   0.00058325   0.0349
      5000.00   0.00064900   0.0309
```

Alternatively, the BG and VISG columns could be left out of the Saturated Table and the B_g Table could include lines for all pressures 0,500,1000, ...,9000.

Undersaturated B_o and viso values could be entered in an Undersaturated Table with header PSAT P BO VISO, as in Example 5. Values of c_{oil} and c_{voil} would not be used in that case.

If a variable $r_s > 0$ were present here, then either of two approaches could be adopted after adding the SRS column in the Saturated Table:

(a) Neglect the variation of B_g and visg with p_{sat}, and enter them as single-valued functions of p in the B_g Table.

(b) Enter the undersaturated B_g and visg as functions of p and p_{sat} using the Undersaturated Table, as in Example 5.

Example 4. Three-phase with variable r_s:

In this example, B_g and visg are treated as single-valued functions of p. The data represent a system where, above the 5500 psia bubble-dew point, R_s and r_s are constant. Undersaturated B_o and viso values are generated from constant c_{oil} and c_{voil} values. The Undersaturated Table is not required. Since the Saturated Table contains B_g and visg values only up to 5500 psia, the B_g Table is necessary to provide B_g and visg data at pressures above 5500 psia.

```
PVTBO 2
      DENSITY  49.499 0.0589   0.0000139  0.0000204
      PRESSURES    8  13 ! 8
      700 1000 2000 3000 4000 4500 5000 5500 5835 6000 7000 8000 8500
C   psia     rb/stb     rb/scf     scf/stb     cp     cp
C       *** SATURATED TABLE ***
      PSAT     BO       BG            RS       VISO   VISG    SRS
       700   1.25906  0.0049355    138.315  0.76798  0.01426   24
      1000  1.30081  0.00338837   210.872  0.6541   0.01493   28.89
      2000  1.45715  0.00163998   493.462  0.41131  0.01833   36.55
      3000  1.64461  0.00111463   849.376  0.27886  0.02481   62.372
      4000  1.87676  0.00095    1307.241  0.1999   0.03548  108.661
      4500  2.02339  0.00092    1599.072  0.17033  0.04328  143.673
      5000  2.22695  0.0009    1993.166  0.14184  0.05507  193.789
      5500  2.72355  0.00088    2869.41   0.09969  0.08336  294.358

 C        *** BG TABLE ***
       P      BG        VISG
      5835  0.00086     0.08826
      6000  0.00084     0.09072
      7000  0.00082     0.10638
      8000  0.00081101  0.15006
      8500  0.00079881  0.16119
```

Example 5. Three-phase with variable r_s, rigorous undersaturated pvt:

This is from test4.dat. The data were generated from an equation of state for a critical-point fluid. The data are voluminous and are given in Table 15.

Example 6. Research-type problems with incompressible water, oil, and gas.

Formation volume factors of 1.0 are entered here to simplify later rate entries. R_s and r_s are (default) 0 in the following examples. For a three-phase case (e.g. spe10, case 1),

```
C     Bwi  cw     denw   visw    cf      pref
 MISC  1    0.     62.4    1.    4.E-6    1000

 PVTBO 1
C           deno     deng     coil     cvoil
 DENSITY  43.68    .0624      0.        0.
 PRESSURES 1 2
 14.7  3000.
 PSAT      P        BG    VISG   BO    VISO
 14.7    14.7  .178107   .01    1.     1.
         3000  .178107   .01    1.     1.
```

For a two-phase water-oil problem,

```
C     Bwi    cw    denw  visw  cf    pref
 MISC  1.     0.    64.   1.  1.E-6  4000

 PVTBO 1
 DENSITY     53.00  .0624   0.  0.
 PRESSURES  1  2
 300.   8000.
 PSAT      P     BO    VISO
  300.   300.     1.    2
        8000.     1.    2
```

At least one of water, oil, gas, and rock must be compressible. For example, the above two examples will not run if cf is entered 0.

Discussion of undersaturated conditions with varying rs (vaporized oil in gas phase)

Let p1,p2,p3,... be the pvt table monotonially increasing pressures. Let rs be > 0 with a hook in the rs vs saturated pressure curve (a minimum at some intermediate psat) and let p* be the saturation pressure at which rs is a minimum. Then undersaturated phase data occur as follows:

For any given psat, an undersaturated oil phase exists at all pressures pi > psat

For any given psat, an undersaturated gas phase exists

at all pi < psat and all pi > psat if psat=p*
at all pi < psat if psat < p*
at all pi > psat if psat > p*

That is, the undersaturated pressures (pgus) for the gas phase obey

pgus < psat for psat <= p*

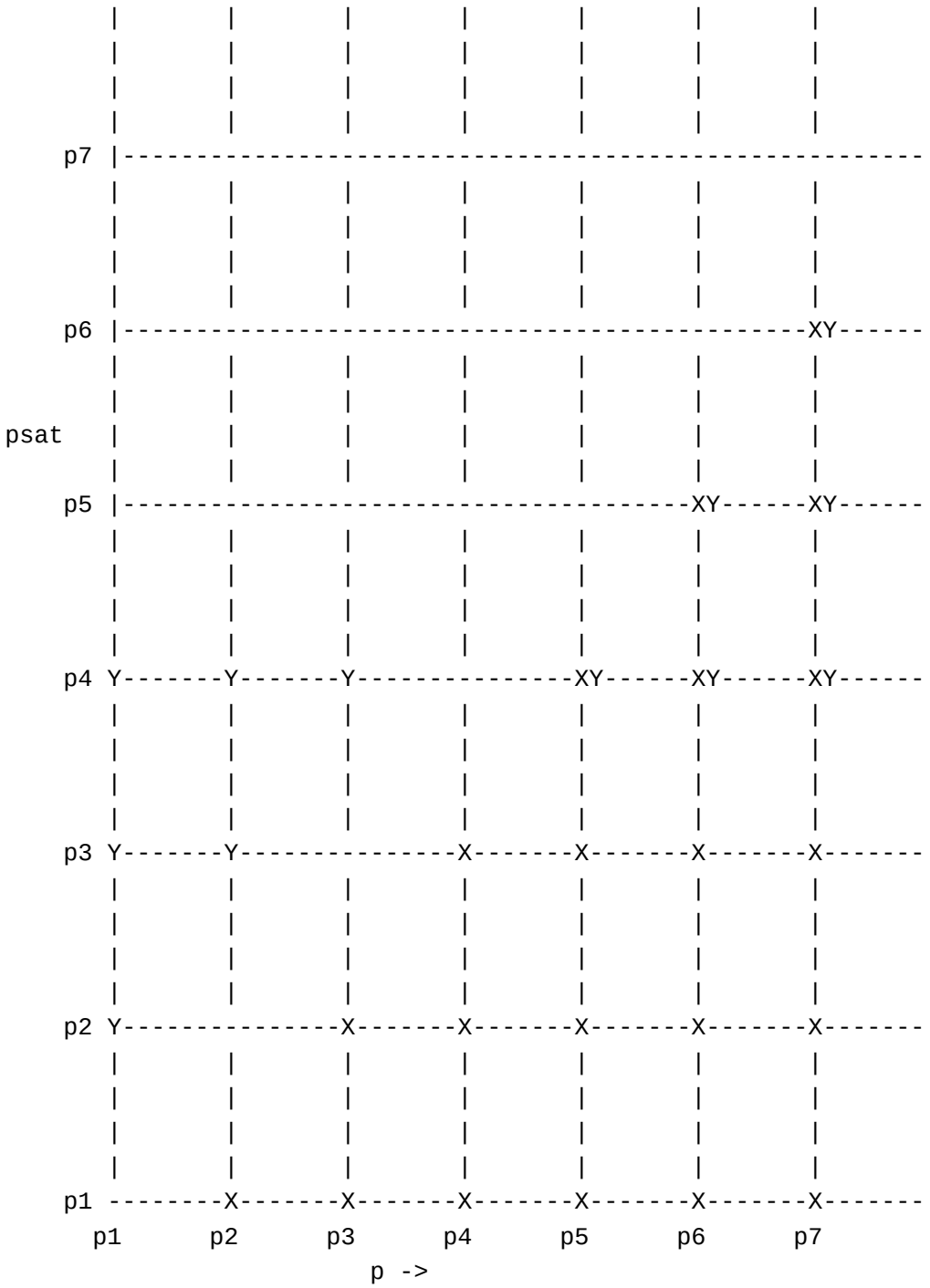
pgus > psat for psat >= p*

For the gas phase with p* > p1, any psat < p* is a lower dew point, any psat > p* is an upper dew point, and psat=p* is both an upper and lower dewpoint.

As an example, the figure below marks undersaturated oil points (X) and undersaturated gas points (Y) for the case where p*=p4.

The printout is a single table of undersaturated properties of undersaturated oil and gas. The first line of each table subsection lists properties at psat and p = psat. For oil, there is no undersaturated condition possible for p < psat(bp); all undersaturated oils exist only at p > psat(bp). If rs is entered with a 'hook' and p1 < p* < pnsat (p* is the pressure at which rs = a minimum) then, for gas, (a) if p < p*, undersaturated condition exists only if p < psat(dp) and (b) if p>p*, undersaturated condition exists only if p > psat(dp).

The ordering of listed pressures p in the table subsection is monotonically increasing except for the first psat entry where p=psat. Only undersaturated entries are printed. If there is no rs hook, then all undersaturated entries are for p > psat and therefore the second subsection p entry is > psat. If there is an rs hook there will be no oil undersaturated entries for any p < the subsection psat. However, there will be gas undersaturated entries at all table pressures p < the subsection psat. That fact explains the appended "←UNDERSATURATED GAS DATA". The 0's printed for the oil properties on those lines reflect the fact that oil of the subsection psat cannot exist in an undersaturated state for p<psat. Any 0's for undersaturated gas Bg,visg at p > subsection psat reflect the fact that there cannot be any undersaturated gas at p>subsection psat if subsection psat < p*.



An example datafile is blackoil.dat. Note the undersaturated pvt printout in blackoil.out. For each psat, it prints the (redundant) saturation point. Then it lists monotonically increasing undersaturated pressure points corresponding to the above figure.

The black oil pvt table of blackoil.dat is created from entered PVTEOS data, using the keyword BLACKOIL. PVT "error" messages are generated by code which simply checks all saturation table columns (Bo,Rs,rs, etc.) for any reversals (violations of monotonicity vs pressure). As printed by Sensor, a single reversal or hook in an rs curve is normal and should be accepted as such.

PVT error override (RUN)

If the pvt "error" message occurs, you should check your pvt table printout to ensure that exceptions to monotonicity are ok. Then add the Initial Data entry RUN to your datafile and it will ignore the error(s) and run.

Note in blackoil.out this "error" message is accompanied by the printout that a reversal in rs at the hook (minimum rs) point is normal. The datafile runs in spite of the printed "error" because the BLACKOIL keyword automatically activates the entry "RUN".

Arguably, an overwhelming portion of 40+ years' black oil pvt tables used in simulation would not give the "is not monotonic" printout 'ERROR' message. That's because rs wasn't entered or didn't have a hook if it was entered or because gas condensate near-dewpoint property reversals didn't exist. In those cases all black oil pvt table properties vs psat are monotonic. For example, spe1,spe2, and spe9. Non-monotonicity may be present correctly due to two things – rs hook or near-dewpoint reversals. Or it may be present incorrectly due to countless mis-types (typos). It is difficult to separate correct vs incorrect. Therefore sensor simply checks for any non-monotonicity (NM) and, if found, alerts the engineer to check his pvt table that the NM is ok. Sensor does print the message that the rs hook NM is normal if present.

In coal bed methane problems (see Section 1.9), the artificial oil properties (B_o=1) will cause errors to be detected in the PVTBO data ($B_o - R_s B_g < 1$). The keyword RUN should be entered in the Initial Data to override these errors.

Foamy oil option for black oil (FOAM)

The foamy oil option is applicable to black oil fluids in both IMPES and Implicit formulations (Implicit is recommended) and is described in detail in Appendix 3. The Initial Data entry is

```
FOAM sgemax vismult
```

where

```
sgemax  = maximum value of entrained gas saturation, fraction
vismult = coefficient for emulsion viscosity
```

Automatic conversion of EOS to black oil table (BLACKOIL,ENDBLACKOIL)

Secondary keywords – EXTEND, PRESSURES, RESERVOIR FLUID, INJECTION GAS, SURFACE, EQUILIBRIUM, SEPARATOR

The Sensor formulation uses the same compositional logic and code for both compositional and black oil problems. The data file for a given problem is therefore nearly the same regardless of whether the problem is to be run in compositional or black oil mode. The keyword BLACKOIL is an additional step toward making it simple to run a problem in either black oil or compositional mode.

A compositional datafile (e.g. spe3.dat,spe5.dat) contains a PVTEOS data section giving the eos parameters for the multicomponent compositional simulation. If the keyword BLACKOIL and its few following datalines are inserted immediately above the PVTEOS dataline, then Sensor uses the method of Whitson and Torp[44] to internally generate the black oil pvt table from the PVTEOS data, and then uses the black oil table for pvt properties rather than the eos.

If several PVTEOS pvttypes exist, then one or more but not necessarily all can be converted to black oil by the BLACKOIL keyword.

In order to obtain better agreement between compositional and black oil runs, saturation pressures in the generated black oil table may be elevated above the psat of the original reservoir fluid[45] by entering one of the optional INJECTION GAS keywords. Elevated saturation pressures are obtained by adding increments of Injection Gas to the original reservoir fluid in a procedure developed by Curtis Whitson of Pera A/S. The INJECTION GAS keywords define the Injection Gas to be used in elevating the table as either: SURFACE – the surface gas obtained by flashing the original reservoir fluid through the entered SEPARATOR, EQUILIBRIUM - bubble point equilibrium gas (applies only when the original reservoir fluid is an oil), or as a specified composition.

The Initial Data BLACKOIL entry (preceding PVTEOS) is

```
BLACKOIL    ipvttype  nsat  ntot (EXTEND)
 PRESSURES  p(1) p(2) p(3) ... p(nsat) p(nsat+1) ... p(ntot) ! psia
 RESERVOIR FLUID
   z1 z2 z3 ... zn ! original reservoir fluid composition, mol fractions
 INJECTION GAS SURFACE      ! use surface gas to elevate table psat, or
 INJECTION GAS EQUILIBRIUM  ! use equilibrium gas to elevate psat, or
 INJECTION GAS              ! use following gas to elevate psat:
   y1 y2 y3 ... yn          ! specified gas composition, mol fractions
 SEPARATOR    ! any entered SEP data will be ignored
   p1   T1    ! stage 1 pressure (psia) and temperature (deg F)
   p2   T2    ! stage 2
    ...
   pnstage Tnstage  ! stage nstage
ENDBLACKOIL
```

where

ipvttype	= pvt type
nsat	= number of saturated pressures for the black oil pvt table.
ntot	= the total number of pressures for the black oil pvt table.
p(nsat)	= maximum psat for the black oil pvt table.
p(1),p(2),..,p(ntot)	= ntot monotonically increasing pvt table pressures, psia
z1 z2 ... zn	= original reservoir fluid composition, mol fractions
y1 y2 ... yn	= Injection Gas composition, mol fractions

The optional label EXTEND causes the program to split out the two black oil components into an extended compositional component description, printed in the Timestep Table. The components of this extended description are the same as the components in the PVTEOS table.

This EXTEND on the BLACKOIL dataline is permissible only in datasets where all pvt types are black oil. The extended composition file write to unit 23 will be active.

Additional minor changes required to convert a compositional run datafile to a black oil datafile are:

Substitute the black oil format INITIAL data for the compositional format INITIAL data.

Change any compositional format INJGAS, OUTSIDE, YPLAT (gas compositions) data to black oil format.

If BLACKOIL is entered, surface separation conditions are entered under the subsidiary keyword SEPARATOR. Any SEP data entered are ignored.

Notes:

1. The ntot value must be greater than or equal to nsat and less than or equal to 30.
2. The original reservoir fluid psat (bubble- or dew-point) must be one of the entered pressures. Its value need not be exact as its entered value will be replaced by the internally calculated psat of the original reservoir fluid.
3. The maximum pressure p(ntot) should exceed the highest grid block pressure occurring in the simulation. Initial pressure and default or entered injector bhp values should assist in apriori selection of p(ntot).
4. The lowest table pressure p(1) should be lower than the lowest grid block pressure occurring in the simulation. Default or entered producer bhp values should assist in selection of p(1).
5. If the maximum pressure p(ntot) is not high enough, the run will repeatedly cut time steps (and probably stop) with warning messages that the maximum table pressure is not high enough.
6. BLACKOIL data must immediately precede the PVTEOS data to which they apply.

Example 1

The test datafile spe3.dat is the SPE Third Comparative Solution Project Problem. The nine-component retrograde gas condensate has an original 3480 psia dew-point and initial reservoir pressure is 3550 psia. Gas cycling using Surface Gas for 3650 days is followed by blowdown to 5480 days. The BLACKOIL entry in that datafile to run the problem in black oil mode is:

```
C        ipvttype   nsat  ntot
BLACKOIL    1        8     9    EXTEND
 PRESSURES  14.7 400 1000 1500 2000 2500 3000 3480 4000
 RESERVOIR FLUID
  .0121 .0194 .6599 .0869 .0591 .0967 .0472 .0153 .0034
 SEPARATOR
   815.  80.
   315.  80.
    65.  80.
    14.7 60.
ENDBLACKOIL
```

The black oil datafile spe3_bo.dat is included in the file spe3.dat. The unedited Plot2Excel plots comparing black oil and compositional results are in the file spe3.xls.

Example 2

The test datafile spe5.dat is Scenario 2 of the SPE Fifth Comparative Solution Project Problem. Wag injection with an enriched gas is used in 90-day cycles to attain a miscible displacement at about 3500 psia. Original oil bubble-point is 2302 psia and a six-component eos compositional fluid description is used. The BLACKOIL entry immediately preceeding the PVTEOS dataline is:

```
BLACKOIL 1  11  12   EXTEND
PRESSURES  500 800 1100 1400 1700 2000 2302 2600 2900 3300 3800 5000
RESERVOIR FLUID
.5 .03 .07 .2 .15 .05   ! original reservoir oil
INJECTION GAS
.77 .2 .03 3*0
SEPARATOR
14.7   60.
ENDBLACKOIL
```

where p(nsat) = 3800 psia is the specified maximum elevated psat. The black oil datafile spe5_bo.dat is included in the file spe5.dat. The file spe5.xls has unedited Plot2Excel plots comparing compositional results to black oil results with (Run b) and without (Run c) elevated psat.

Other Examples:

Datafile test3.dat has five runs a,b,c,d,e for compositional and black oil simulations of CO2 and Surface Gas cycling of an undersaturated oil reservoir at an elevated pressure. See the explanatory TITLE comments and data in test3.dat. Unedited plots comparing the five runs' results are in test3.xls. Datafile test4.dat has compositional and black oil runs for depletion of an at-critical reservoir with a large variation of initial composition with depth. Plots in test4.xls compare the compositional and black oil results. Also see datafile blackoil.dat. SensorPlot datafiles are spe3.sp, spe5.sp, test3.sp, and test4.sp.

First contact miscibility option (FCM)

First contact miscibility for compositional pvt types may be specified in two ways as described below. The second option provides significant gains in efficiency through internal 2-component pseudoization of the eos description, and allows specification of bypassed oil and dispersion control/extension.

No flash or saturation pressure calculations are performed for the first contact miscible case. All blocks are two-phase oil-water for all time. The simulation involves only two-phase hydrocarbon-water blocks. The rel perm of the hydrocarbon ("oil") phase, k_{rh}, is a function of

water saturation S_w, which allows water injection or WAG in addition to solvent ("gas") injection. If the miscibility assumption is not warranted, large errors may occur. The composition of the hydrocarbon phase in a grid block will change in space and time, from original oil composition to solvent composition. The eos (or its pseudoization) is used to compute the density of this phase as a function of its composition and pressure. The relative permeability $k_{rh}(S_w)$ is interpolated between $k_{row}(S_w)$, entered in the oil-water rel perm Table, and krg (evaluated at $S_g=1-S_w$), entered in the gas-oil rel perm Table. The interpolation is linear in Z where Z is the time-variant mol fraction of solvent in the block's hydrocarbon phase. Thus as a block progresses from original oil to solvent composition, its $k_{rh}(S_w)$ will continuously change from the oil-water $k_{row}(S_w)$ to the gas $k_{rg}(S_w)$ (evaluated at $S_g=1-S_w$). The end-point hydrocarbon saturation to water displacement linearly changes from S_{orw} to S_{gr} as solvent mol fraction Z increases from 0 to 1.0.

First contact miscibility assumes that all blocks are undersaturated for all time. A nearly equivalent assumption is that the psat-Z phase diagram exhibits a closed curve (phase envelope) with a maximum pressure less than field operating pressure. In reality, the field may have space-time regions and/or near-producer regions which are saturated. The usage of first contact miscibility in such cases assumes that three-phase effects in such regions on overall field results are small.

Water injection or WAG may be used. Pcwo data may be entered or omitted as desired.

KRIFT and Pcgo play no roles in first contact miscibility blocks.

First contact miscibility treatment can be invoked either through specification of the OIL keyword in the INITIAL data, or through specification of the FCM keyword.

Specify first contact miscibility using Keyword OIL

If OIL is entered on the INITIAL dataline:

1. Calculations are performed in Nc-component eos mode.
2. Initial composition may vary spatially.
3. Multiple pvt types are allowed.
4. If you want first contact miscibility treatment in part of the grid but not in other parts, put OIL on Initialization Regions' INITIAL lines only where you want first contact miscibility treatment.

Specify first contact miscibility using Keyword FCM
Secondary keywords – OIL, SOLVENT, SEPARATOR

If the few-line FCM datablock is entered:

1. Calculations are performed in pseudo two-component eos mode.
2. Initial composition must be uniform throughout the reservoir.
3. Only 1 pvttype is allowed
4. Injected solvent composition must be uniform in space and time.
5. The full grid is treated using first contact miscibility.

6. INJGAS entries are not needed and are ignored if present.
7. OUTSIDE entries are recognized but the gas composition is ignored, replaced internally by 0. 1.
8. YPLAT entries are not needed and are ignored if present (any platform reinjected gas composition is internally set to 0. 1.)
9. Initial composition under INITIAL must be 1. 0.
10. OIL on the INITIAL line is not needed, is ignored if present.
11. FCM data must be placed immediately before the PVTEOS dataline.

In this FCM case, the program internally generates two pseudo components: component 1 = original reservoir oil, component 2 = solvent. This internal pseudoization is rigorous - hydrocarbon fluid density and viscosity, as functions of pressure and composition, calculated in two-pseudo-component mode are the same as calculated in Nc-component mode. The two-component surface separation logic gives the same gor and surface fluid densities as would be calculated in the full Nc-component mode. The Timestep Table prints extended Nc-component results and extended Nc-component field compositions are written to Fortran Unit 23 (see Manual, keyword EXTEND, for description of that write). The program automatically performs this extended Nc-component printout and write - do not enter the keyword EXTEND.

Runs using OIL and FCM will give the same results but the FCM run should be faster. For example, spe79692.dat (Nc = 7) using implicit takes 10 cpu seconds using OIL and 2 cpu seconds using FCM (2.8 GHz desktop). Storage is also significantly less for FCM vs OIL.

FCM data are entered (immediately before the PVTEOS data) as

```
FCM (x*) (K)
 OIL
   z1 z2 z3 ... zn   ! mole fraction composition of original oil
 SOLVENT
   y1 y2 y3 ... yn   ! mole fraction composition of injected solvent
 SEPARATOR
   p1   T1    ! stage 1 pressure (psia) and temperature (deg F)
   p2   T2    ! stage 2
    ...
   pnstage Tnstage  ! stage nstage
```

Bypassed oil and dispersion control with the FCM option are as follows:

x* = mol fraction of bypassed oil

K = dispersion control coefficient

Default x* = 0. Default K = 1.0

If both x*, K are omitted, default values are used

If x* is entered but K is omitted, default K=1 is used

Oil is component 1, solvent is component 2.

X_1,X_2 are flowing mol fractions or molar fractional flows. x_1,x_2 are insitu mol fractions.

$$X_2=fx_2=1-X_1 \quad (1)$$

$$f=K/(1-x^*+x_2(K-1)) \quad (2)$$

$$dX_2/dx_2 = dX_1/dx_1 = X' = f^2(1-x^*)/K \quad (3)$$

q = hydrocarbon phase flow rate out of a gridblock, rb/d

$CFL = qX'\Delta t/S_oPV$

PV = gridblock pore volume, rb

S_o = hydrocarbon phase saturation = $1-S_w$, fraction

For a CFL of 1, impes stable step computed and used is

$$\Delta t = S_oPV/(qX') \quad (4)$$

K=1 is neutral - dispersion will be normal numerical dispersion.

K>1 increases dispersion above the numerical dispersion level, as was used by Koval (SPEJ June 1963 page 145) to represent viscous fingering in unfavorable mobility ratio miscible floods. He suggested a K value of (.78+.22V**1/4)**4 in his context of K usage, where V is viscosity ratio μ_{oil}/μ_{solv}.

K<1 controls or reduces numerical dispersion.

Compared to the cpu time of a run with x*=0, K=1, an Impes run cpu time will be 1/(K(1-x*)) and K/(1-x*) times larger for K<1 and K>1, respectively.

FCM can be used in tertiary recovery runs where solvent injection is preceded by a water-injection period. The subsequent miscible (solvent-injection) period then reflects tertiary recovery from watered-out regions.

Example 1D datafile fcm1.dat gives results for any x*, K values entered. It also gives results for either straight solvent injection or tertiary recovery by solvent injection after waterflooding. See the comments in the datafile. Solvent/oil mol fractions are reported in the Timestep Table (edit the outfile and search for "(STEP)"). Example datafile fcm2.dat is a small 3D problem.

Using spe79692.dat (Nc=7) as an example, the FCM datablock is

```
FCM
  OIL   ! mole fraction composition of original oil
    3.742000123E-01  9.329999983E-02  1.421000017E-01
    1.002000012E-01  1.994482428E-01  6.217515841E-02
    2.857658826E-02
  SOLVENT   ! mole fraction composition of injected solvent
   .7553 .1501 .0902 .0044 3*0
  SEPARATOR
    264.7  180   ! P psia  T deg F
     59.7  180
     14.7   60
```

For further discussion of spe79692.dat, see the **Miscible** page on our website. Plots showing FCM vs. non-FCM results for this case are given in spe79692.xls. While oil recovery is close, 3 phase effects in the non-FCM case significantly decrease injected solvent throughput, compared to the FCM case.

Note:

Analysis of the fluid phase envelope (P vs. $Z_{solvent}$) should always be performed to make sure that operating conditions substantially remain in the undersaturated region. The error resulting from an improper assumption of first contact miscibility may be small or not. The model should be used to determine the degree of error resulting from any operation in the 2-phase hydrocarbon region.

Discussion of compositional simulation of miscible floods

Many authors have published results showing that compositional simulation of miscible displacements gives oil recovery which continues to change with increasing grid refinement. In particular, Stalkup & Dean illustrate this and give references to many other such studies (SPE/DOE 20178 presented at the SPE/DOE Seventh Symposium on Enhanced Oil Recovery, Tulsa, 1990).

Datafile spe20178.dat has many datafiles showing the effect of increasing grid refinement on compositional simulated oil recovery for the Model 1 problem of SPE 20178 (oil/solvent viscosity ratio is about 45). Datafiles spe79692.dat and misc.dat show minor effects of grid refinement on recovery (adverse mobility ratio of about 10). We have no good explanation of this effect or of why it is significant in some cases and insignificant in other cases. Arguably, the effect may be greater for large adverse mobility ratio (say oil/solvent viscosity ratio of 40 or more as opposed to 10 or less) and when effects of gravity and heterogeneity do not dominate the character of unstable viscous fingering.

For the FCM spe79692.dat quarter 5-spot problem, results for 10x10x10 and 40x40x10 grids show small effect of refinement. Datafile misc.dat is a MCM miscible case and shows little difference between 10x10x10 and 40x40x10 quarter 5-spot results.

Equation-of-state pvt data (PVTEOS)

Secondary keywords – PR1977, SRK, CPT, MW, TC, TCF, PC, ZCRIT, AC, OMEGA, OMEGB, SHIFT, PCHOR, BIN, OMEGAS, OMEGBS, SHIFTS, BINS

The equation-of-state (eos), compositional pvt data are entered following the keyword PVTEOS as follows:

```
PVTEOS n (pcunit) (tcunit) (PR1977) (SRK)
temp   (tunit)    !   reservoir temperature, default deg F
CPT    MW   TC  PC  ZCRIT  AC   OMEGA   OMEGB   SHIFT  ! header line
cpti   mwi Tci Pci   Zci   aci  omegai  omegbi  shifti  ! nc lines in
                                                ! order i = 1,2,...,nc
```

where

n = compositional pvt type (= 1 if n is omitted)

pcunit = optional label for units of PC, ATM or BAR, default is psia

tcunit = optional label for units of TC, K or R, default degrees R

PR1977 = optional label indicating to use original PR eos

SRK = optional label indicating to use Soave-Redlich-Kwong EOS

temp = reservoir temperature

tunit = optional label for units of temp, K, R, or C, default degrees F

n_c = number of components, excluding water

and, for component i,

cpti = component name, 4 or fewer characters, starting with a letter

mw_i = molecular weight

T_{ci} = critical temperature, deg R

P_{ci} = critical pressure, psia

Z_{ci} = zcritical

ac_i = Acentric factor

$omega_i$ = cubic eos constant

$omegb_i$ = cubic eos constant

$shift_i$ = shift factor

Enter all PVTEOS data before entering any INITIAL data. The minimum number of components is two, excluding water. The Peng-Robinson equation of state[16] is default. If you want the Soave-Redlich-Kwong equation of state[17], then enter the keyword SRK on the PVTEOS dataline.

Enter one block of PVTEOS data for each compositional pvt type.

The header line symbols identify the columns of the subsequent data. The first header line symbol must be CPT; the remaining symbols may be in any order but must correspond to the columns of the subsequent data. If multiple compositional pvt types are entered, they must all use the same number and names of components.

Eligible header line column symbols additional to those above are:

TCF critical temperature in deg F (optional to TC)

OMEGAS omega for surface separation

OMEGBS omegb for surface separation

PCHOR parachor, used only in calculating interfacial tension

The PCHOR, OMEGA, OMEGB, OMEGAS, OMEGBS, SHIFT, and SHIFTS header symbols (columns) under PVTEOS are optional. The remaining header symbols (columns), CPT, MW, TC, PC, ZCRIT, and AC are mandatory.

The shift factors, entered under keyword SHIFT (and separation values SHIFTS), are defined by Jhaveri et al[9] and represent the volume translation published by Peneloux et al[10]. Whitson[11] and Soreide[12] describe methods and correlations to generate shift factor values at reservoir temperature.

Interfacial tension is calculated from the Macleod-Sugden equation[15],

$$\text{tension} = \{ \text{SUMi}(P_i(\rho_o * x_i - \rho_g * y_i)) \}^4$$

where tension is in dynes/cm, SUMi denotes summation over i=1,n_c, and

P_i = Parachor of component i

ρ_o = oil phase density, g-mol/cc

ρ_g = gas phase density, g-mol/cc

x_i = mol fraction of component i in the oil phase

y_i = mol fraction of component i in the gas phase

Note that if parachor values are not entered or are entered 0, all calculated interfacial tensions will be 0.

The eos binary coefficient data are optional and entered under PVTEOS data as follows:

```
BIN
  b12 b13 b14 b15 ..... b1n
      b23 b24 b25 ..... b2n
          b34 b35 ..... b3n
              ...
                    bn-1,n
```

The binary coefficient matrix (bij) is symmetric with 0 diagonal. The above enters only the upper triangular portion of the matrix, n_c-1 lines. The binary coefficients for surface separation, if different from the above, are entered in the same format using the keyword BINS.

Default PCHOR, SHIFT, and BIN values are 0.

Default BINS values are BIN values.

Default OMEGA, OMEGB values are eos theoretical values.

Default OMEGAS, OMEGBS values are OMEGA, OMEGB values.

If the OMEGA (OMEGB) values are entered negative, they are interpreted (in absolute value) as multiplicative factors on the equation-of-state theoretical OMEGA (OMEGB) values. This also applies for the surface OMEGAS, OMEGBS values if they are entered.

Example:

```
PVTEOS
266.          ! reservoir temperature, deg F
CPT    PC      TC     AC    MW  ZCRIT PCHOR  OMEGA OMEGB OMEGAS OMEGBS
C1   673.1   344.22 .008  16.04 .355  77.   .43933 .0777  .510  .082
C2-6 636.3   670.95 .1636 47.38 .335 159.6 .492   .081   .488  .085
C7+  268.43 1328.2  .578  228   .27  687   .436   .072   .435  .075
```

```
BIN
.00237 .13908
       .00301
BINS
.00237 .05096
       .00301
```

Some example files for PVTEOS are spe3.dat, spe5.dat, test3.dat, and test4.dat.

LBC viscosity coefficients (CVISO)

Oil and gas phase viscosities in compositional eos cases are calculated from the Lohrenz et al correlation[13]. That correlation involves five polynomial coefficients and gives their values. Ahmed[14] suggests changing those values to match viscosity data. Most pvt programs regress on viscosity data to determine the values. The five values, generally denoted a0-a4 in the literature, are entered using the keyword CVIS0 under PVTEOS,

```
CVIS0  a0 a1 a2 a3 a4
```

If not entered, the default values are those given by Lohrenz et al.

BIN and CVIS0 are subentries under PVTEOS, i.e. when multiple PVTEOS tables are input, the order of entry is:

```
PVTEOS 1
BIN
CVIS0
PVTEOS 2
BIN
CVIS0
...
```

Automatic conversion of EOS to K-value table (KVTABLE, NOKVTABLE)

This optional entry applies only in compositional cases. If

```
KVTABLE  ires    ! ires = Reservoir number
```

is entered in Initial Data, the program will internally calculate and store a table of K-values vs pressure for Reservoir ires. The table is then used in place of the EOS to obtain K-values. The (compositional) pvt type must be constant throughout Reservoir ires and initial composition cannot vary with depth.

The purpose of KVTABLE use is reduction of cpu time. This reduction will be greater as the number of components, n_c, increases. The only purpose of this feature is reduction of cpu time while maintaining the accuracy of the EOS description. If it is tried and does not do that, it should not be used.

This feature is nearly rigorous for compositional problems where depletion, with or without water injection and/or influx, is the dominant reservoir process. That is, calculated results with KVTABLE entered will closely approximate those without the entry. The difference in results

caused by KVTABLE entry when gas injection is present is problem-dependent. Datafile spe3.dat involves 9-component gas cycling at original pressure for the first 3650 days; results are essentially the same with and without KVTABLE entry. Cpu time is 33% less with KVTABLE.

KVTABLE should not be used in a Reservoir where initial composition varies with depth.

In 1988, Aaron Zick (then with Arco) suggested the following method for using KVTABLE to reduce cpu time without loss of accuracy in gas injection cases:

(a) Use the TRACER feature to calculate the fraction of injected gas vs time in all blocks.

(b) Then automatically switch each block from tabular K-value usage to the rigorous eos when its fraction of injected gas reaches (say) .05.

This method can be invoked for IMPES cases through a data line following the KVTABLE specification. Using the spe5.dat datafiles as an example, initial values of the two fractions of traced component C1 are (0,1) and injected gas C1 tracer fractions are (1,0).

```
TRACER
 2  C1     ! assign 2 tracer fractions to traced component C1
C     the program will initialize tracer fractions as (0,1) by
C     default so the TRACERF entry is not necessary.
KVTABLE
  1 C1 .05   ! switch to eos when tracer fraction 1 of cpt C1 reaches
             ! a value of .05
C     use the following Recurrent Data entry
WELLTRACER
    INJ   C1
       1  0         ! tracer fractions fj in injected C1
```

In the multiple reservoir case, the KVTABLE feature may be used, with or without the TRACER switch, in some reservoirs and not in others. Datafile test1.dat illustrates input data for this. The only purpose of the KVTABLE with TRACER is reduction of cpu time. If it does not reduce cpu time, as may be the case for small to moderate n_c values, then do not use it.

In some cases, the first m years of the n-year compositional problem may involve only depletion and water injection/influx. A simple way to reduce the cpu of the first m years without losing accuracy in the later years is as follows: Enter "KVTABLE ires" in Initial Data, then enter

```
NOKVTABLE ires
```

at the end of m years in Recurrent Data when gas injection begins.

Interfacial tension (TENSION)

In default mode, Sensor does not use interfacial tension. To activate the calculation and use of tension, enter in Initial Data:

```
TENSION tenref tenmax   ! dynes/cm
```

where

tenref = tension at which entered P_{cgo} values were measured
tenmax= maximum tension, optional, default = tenref

Table values of gas-oil capillary pressures P_{cgo} are multiplied by tension/tenref. The purpose of tenmax is avoidance of an extreme increase in P_{cgo} which can occur as tension increases due to pressure decline or stripping of oil by injected gas. Entry of tenref = 0 or of tenmax/tenref > 10 is flagged as a fatal error.

Effect of interfacial tension on relative permeability (KRIFT)

The Initial Data entry

```
KRIFT
```

activates the effect of interfacial tension (IFT) on oil and gas rel perms kro,krg. There is no effect in any block where IFT is above the cutoff or threshold IFT value. The default value of cutoff IFT is 1/5 of initial average tension. Initial average tension is the hydrocarbon volume average value of IFT at time 0. You can override this cutoff value by entering "KRIFT tencut" where tencut, dynes/cm, is the cutoff value you desire.

If IFT is below the cutoff value, kro and krg are interpolated between values at 0 tension and normal immiscible values at cutoff tension. At 0 tension, kro and krg are straight-line functions of So or Sg, with 0 endpoint saturations. The value of kro+krg at 0 tension is .5(krow(Sw)+krg(Sg=1-Sw)), calculated using endpoint saturation = .5(Sorw+Sgr).

KRIFT usage is primarily intended for miscible flooding where the first contact miscibility assumption is not warranted - i.e. where there are extensive or significant three-phase (saturated) space-time regions, possibly including blowdown. For miscible displacements, Pcgo effects should arguably be ignored. If KRIFT is entered, Pcgo values are ignored (set to 0).

Edit your outfile and search for KRIFT to see printout of IFT-related data.

See example datafiles spe79692 amd misc.dat, note the TITLE comments.

Surface separator data (SEP)

Surface separator data are necessary only for compositional problems. The program uses separator #1 to calculate initial gas and oil in place. If you do not enter SEP in Initial Data, the program by default sets one single-stage separator with p = 14.7 psia and T = 60 deg F.

If the BLACKOIL option is used to convert the compositional EOS description for a pvt type to black oil, then these SEP data are ignored, in favor of the SEPARATOR data specified under the BLACKOIL keyword applying to the pvt type.

Separator data can be changed or added in Recurrent Data using the keyword SEP. If you enter no WELLSEP data assigning separators to wells, then the program by default uses separator # 1 for all wells.

In the multiple pvt case, the model needs to know which separator to use in flashing each fluid pvt type to determine in-place surface oil and gas. By default, all pvt types use separator 1. You can change this by using as many of the following Initial Data entries as desired:

```
ASSIGN  SEP i  PVT j
```

This entry assigns separator i to pvt type j. A well uses a separator determined by the pvt type of the well's first perf (perfs are entered under the keyword WELL of Recurrent Data). For

example, if a well's first perf is pvt type 3 and pvt type 3 uses separator 4, then the well uses separator 4. You can change the separator used for the well by subsequent entry of the Recurrent Data WELLSEP keyword. To see the separator assignments for the wells, edit your outfile and search for the string "WELL INPUT".

A separator may be entered as a set of separator stage conditions to be used in a multistage flash, or as a table of liquid recovery factors, referred to as a separator table. In both cases, the feed to the separator is the wellstream.

Multistage flash separators are specified as in the following example.

Example:

```
SEP 1                !    separator 1
      1014.7 150.0   !    p (psia) and T (deg F) of each stage
       264.7  80.0
        14.7  60.0
SEP  2               !   separator 2
       814.7  130    !   p and T of each stage
        14.7   60
```

The feed to the first stage is the wellstream. The feed to each subsequent stage is the liquid from the previous stage.

A separator table is specified as

```
SEP  itab   izc  TABLE ! itab = separator table #
zc   f1  f2  … fnc-1 ! correlation zc and nc-1 liquid recovery factors
zc   f1  f2  … fnc-1
…
```

where zc is the correlation mole fraction, defined as the sum of component number izc and higher mole fractions in the feed. Each of the liquid recovery factors f(i) is the molar fraction of component i in the feed recovered as liquid. Given 1 mole of feed of compositon z(i), the liquid contains f(i)*z(i) moles of component i and the vapor contains (1-f(i))*z(i) moles of component i.

Example:

```
SEP 1 5  TABLE    ! separator 1   cpt 5
C    C7+    C1N2   C2CO    C3-4    C5-6     HF1    HF2
  0.0000   0.011  0.055   0.151   0.362    0.715  1.000
  0.020    0.011  0.055   0.151   0.362    0.715  1.000
  0.0261   0.023  0.079   0.177   0.391    0.776  1.000
  0.0453   0.035  0.127   0.284   0.496    0.843  1.000
  0.0710   0.051  0.173   0.362   0.619    0.872  1.000
  0.0911   0.066  0.217   0.427   0.661    0.889  1.000
  0.1096   0.073  0.235   0.451   0.693    0.911  1.000
  1.0000   0.073  0.235   0.451   0.693    0.911  1.000
```

The first column is the correlation mol fraction, equal to the sum of component 5 and higher numbered (as ordered in the PVTEOS data, usually lightest to heaviest) component mol fractions in the feed.

An example datafile is septable.dat.

Relative permeability and capillary pressure (SWT, SGT, SLT, SGTR, SGWT)

Relative permeability and capillary pressure data may be entered as tables or as analytical functions. The normal tabular data are described here; analytical function input are described in a following section. The term Saturation Table refers to the pair of one water-oil table, with data for $S_g = 0$, and one gas-oil table, with data for $S_w = S_{wc}$. Each grid block is assigned to one of these tables through the array ROCKTYPE. If you have multiple (say, n) Saturation Tables, number them 1,2,3,...,n. If there is only one Saturation Table (rocktype) then the program defaults rocktype to 1 for all blocks and the ROCKTYPE array need not be entered.

Relative permeability to water, $k_{rw}(S_w)$, is normally a single-valued function of water saturation. Optionally, it may be treated as a function of both S_w and S_g using Baker's saturation-weighted linear interpolation method (SWLI)[18]. Relative permeability to gas, $k_{rg}(S_g)$, is a single-valued function of gas saturation.

Sensor calculates three-phase oil relative permeability using Stone's first method[19], as extended by Fayers[20] to treat minimum or residual oil saturation as a function of S_g. Optionally, Stone's second method[21] or Baker's SWLI method may be used. Appendix 4 describes the Stone and SWLI methods. To use Stone's second method, enter STONE 2 in Initial Data. To use the SWLI method for k_{ro}, enter

```
KROINT
i     1    ! use SWLI for kro in Saturation Table i
j - k  1   ! use SWLI for kro in Saturation Tables j-k, inclusive
m     0    ! do not use SWLI for kro in Saturation Table m
```

... any number of lines ...

The default value of the k_{roint} integer (1 or 0 in the above lines) is 0 for all Saturation Tables.

Entered capillary pressure data may be zeroed or altered as described in a following section. The entered capillary pressure curves may be augmented by a vertical equilibrium component, also described in a following section.

The two-phase water-oil saturation table is entered using the keyword SWT. The two-phase gas-oil saturation table is entered using the keyword SGT or SLT. You may enter as many Saturation Tables (rocktypes) as desired. The order of the tables is immaterial - e.g. you may enter all SWT tables followed by all SGT tables or enter pairs of SWT and SGT tables.

```
SWT n                            !  water-oil data, Table n
  Sw krw  krow  (Pcwo) (Pcwoi)      !  any number of lines

SGT n                               ! gas-oil data, Table n
  SGTR Sgr    ! maximum residual (trapped) gas saturation  optional
  Sg  krg  krog  (Pcgo)             !  any number of lines
```

or

```
SLT n                               ! gas-oil data, Table n
```

```
SGTR Sgr    ! maximum residual (trapped) gas saturation  optional
Sliq  krg  krog  (Pcgo)           ! any number of lines
```

Notes:

1. The S_w entries under SWT must increase monotonically from S_{wc} to 1.0.
2. The S_g entries under SGT must increase monotonically from 0 to 1-S_{wc}.
3. The S_{liq} entries under SLT are S_{wc}+S_o = 1- S_g and must increase monotonically from S_{wc} to 1.0.
4. Either SGT or SLT may be used to enter gas-oil data.
5. All saturations and relative permeabilities are fractions.
6. Capillary pressures are in units of psi.
7. All capillary pressure columns are optional
8. Pcwoi may not be input unless Pcwo is input.
9. If input, the values of P_{cwo} and P_{cwoi} must be equal at $S_w = S_{wc}$.

Krg hysteresis

If S_{gr} is entered > S_{gc}, then k_{rg} is calculated with hysteresis as described in Appendix 5. The method closely resembles that proposed by Land[22] and Carlson[23]. If S_{gr} is entered less than or equal to S_{gc} (the default) then no k_{rg} hysteresis occurs.

SWLI method for Krw

To use the SWLI method for k_{rw}, enter a third, two-column SGWT table as

```
SGWT n     !  Saturation Table n (Rocktype n)
Sw  krwg   ! as many entries as desired
```

The S_w entries must increase monotonically from S_{wcg} to 1, where S_{wcg} is "connate" or residual water saturation to gas in the gas-water system. Oil has two residual saturations, S_{orw} in a two-phase water-oil system (S_g=0) and S_{org} in a two-phase gas-oil system (S_w =S_{wc}). Similarly in the SWLI case, water may have two different residual saturations, S_{wc} in a two-phase water oil system (S_g=0) and S_{wcg} in a two-phase gas-water system (S_o =0).

The k_{rwg} values are water relative permeability in a gas-water system with S_o =0. The k_{rwg} values must increase monotonically from 0 to k_{rw}(S_w =1.0), which must equal the last k_{rw} value entered under SWT. You may enter the two column table using S_g in place of S_w. In that case, the S_g values must increase monotonically from 0 to 1-S_{wcg} and the k_{rwg} values must decrease from k_{rw} (S_w =1.0) to 0.

Black oil gas-water problems

In black oil gas-water problems you still need to enter the normal water-oil SWT Table and gas-oil SLT or SGT Table for relative permeabilities and capillary pressures. Those tables need to include krow in the former Table and krog in the latter Table but the krow and krog values are not used. Gas-water Pcgw will be the sum of whatever PCWO and PCGO values you may enter in the two Tables.

Gridblock assignments

Each grid block is assigned to one of the Tables by the integer array ROCKTYPE. If the ROCKTYPE array is not entered, all grid blocks are assigned to Saturation Table 1.

Endpoint saturations

The entered tabular data define the values of endpoint saturations S_{wc}, S_{orw}, S_{org}, S_{gc} and S_{gr}. If SGTR is not entered, the table value of S_{gr} is set equal to S_{gc}. Each grid block is assigned values of endpoint saturations which equal those of the Saturation Table assigned to the block, unless overridden by entry of the arrays SWC, SORW, SORG, SGC, SWCG, and SGR described in Section 1.5.

The entered Saturation Tables are normalized in saturations as described in Appendix 6. The k_r and P_c for each grid block are calculated by denormalization using the grid block endpoint saturation values.

Krg concavity

The first nonzero k_{rg} entry in the SGT Saturation Table arguably should exhibit concave upward behavior vs S_g with respect to its two neighbor k_{rg} values. Highly concave downward behavior in this regard can cause significant computational problems. The Initial Data entry

```
CONCAVE n
```

where

n = 0 If k_{rg} vs S_g at S_{gc} is concave downward, k_{rg} is linearized

n = 1 k_{rg} is unchanged by the model

n = 2 same as n=1 except printout notes the presence or absence of any downward concave k_{rg} (edit and search for "CONCAVE").

n = 1 is default.

is provided in this regard. For example, if entered with n=0, the data

```
C              Sg    krg
               0.     0.
              .05     0.
              .1     .07
              .15    .09
              .25    .16
               .......
```

would be internally changed to

```
C              Sg    krg
               0.     0.
              .05     0.
              .1     .045
              .15    .09
              .25    .16
               .......
```

In some such cases, this entry reduces oscillation and instability with little effect on results.

Output tables

The printed Saturation Tables include the derivatives $d(k_{rw})/d\ S_w$, $d(k_{row})/d\ S_w$, $d(k_{rg})/dS_g$, etc. You can edit your outfile and search downward from the top for "TY DA" to view these derivatives. Highly erratic behavior of these derivatives may contribute to computational difficulty and should be avoided.

Table respacing (NKR)

Highly erratic tabular relative permeability and/or capillary pressure tabular data can cause computational difficulty such as oscillations and failure to converge. The step-function nature of the spe9.dat k_{row} and P_{cwo} data is an example of such data. In these cases the Initial Data entry

```
NKR n
```

may be helpful. The entry equally spaces the entered tabular k_r and P_c data using n equally spaced saturation increments. The entered P_c curves are used unchanged in initialization (equilibration). Thus if NKR n is entered and P_c affects any initial saturations, the NKR entry may result in some minor movement(change) at zero rate. However, it will not affect calculated initial fluid-in-place values. The NKR entry also reduces cpu time slightly since table look-up is much faster with equally spaced saturations.

Example:

```
SWT 1    ! Saturation Table 1  (Rocktype 1)
C  Sw        krw     krow   !  Pcwo   Pcwoi (optional)
  0.15000    0.000   1.000  !   0.0    0.0
  0.18186    0.001   0.958  !   0.0    0.0
   .....
  0.42842    0.305   0.318  !   0.0    0.0
  0.48000    0.390   0.000  !   0.0    0.0
  1.00000    0.700   0.000  !   0.0    0.0

SGT 1    ! Saturation Table 1  (Rocktype 1)
C    Sg        krg    krog   ! Pcgo
  0.00000     0.000   1.000  ! 0.0
  0.02000     0.050   0.950  ! 0.0
  0.05000     0.100   0.780  ! 0.0
   .....
  0.45000     0.630   0.050  ! 0.0
  0.50000     0.650   0.020  ! 0.0
  0.57000     0.670   0.000  ! 0.0
  0.85000     1.000   0.000  ! 0.0
```

If P_{cwo} (drainage) and P_{cwoi} (imbibition) water-oil capillary pressure curves are both zero, you may omit columns 4 and 5 in the SWT table. If you enter P_{cwo} as column 4 and omit column 5 then the program internally sets P_{cwoi} equal to the entered P_{cwo}. This internal set is done on a line-by-line basis; for example:

```
SWT 4
```

```
.2    0.    1.    44.6           ! pcwoi is set equal to 44.6
.3    .01   .7    17.5           ! pcwoi is set equal to 17.5
.5    .11   .18    6.1    1.1    ! pcwoi is 1.1
.7    .27    0.    2.2    0.     ! pcwoi is 0.
  ..........
```

Leverett J function (JREF , JLEVMAX)

In default mode, Sensor uses P_{cwo} and P_{cgo} from the entered Saturation Tables, with appropriate normalization described in Appendix 6. The J-function option applies Leverett's observation of linear dependence of the magnitude of capillary pressure with the square root of the ratio of permeability to porosity. The JREF keyword specifies the reference value of sqrt(k/ϕ) corresponding to the entered capillary pressures in the Saturation Tables. JREF can either be entered in Initial Data as a single global value,

```
JREF  v (nopt)
```

or it can be entered as a grid block array as described in Section 1.5.

If JREF is entered, then the program multiplies all P_{cwo},P_{cgo} obtained from the Tables by jlev = v/J where J is a grid block array equal to sqrt(k/ϕ), and v is the gridblock reference value, which defaults to the global reference value if the JREF array is not entered. By default the program limits the value of the scaling factor jlev to a maximum of 1000 (versions previous to May 20, 2008 used a default maximum of 10000). If you desire a different maximum, then enter

```
JLEVMAX jlevmax
```

The k and ϕ are grid block permeability (md) and porosity (fraction) and v is the reference value of sqrt(k/ϕ). Grid block permeability k is sqrt($k_x k_y$). If you want another definition of k, include an optional specification of nopt following the JREF array input option label as described in Section 1.5, or as a second number on the global JREF entry:

```
JREF v  1   ! use k = kx
       or
JREF v  2   ! use k = .5*(kx + ky)
       or
JREF v  3   ! use k = Maximum(kx,ky,kz)
```

Vertical equilibrium (VE, DIP)

```
VE (DIP)
```

The vertical equilibrium option is activated by entry of VE, VE DIP, or the THVE array, or by use of the DUAL option. VE and VE dip should be used only for unfaulted Cartesian grids. The THVE array should always be input for faulted or non-Cartesian grids when the vertical equilibrium option is active.

If VE is input, Sensor uses Quandelle's approximate ve P_c curve, defined as the sum of P_{crock} and a straight line,

$$P_c = P_{crock} + \alpha * (1 - 2 * S_n)$$

where α is .5*h*Δγ, h is gross block thickness, and

	water-oil P_c	gas-oil P_c
$\Delta\gamma$ =	$\gamma_w - \gamma_o$	$\gamma_o - \gamma_g$
S_n =	$(S_w-S_{wc})/(1-S_{wc})$	$(1- S_g-S_{wc})/(1-S_{wc})$

The proper value of h to use in the above equations is the vertical span of the block, or the VE thickness THVE, which is given by zmax – zmin, where zmax and zmin are the maximum and minimum depths of any point contained in the block. Entry of VE DIP for Cartesian grids causes the program to internally augment the gross block thickness by dip angle to compute THVE, the ve thickness for each block (in this case, geometrical information such as depths and block dimensions should not be omitted (i.e., specified as 0) for inactive cells, since these are needed by the program to compute dip angles). If a gridblock with Δx = 500 ft and Δz = 20 ft dips at 5 degrees in the x direction, the difference between uppermost and lowermost points of the block is about 60 ft and use of this value in place of 20 ft is somewhat more accurate.

Entry of the THVE array overrides any internally calculated value. For grids other than Cartesian (corner point, unstructured, …), the THVE array should always be computed by the gridding program and specified directly to Sensor.

The γ_w, γ_o, and γ_g are the water, oil, and gas phase gradients, psi/ft. This altered P_c can give more accurate values of initial fluids in place if a gas-oil and/or water-oil contact exists. If rock P_c is large and highly nonlinear, rock P_c may be more accurate than this approximate ve P_c.

Analytical relative permeability (k_r) and capillary pressure (P_c)(KRANALYTICAL)

Analytical k_r and P_c data are often used in research problems or for problems where there is large scatter and uncertainty in relative permeability data. The use of analytical expressions makes it easy to change k_r data:

a) from run to run in history matching.

b) for the purpose of sensitivity runs

We recommend use of the KRANALYTICAL Sensor option vs entry of Tables (SWT,SGT,SLT) for all real cases. Lab-determined kr vs S plots are often scattergrams for rock samples of the same rocktype and perm and porosity levels. Samples of the same type just a short distance apart can give significantly different lab kr curves.

The most important properties of kr curves are their endpoint saturations and curvature. KRANALYTICAL allows representation of these properties using the minimum possible number of variables.

Sensor analytical kr curves are used directly in the simulation. That is, the analytical expressions are indeed used to print out a Table in the output file, but Sensor does not use interpolation in that Table – it uses the analytical expressions themselves. For example consider the analytical $krg=Sg^2$. In a Table of .05 Sg increments, at Sg=.01 interpolation gives krg=0.0005, while use of the analytical expression itself gives krg=0.0001, five-fold less. This can be important in stable-step logic and may or may not have some effect on the accuracy of simulator results.

To use analytical relative permeability data, enter:

```
KRANALYTICAL n     ! n = table #
  Swc Sorw Sorg Sgc (Sgr)
  krwro  krgro  krocw
  nw now ng nog (nwg Swcg)
```

where

S_{wc} = connate water saturation, fraction
S_{orw} = residual oil saturation to water
S_{org} = residual oil saturation to gas
S_{gc} = critical gas saturation
S_{gr} = trapped gas saturation, optional, default=S_{gc}
k_{rwro} = relative permeability of water at S_w=1-S_{orw}, S_g=0
k_{rgro} = relative permeability of gas at S_w=S_{wc}, S_o=S_{org}
k_{rocw} = relative permeability of oil at S_w=S_{wc}, S_g=0
n_w,n_{ow},n_g,n_{og} = exponents for analytical k_r

The relative permeabilities are calculated as

$$k_{rw} = k_{rwro} * [(S_w - S_{wc})/(1 - S_{orw} - S_{wc})]^{nw}$$
$$k_{row} = k_{rocw} * [(1 - S_{orw} - S_w)/(1- S_{orw} - S_{wc})]^{now}$$
$$k_{rog} = k_{rocw} * [(1 - S_{org} - S_{wc} - S_g)/(1 - S_{org} - S_{wc})]^{nog}$$
$$k_{rg} = k_{rgro} * [(S_g - S_{gc})/(1 - S_{org} - S_{wc} - S_{gc})]^{ng}$$

The calculated k_{rw} and k_{rg} values are limited by 1.0 and k_{rwro} and k_{rgro} must be 1.0 or less.

The pivot points for the water and gas curves as shown above are krw=krwro at Sw=1-Sorw and krg=krgro at Sg=1-Swc-Sorg, respectively. If you would rather enter krw at Sw=1 as the pivot point for the water curve, then enter krwro negative as -v and the program will set krw at Sw=1 to v and will calculate krw as

krw = v * ((Sw-Swc)/(1-Swc))**nw

If you would rather enter krg at Sg=1-Swc as the gas pivot point, then enter krgro negative as -v and the program will set krgcw=v and will calculate krg as

krg = v * ((Sg-Sgc)/(1-Swc-Sgc))**ng

For a water-dry gas case when no oil saturation ever occurs, it is permissible to set krocw to zero along with Sorw=Sorg=0.

To use the SWLI method or Stone's Method 2 for three-phase k_{ro}, see the keywords KROINT and STONE 2 discussed in the preceding 'Relative Permeability and Capillary Pressure' section.

To use the SWLI method for k_{rw} (discussed in Appendix 4), add n_{wg} and S_{wcg} values as fifth and sixth numbers on the "nw now ng nog" dataline. The k_{rwg} value for the SWLI method will then be calculated as

$$k_{rwg} = k_{rw} (S_w=1.0) * ((S_w-S_{wcg})/(1-S_{wcg}))^{nwg}$$

If only the above three lines of data following KRANALYTICAL are entered then all capillary pressure curves are 0. If you desire nonzero P_c data then enter any one, any two, or all three of the additional three lines of data:

```
a1 a2 a3        PCWO
```

```
b1 b2 b3 b4 b5  PCWOI
c1 c2 c3        PCGO
```

If the PCWO dataline is entered but the PCWOI dataline is not entered, then the program internally sets imbibition P_{cwoi} to drainage P_{cwo}.

The P_c are calculated as

$$P_{cwo} = a1 + a2 * (1-S_{wn})^{a3}$$

$$P_{cwoi} = b1 + b2 * (1-S_{wn})^{b3} - b4 * S_{wn}^{b5}$$

$$P_{cgo} = c1 + c2 * S_{gn}^{c3}$$

where

P_{cwo} = drainage water-oil P_c

P_{cwoi} = imbibition water-oil P_c

P_{cgo} = drainage gas-oil P_c

$S_{wn} = (S_w-S_{wc})/(1-S_{wc})$

$S_{gn} = S_g/(1-S_{wc})$

Notes:

1. All endpoint saturations S_{wc}, S_{orw}, S_{org}, and S_{gc} must lie between 0 and 1.0.
2. Each of $S_{wc} + S_{orw}$ and $S_{wc} + S_{org} + S_{gc}$ should be significantly less than 1.0.
3. All the exponents n_w, n_{ow}, n_g, and n_{og} shoud be 1.0 or larger.
4. The k_{rocw} value must be positive.
5. The k_{rwro} and k_{rgro} values must be < 1.0
6. P_{cwo} and P_{cwoi} values must be equal at $S_w = S_{wc}$.
7. dP_{cwo}/dS_w must be negative and dP_{cgo}/dS_g must be positive.

Black oil gas-water problems

In black oil gas-water problems, you can enter krocw as 0 and enter now and nog as 1. The gas-water capillary pressure will be the sum of whatever you enter for PCWO and PCGO.

Examples:

This enters k_r data only

All P_c curves are 0

```
KRANALYTICAL 1
.2   .3   .18  .02
.27  .31  .92
2.2  2.5  3. 1.7
```

This enters k_r and P_{cwo} data

```
KRANALYTICAL 1
.2   .3   .18  .02
.27  .31  .92
2.2  2.5  3.  1.7
0.  35.  8.    PCWO
```

All endpoint saturations S_{wc}, S_{orw}, S_{org}, and S_{gc} must lie between 0 and 1.0. Each of $S_{wc} + S_{orw}$ and $S_{wc} + S_{org} + S_{gc}$ should be significantly less than 1.0. All the exponents n_w, n_{ow}, n_g, and n_{og}

shoud be 1.0 or larger. The k_{rocw} value must be positive. The k_{rwro} and k_{rgro} values must be < 1. P_{cwo} and P_{cwoi} values must be equal at $S_w = S_{wc}$. dP_{cwo}/dS_w must be negative and dP_{cgo}/dS_g must be positive.

Capillary pressure alteration (PCMULT1, PCMULT2)

Entered capillary pressure data, P_{cwo} and P_{cgo}, can be altered

1) before use in initialization, and/or
2) after use in initialization

The entry

```
PCMULT1  n  fw1 fg1
```

multiplies entered P_{cwo} and P_{cwoi} data by fw1 and P_{cgo} data by fg1 for Saturation Table n. The Table number n may be entered as a table range, n1 - n2. If n is omitted, the factors are applied to all Saturation Tables. This entry is convenient in rare cases when extensive Saturation Table data are entered and one wishes to (say) zero all P_c data without the burden of retyping all the tables.

The entry

```
PCMULT2 n fw2 fg2
```

applies multiplicative factors fw2,fg2 to the entered capillary presssures after they are used in initialization. This gives a difference in P_c used in initialization and P_c used in the run. Some movement (at zero well rates) will result. This entry using factors less than 1.0 may help in Impes problems where severe P_c instability occurs.

Relative permeability limits (KRWLIMIT, KROLIMIT, KRGLIMIT)

Relative permeabilities may be > 1.0 if desired. The program imposes maximum limits of krwlimit, krolimit, and krglimit on the krw, kro, and krg, respectively, used in the simulation in well terms and interblock flow terms. Default values of these limits are 1.0. The limits may be changed by the optional Initial Data entries:

KRWLIMIT krwlimit

KROLIMIT krolimit

KRGLIMIT krglimit

Logic of relative permeability endpoint scaling and limits

Tabular or analytic relative permeabilities are entered in Initial Data by rocktype as functions (f,g,h) of saturations,

krw=f(Sw) kro=g(krow(Sw),krog(Sg)) krg=h(Sg)

Let

krwro = krw at Sw=1-Sorw and Sg=0

krocw = kro at Sw=Swc and Sg=0

krgro = krg at Sg=1-Swc-Sorg

be the endpoint kr's determined/specified by those functions. Let krwro*, krocw*, and krgro* be the optionally entered endpoint saturation arrays. If krwro* are not entered, then krwro*=krwro, and similarly for krocw* and krgro*. Let

α = krwro*/krwro β= krocw*/krocw γ = krgro*/krgro

Then the rel perms used in the simulation are

krw=αf(Sw)

kro = g(βkrow(Sw),βkrog(Sg))

krg = γh(Sg)

where βkrow(Sw) and βkrog(Sg) are limited by krolimit and each of krw, kro, krg are limited by krwlimit, krolimit, and krglimit, respectively.

Compaction tables (COMPACTABLE)

This keyword is used to enter tabular data for compaction with hysteresis and, optionally, with water weakening[24]. Enter as many tables as desired under the single entry COMPACTABLE. The tables can enter either compressibility as c_f*10^6 or porosity ϕ as a fraction or per cent. The definition of c_f is

$$c_f = -\frac{1}{\phi}\frac{d\phi}{ds} \quad 1/psi$$

where stress s = p_{ob} - p, ϕ is a fraction, and overburden pressure p_{ob} = a + b*depth. If c_f values are entered, Sensor uses them to compute table porosity values. The table porosity values are functions of the three parameters: stress s, Initial Porosity ϕ_0, and a water saturation parameter s_w, defined below.

The grid block porosity is calculated using bilinear interpolation in the table of ϕ values. At time 0, the value of ϕ_0 for each block is calculated so that each interpolated block ϕ value equals its initial (data) value. This ϕ_0 value for a block will differ slightly from the block's (data) initial porosity.

Sensor modifies the normal bilinear interpolation to increase accuracy and avoid derivative discontinuities. To illustrate, let stress s lie between table stress entries s_1 and s_2, with ϕ_1 and ϕ_2 denoting table ϕ values at s_1 and s_2, respectively. Linear interpolation gives

$$\phi = \phi_1 + (\phi_2-\phi_1)(s-s_1)/(s_2-s_1) \qquad (1)$$

The derivative of this interpolated f value, $d\phi/ds$ or $d\phi/dp$, exhibits step discontinuities at $s=s_1$ and $s=s_2$. A more accurate interpolated ϕ is given by

$$-\ln(\phi/\phi_1) = I(s_1,s)\,(c_f ds) = .5(c_{f1}+c_f)(s-s_1) \qquad (2)$$

where $I(s_1,s)$ denotes the integral from s_1 to s, c_{f1} is table compressibility at stress s_1, and c_f is linearly interpolated at s between s1 and s_2. The ϕ from Eq (1) varies linearly with stress while the ϕ from Eq (2) does not. The derivative $d\phi/ds$ using Eq (2) obeys $d\phi/ds = -c_f\phi$ and is continuous at all stress values. In short, Sensor uses linear interpolation in c_f, not in ϕ.

Extrapolation

Let (s_1,s_n), (s_{w1},s_{wn}), (ϕ_{01},ϕ_{0n}) denote the smallest and largest of the entered table parameters stress, s_w, and Initial Porosity, respectively. If $s_w < s_{w1}$, ϕ is calculated using table data at $s_w=s_{w1}$. If $s_w > s_{wn}$, ϕ is calculated using table data at $s_w=s_{wn}$. If $\phi_0 < \phi_{01}$, ϕ is calculated using table data at $\phi_0=\phi_{01}$. If $\phi_0 > \phi_{0n}$, ϕ is calculated using table data at $\phi_0=\phi_{0n}$. If stress $s < s_1$, ϕ is calculated from $\ln(\phi/\phi_1)=-c_{f1}(s-s_1)$, where ϕ_1 and c_{f1} are values at $s=s_1$. If $s > s_n$, ϕ is calculated from $\ln(\phi/\phi_n)=-c_{fn}(s-s_n)$, where ϕ_n and c_{fn} are values at $s=s_n$.

Hysteresis

If block pressure is above historical minimum pressure pmin, then porosity is calculated as

$$\phi = \phi_{min} (1 + c_{f0}(p-p_{min}))$$

where ϕ_{min} is porosity calculated from the table at p_{min} (and s_w, if applicable) and c_{f0} is compressibility calculated from the table at the first (lowest) stress value s_1 and at the first (lowest) s_w value, s_{w1}.

Example:

Two Tables, no water weakening (enter nsw=0).

```
COMPACTABLE
  1 10  8  0     ! Table 1   ntable nstress nphi nsw
  9000  0        !  a b
  0 1500 3000 3500 4500 5000 5500 6500 7500 8000 ! nstress si values
 .25 .27 .29 .31 .33 .35 .37 .39                 ! nphi ϕ0 values

   2.63 2.80 2.60 2.17 2.20 2.23 3.03 3.70 ! nphi cf values at s1
   2.63 2.85 2.73 2.39 2.50 2.61 4.47 7.24 ! nphi cf values at s2
   2.64 2.86 2.75 2.63 3.06 4.80 7.85 13.3
   2.64 2.88 2.84 3.12 4.60 8.31 13.4 21.4
   2.65 2.97 3.48 5.11 8.29 13.1 22.2 40.5
   2.65 3.06 4.11 6.93 11.5 18.9 32.2 52.7
      2.65 3.23 4.58 8.31 15.3 27.2 43.4 60.9
      2.65 3.22 4.99 10.5 21.5 39.1 61.4 82.8
      2.66 3.43 6.09 13.7 28.2 49.4 72.3 86.8
      2.66 3.52 7.16 18.0 36.5 56.3 68.9 73.3  ! nphi cf values at sn

        2 10  8 0    ! Table 2
          9000  0
          0 1500 3000 3500 4500 5000 5500 6500 7500 8000
         .25 .27 .29 .31 .33 .35 .37 .39

            1.79 2.89 4.13 4.69 4.76 4.80 5.24 5.93
            1.79 2.89 4.14 4.69 4.77 4.80 5.24 6.34
            1.79 2.90 4.16 4.78 6.01 12.0 31.0 66.3
```

```
1.80 2.90 4.16 4.94 7.31 18.2 46.0 86.1
1.80 2.91 4.37 8.74 24.9 57.7 93.7 114.0
1.80 2.91 5.69 16.6 43.0 78.0 100. 107.0
1.80 3.08 7.24 20.9 47.5 74.3 86.9 89.6
1.84 5.09 16.5 38.1 59.2 69.0 71.0 71.0
2.53 9.07 24.4 43.9 55.8 59.6 59.8 59.8
3.63 12.4 29.0 43.9 50.4 51.6 51.6 51.6
```

ntable table number

nstress number of stress values

nphi number of Initial Porosities

nsw number of s_w parameter values

s stress values

ϕ_0 Initial Porosities, fraction or per cent

s_w s_w parameters, fraction or per cent

c_f pore-volume compressibility * 10^6, 1/psi

p_{ob} = a + b * depth, overburden pressure, psia

stress s = p_{ob} - p, psi

Initial Porosities, stress, and s_w do not have to be equally spaced, but must be entered in monotonically increasing order.

Porosities (fraction or per cent) may be entered in place of compressibilities. However, experience shows that the c_f values derived internally from entered porosities can be highly erratic and can cause run failure. You should view the printed compaction tables to check against erratic c_f. Edit your outfile and search from the top down for "COMPAC".

If you enter porosity rather than compressibility, then append "PHI" on the COMPACTABLE line, i.e. "COMPACTABLE PHI". If there are no water weakening data, enter nsw as 0.

If compaction tables are entered, the integer array keyword COMPACTYPE (Section 1.5) must be entered to assign grid blocks to appropriate compaction tables. Grid blocks not assigned to any compaction table have porosities calculated from $\phi = \phi_0 e^{(cf\ (p-p0))}$ where ϕ_0 is initial porosity (the value entered as data), p_0 is initial block pressure, and c_f is entered on the MISC line.

Water Weakening

See files test9.dat and test9.out for an example of compaction with water weakening. Full water weakening occurs at a water saturation less than 1-S_{orw}. The last (largest) table s_w value, s_{wn}, is this maximum water saturation, denoted below as swmax. For an unfractured reservoir "NOFRAC" must appear on the COMPACTABLE dataline and the s_w value is taken as grid block water saturation,

$$s_w = S_w \qquad (1)$$

In a naturally fractured reservoir, full water weakening will not occur until the water front passes completely through the block. To account for the fact that only a portion of the block is weakened as water flows through the block, the following s_w value is used in the table lookup:

$$s_w = S_{wi} + (swmax - S_{wi})(S_w - S_{wi})/(1 - S_{orw} - S_{wi}) \quad (2)$$

where

S_{wi} = initial water saturation in the grid block

S_w = current water saturation in the grid block

A historical maximum water saturation, S_{wmax}, is carried for each grid block. If S_w is less than S_{wmax} then S_{wmax} is used in place of S_w in the above equation. If $swmax-S_{wi} < 0$ or $1-S_{orw}-S_{wi} < 0$ then s_w=swmax is used in the table lookup. No extrapolation in s_w is used. If $s_w < s_{w1}$, s_{w1} is used in the table lookup. If $s_w >$ swmax, swmax is used in the table lookup.

Eqn (1) is used with all tables entered under a COMPACTABLE NOFRAC entry. Eqn (2) is used with all tables listed under a COMPACTABLE entry. In either case, PHI must appear on the COMPACTABLE line if porosities are entered.

Transmissibility stress dependence tables (TMODTABLE)

Transmissibilities may optionally be treated as a function of stress, where stress $s = p_{ob} - p$, overburden pressure $p_{ob} = a + b*z$, and p and z are gridblock pressure and depth. Each TMODTABLE table number n specifies a and b and gives x, y, and z-direction transmissibility multipliers (txmod, tymod, tzmod) as a function of stress. Gridblocks are assigned to TMODTABLE tables using the TMODTYPE array. Stress s must increase monotonically in the table:

```
  TMODTABLE n
  a  b
C STRESS TXMOD TYMOD TZMOD
     s   txmod tymod tzmod ! any number of these lines
```

Transmissibilities T are reset at time 0 and at end of each timestep as

T(new)=tmod*T(original)

Entries of MULTIPLY can also have dynamic effect on transmissiblities. At time zero, T(original) is set to initial transmissibility. When TMODTABLEs are used, any MULTIPLY entries are applied to T(original).

Txmod, tymod, and tzmod values are computed for each cell based on its pressure, with no extrapolation outside table limits, and no hysteresis. Then, for each neighbor transmissibility in each direction, the connected cell tmod values are averaged and applied to the transmissibility.

Edit/search your outfile for TMOD to see the TMODTABLE printout.

P-Z diagram (Swelling Test) calculation (P-Z)

This P-Z feature calculates and prints P-Z diagrams (swelling tests) at 0 time for various Injectants in compositional problems. The P-Z diagram is a plot of psat (bubble point or dew-point) vs mol fraction Z of Injectant in mixtures of Injectant and reservoir fluid.

The primary purpose is to (1) detect and flag equation-of-state (EOS) data which are so distorted that subsequent simulation time, effort, and cost are inadvisable. Other purposes are:

(2) Display any possibility of direct-contact miscibility (FCM). If the P-Z diagram is closed for a given Injectant then faster, FCM simulations may be performed if the reservoir is operated at pressures exceeding the maximum psat (psat*) of the diagram.

(3) You can enter numerous (maximum of 10) Injectant compositions with increasing degrees of enrichment to seek a closed diagram with psat* allowing FCM. The recovery for FCM may be higher than that of multiple-contact-miscibility (MCM) or immiscible operation.

(4) Location of the critical point.

The default Injectants used for P-Z diagram generation are:

Surface gas

CO2 if CO2 is one of the EOS components

N2 if N2 is one of the EOS components

You can specify additional Injectants by entering in Initial Data after the GRID dataline

```
P-Z  ninj (NOPRINT)
 z1,z2,…,zn  ! Injectant composition, ninj of these lines
```

where

ninj = number of additional Injectants

n = number of EOS components (Nc)

z_i = composition of additional Injectant (mol fractions)

If

```
P-Z OFF
```

is entered, the P-Z diagram calculation is skipped.

If NOPRINT is entered, the P-Z diagram is calculated but printout caused by detection of possible EOS distortion is suppressed. You might enter NOPRINT (or OFF) if previous run(s) indicated signs of distortion which you have checked and are ok (e.g. oil heavier than water or K-values increasing with increasing molecular weight (e.g. test2.dat)).

The conditions flagged as possible EOS distortion are:

- oil heavier than water
- oil lighter than gas (by 0.05 lbs/cu ft or more)
- K-values increasing with increasing component molecular weight
- oil or gas viscosities which are ‘unusual’ (oil viscosity > 50 cp or gas viscosity > 0.5 cp)

If surface separation of the entered composition fails (i.e. gives all liquid or all gas) then the program skips the P-Z diagram calculation for the surface gas injectant.

To view the diagram, search your output file for "P-Z".

Any possible distortion of the EOS is flagged – search your output file for "*EOS".

The diagram(s) are generated for each reservoir fluid composition listed in your composition vs depth table under INITIAL. If a large number of such compositions is entered then, if you wish, you can deactivate the P-Z diagram generation by entering P-Z OFF after the GRID dataline.

To explore the P-Z diagrams, you can, if you wish, perform Initialization-only runs – enter END as the first entry of Recurrent Data. Such runs require very little cpu time.

Example datasets are eos1.dat,out and eos2.dat,out.

Note:

1. The P-Z diagram is 'closed' if psat is found for all mixtures $0 \leq Z \leq 1$ and is 'open' otherwise, with the following exception. A closed diagram may print "PSAT NOT FOUND" (search output file for "FOUND") at Z values at or close to Z=1.0 simply because the pure Injectant does not have a psat. In that case, if all other psat values exist, the diagram is closed.
2. The literature denotes the minimum miscibility pressure (MMP) as the lowest pressure at which multiple-contact miscibility is achievable. The closed-diagram psat* (maximum psat for $0 \leq Z \leq 1.0$) discussed here is the lowest pressure at which direct-contact (FCM) miscibility is achievable and might be denoted by (MDCMP). This distinction leads to a purpose of the P-Z feature which may be as important as that of detecting EOS distortion. As an example, consider an enriched Injectant which yields (by PVT program analysis) an MMP of, say, 2300 psia, indicating that multiple-contact miscibility may be achieved if the reservoir is operated at pressures exceeding 2300 psia. If the printed (outfile) P-Z shows a closed-diagram psat* of 2500 psia, then direct-contact miscibility could be attained by operating the reservoir at pressures only 200 psi higher.
3. For an open P-Z diagram (search outfile for "PSAT NOT FOUND"), neither direct-contact nor, we believe, multiple-contact miscible displacement is possible.
4. Our tests to date indicate that any eos distortion flagged by this P-Z feature does not relate to the issue of three hydrocarbon phases (e.g. low-temperature CO2 floods). That is, the existence of three hydrocarbon phases (assuming a 'good' eos) will not cause flagged eos distortion messages.
5. The 'unusual' gas viscosity > 0.5 cp condition flagged is arbitrary to the extent that gas phase viscosities at pressures up to several thousand psia are generally < 0.1 cp but a high molecular weight gas at sufficiently high pressure may have a viscosity greater than 0.5 cp.

1.5 Grid Block Properties: Arrays

Grid block properties such as depth, thickness, porosity, etc., are entered as arrays. Following sections describe global limiters of those array values, the format of the arrays, and the individual properties.

Global property limits (PVCUT, THCUT, TRFCUT, TRFMAX, TZMAX, THVEMAX)

```
PVCUT v  ! All grid blocks with pv < v res bbls are zeroed (inactive).
THCUT v  ! All grid blocks with net thickness < v ft are zeroed.
TRFCUT v ! All fault non-neighbor connections with transmissibilities
         !             < v rb-cp/d-psi are eliminated.
TRFMAX v ! Fault connection transmissibilities > v are set equal to
         !              v rb-cp/d-psi.
TZMAX v  ! z-direction transmissibilities > v are set equal to v rb-cp/d-psi.
THVEMAX v ! ve thicknesses > v are set equal to v ft.
```

If you enter any of the above, put them near the beginning of your datafile following GRID rather than deep in array/other data, where they may remain unnoticed or forgotten.

TZMAX can also be entered at any time in the Recurrent Data.

Any TRFCUT,TRFMAX entries must precede any/all FAULT entries in Initial Data.

Additional global limits on dual porosity variables, MAXLZ/TH, TEXMAX, and TEXDMAX are described in the dual porosity section.

Array Format

Gridblock property array data are entered in the keyword format:

PROP MODE

followed by data and optionally followed by keyword MOD. Eligible keywords PROP are:

DELX	x-direction block dimension, feet
DELY	y-direction block dimension, feet
KX(KY,KZ)	x-(y-,z-)direction permeability, md
POROS	porosity, fraction
PV	pore volume, rb, optional
PVF	multiplicative factor on PV, optional, default=1.
DEPTH	depth to top of block at xy center, feet
THICKNESS	gross thickness, feet
THICKNESS NET	net thickness, feet, optional
NET/GROSS	ratio of net to gross thickness, fraction, optional (default=1.0)

ROCKTYPE	rocktype, number of Saturation Table assigned to the block, integer, optional (default = 1)
COMPACTYPE	compaction table number, integer, optional (default=0)
INITREG	Initialization (equilibration) Region number (default=1)
CF	rock pore volume compressibility, 1/psi, optional, default is MISC value
PVTTYPE	pvt type, integer, needed for multiple-pvt problems (default=1)
REGION	Region number to which the gridblock is assigned, default = 0 (no assignment to any Region), optional
SWC	connate water saturation, fraction, optional
SWCG	connate water saturation S_{wcg} in a gas-water system, used only if k_{rw} is calculated using Baker's SWLI method.
SORW	residual oil saturation to water, fraction, optional
SORG	residual oil saturation to gas, fraction, optional
SGC	critical gas saturation, fraction, optional
SGR	maximum residual (trapped) gas saturation, fraction, optional
KRWRO	water relative permeability at residual oil saturation to water, optional
KROCW	oil relative permeability to water at connate water saturation, optional
KRGRO	gas relative permeability at residual oil saturation to gas, optional
JREF	Leverett J-function array, optional
TX	x-direction transmissibility, rb-cp/d-psi, optional
TY	y- " " " "
TZ	z- " " " "
TXF	multiplicative factor on TX, optional, default=1.
TYF	multiplicative factor on TY, optional, default=1.
TZF	multiplicative factor on TZ, optional, default=1.
TRACERF	initial tracer fractions, optional
RESERVOIR	reservoir number, optional, default=1

Eligible keywords MODE are:

VALUE, CON, XVAR, YVAR, ZVAR, EQUALS KX, EQUALS KY, LAYER

The entry format for each array PROP is the same,

```
PROP   MODE
 value(s)
MOD                          !  optional
```

```
  i1 i2 j1 j2 k1 k2   @  v   !  any number of lines
```

The value(s) entered following PROP depend upon CON,XVAR, etc. as

CON 1 value

XVAR N_x values

YVAR N_y values

ZVAR N_z values

VALUE $N_xN_yN_z$ values, ijk indices varying x(i) first, y(j) second, z(k) third.

(((PROP(I,J,K),I=1,NX),J=1,NY),K=1,NZ)

LAYER N_xN_y values, x(i) first, y(j) second

((PROP(I,J),I=1,NX),J=1,NY)

The six integers i1 - k2 define a portion of the grid for modification. The symbol @ is one of: = + * > < CUT and must have one or more blanks on both sides of it. The value v is a modification value, as follows:

symbol	action within the entered grid portion
=	PROP = v
+	PROP = PROP + v
*	PROP = PROP * v
>	All nonzero PROP values must be > v (nonzero values which are less than v are reset to v)
<	All nonzero PROP values must be < v (larger values are reset to v)
CUT	If PROP is less than or equal to v, PROP = 0

An additional array format is available for specifying values of a multiplier array to be applied to previously-defined x-direction permeability (MODE indicates the manner of input of the multiplier array, any of the modes other than EQUALS are allowed):

```
PROP EQUALS KX* MODE
 value(s)
MOD                          !  optional
 i1 i2 j1 j2 k1 k2   @  v    !  any number of lines
```

If transmissibilities are entered from a grid program, using TX, TY, TZ, then the arrays DELX, DELY, THICKNESS or THICKNESS NET, KX, KY, and KZ are not needed or used in calculating transmissibilities. However, some or all of these arrays are needed if:

- well Pi's are to be internally calculated,
- Fetkovich aquifer PI's are to be internally allocated to connected faces,
- VE calculations are being performed and the THVE array is not input, or
- TEX and TEXD are to be internally computed for DUAL cases.

For corner-point or unstructured grids, the arrays DELX, DELY, THICKNESS, and THICKNESS NET are Cartesian approximations of the true block geometry and may be sufficient for engineering purposes; otherwise all values noted above that can be internally computed should instead be computed externally (by the gridding program) and directly specified as data (well PI's, aquifer connection T's, THVE array, TEX and TEXD).

Similarly, if pore volumes are entered from a grid program using the PV array, then the POROS array is not needed or used in calculating pore volume. However, if compaction tables are entered, porosity must be known and the POROS array is required.

Gridblock dimensions (DELX, DELY)

The keywords DELX and DELY enter the gridblock $\Delta x(i)$ and $\Delta y(j)$ horizontally measured dimensions (feet), normally as follows:

DELX XVAR

$\Delta x(i)$ $i = 1,2,3,..,N_x$ N_x values

DELY YVAR

$\Delta y(j)$ $j = 1,2,3,..,N_y$ N_y values

Optionally, Δx and Δy may be entered as full-grid arrays, using VALUE.

Examples:

```
DELX  VALUE
    NxNyNz Δx values in normal xyz (ijk) order
DELY  VALUE
    NxNyNz Δy values in normal xyz (ijk) order
```

For a variable-width x-z cross-section, enter DELY as

```
DELY XVAR
   Nx values
```

Gridblock thickness (THICKNESS, THICKNESS NET, NET/GROSS)

```
THICKNESS or THICKNESS NET or NET/GROSS   MODE
 value(s), ft or ft or fraction
MOD                          !  optional
 i1 i2 j1 j2 k1 k2   @  v    !  any number of lines
```

THICKNESS is almost always required. An exception may be the case of input pore volumes, depths (using VALUE option), transmissibilities, and well PI values, and when no option requiring gross thickness is used. If net thickness is not entered using NET/GROSS or THICKNESS NET then the program internally sets net thickness to gross thickness. Pore volume is calculated by multiplying entered porosities by net thickness (not by gross thickness). Transmissibilities and well productivity indices are calculated by multiplying entered permeabilities by net thickness (not by gross thickness).

Gridblock depth (DEPTH, CENTER)

Depth is measured positively, vertically downward, in units of feet. Default DEPTH specifications are at the top, xy center of the grid block. Optionally, depths can be specified at the xyz block centers (CENTER option). All depths are measured relative to a common datum. Depth is 0 at this datum. The datum is arbitrary. For the purpose of discussion throughout this manual, the datum is chosen as sea level.

LAYER is used with DEPTH to read only the top layer N_xN_y values of depth. If you enter full grid DEPTH using VALUE, gaps and overlaps will generally occur. Little error checking is done in that regard.

If the reservoir is horizontal, then enter

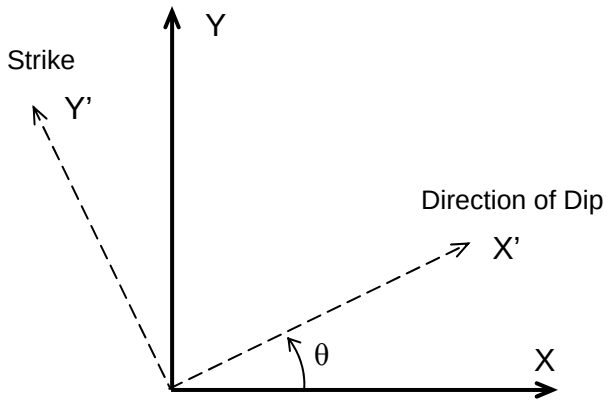
```
DEPTH  CON
 v
```

where v is the constant depth in feet to the top surface of the reservoir.

If "DEPTH CON" or "DEPTH LAYER" is used, the depths to the tops of all other layers are calculated internally from the entered depths to the top of the first layer and the entered gross thicknesses. The model internally calculates final grid block depths as the depths at the grid block centers (i.e. at the grid points).

If you enter CENTER on the DEPTH dataline, the program will interpret your entered depth values as depths to block centers and will not internally change them. For example, DEPTH CON CENTER or DEPTH LAYER CENTER or DEPTH VALUE CENTER.

In cases using the DUAL option, entered values of DEPTH for fracture gridblocks are arbitrary – they are internally set to matrix block values.



Case of a Dipping Slab Reservoir

The reservoir is a dipping slab (rectangular parallelopiped) with dimensions L_x, L_y, L_z in the x, y, and z-directions, respectively. As shown in the Figure, the XYZ axes form an orthogonal cartesian coordinate system; the XY plane is the top surface of the reservoir and the (downward) Z axis is not necessarily vertical. The direction of dip - direction of steepest descent - is that of the X' axis shown in the Figure. The Y' axis is the strike, perpendicular to the direction of dip.

Enter DELX, DELY, and THICKNESS (or DELZ) measured along the XYZ axes to define the grid. Then enter

DEPTH z α θ

z = depth to the top center of grid block (1,1,1), feet.

α = dip angle, the angle between the X' axis and the horizontal, degrees.

θ = the angle of the dip direction from the X axis, degrees.

Do not use the CENTER option on the DEPTH card.

Example:

For a 5-layer problem, initial pressure is entered as 5600 psia at a reference depth of 6020 ft and entered data are

```
DEPTH CON
  6000
THICKNESS  CON
   40
```

The program will print the depth array (map) showing layer depths of 6020, 6060, 6100, 6140, 6180 ft (block-center depths). The initial pressure map will show a value of 5600 psia in layer 1.

Gridblock porosity or pore volume (POROS, PV)

```
POROS or PV   MODE
 value(s), fraction or reservoir barrels (rb)
MOD                          !  optional
 i1 i2 j1 j2 k1 k2   @  v    !  any number of lines
```

If POROSBASE is not entered, then entered gridblock porosities or pore volumes are values at initial pressure. If POROSBASE is entered, then entered gridblock porosities or pore volumes are values at the specified global value of pbase.

If PV is entered, POROS is not necessary for calculation of pore volume. However, if compaction is used, porosity is required and POROS must be entered (fraction).

The model is coded in active-block mode, incurring no storage or cpu costs for grid blocks missing due to reservoir geometry. It is important to enter zero porosity for missing grid blocks. Alternatively, you can enter some cutoff value for porosity in missing blocks using the MOD keyword with the POROS array or you can use the MODIFY PV keyword as discussed below.

Gridblock permeabilities (KX, KY, KZ)

```
KX or KY or KZ   MODE
 value(s), md
MOD                          !  optional
 i1 i2 j1 j2 k1 k2   @  v    !  any number of lines
```

Additional options are as follows:

```
KX VALUE
  ... NxNyNz values
KY  EQUALS KX
KZ  EQUALS KX
MOD
 i1 i2  j1 j2  k1 k2  * .01
```

This results in $k_y = k_x$ and $k_z = .01k_x$ throughout the specified region. For the case of uniform anisotropy in the xy planes, KY could be entered as

```
KY   EQUALS KX
MOD
 1 Nx 1 Ny 1 Nz  * v
```

which results in $k_y = v\ k_x$ throughout the grid. The option

```
KX  VALUE
 .. data
KZ   EQUALS KX*  VALUE  ! (or KY  EQUALS KX*  VALUE )
  ... NxNyNz values ...
```

results in k_z (or k_y) equalling k_x multiplied by a spatially variable factor (a multiplier array).

Permeabilities may optionally be specified as a function of porosity and connate water saturation (Swc), by rock type. The input format is

```
KX or KY or KZ   FUNCTION
irt c1 c2 c3 c4 c5
MOD                          !  optional, applied to block perm values
 i1 i2 j1 j2 k1 k2   @  v    !  any number of lines
```

where permeability is functionally expressed as

$$K = (c2*\Phi^{c3} * Swc^{c4}+c5)^{1/c1}$$

and where irt is rock type, Φ is porosity, and Swc is connate water saturation.

The functions used may give unanticipated or unrealistic permeability values. Extremely large permeability values will cause run cpu times much larger than anticipated or necessary. We recommend that each permeability FUNCTION array entry should be followed with a global MOD dataline imposing a maximum permissible value. Also consider using a global MOD CUT dataline avoiding permeabilities of very small values, which will have negligible effect on production.

Notes on use of permeability FUNCTION option:

1. All Saturation Tables, the POROS array, and (if entered) the SWC and ROCKTYPE arrays must be entered before any permeability FUNCTION entries.
2. Entered irt values must include all rocktypes assigned to gridblocks by an entered ROCKTYPE array. If the ROCKTYPE array is not entered, then data for the default rock type irt=1 are required.
3. Entered irt values must lie in the range 1 to nrtmax, where nrtmax is the highest entered Saturation Table number.
4. A Saturation Table must be entered for each entered irt.
5. If c1 is entered as zero, a fatal error occurs.
6. Rock type (irt) may be entered as a range, i.e. 1 – 5, with spaces between numbers and dash.

See example problem testfunction.dat and its output file.

Gridblock transmissibilities (TX, TY, TZ)

```
TX or TY or TZ   MODE
 value(s) , rb-cp/d-psi
MOD                          !  optional
 i1 i2 j1 j2 k1 k2   @  v    !  any number of lines
```

Transmissibilities are plus-face values. That is, the T_x stored for grid block i,j,k is the x-direction transmissibility between blocks i,j,k and i+1,j,k. If you are entering and/or modifying transmissibility values as minus-face values, then enter the keyword TRMINUS (no data) in Initial Data.

Gridblock pore volume and transmissibility multiplicative modifiers (PVF, TXF, TYF, TZF)

```
TXF or TYF or TZF   MODE
 value(s)
MOD                          !  optional
 i1 i2 j1 j2 k1 k2   @  v    !  any number of lines
```

The TXF array is initialized to 1.0 prior to reading of Initial Data. After Initial Data are read, the TX transmissibilty array is internally calculated if it was not entered. The TX array values are then multiplied by the respective values in the TXF array. The TX and TXF values are then altered by any MODIFY TX entries of the multiplicative type. The TX values are also altered by any MODIFY TX entries of the non-multiplicative type. The TXF map may be printed and written to file for 3D visualization using the keywords MAPSPRINT and MAPSFILE, respectively. The use of PVF, TYF and TZF with PV, TY and TZ, respectively, is identical to that just described for TXF and TX.

Transmissibility multipliers are plus-face values. That is, the T_{xf} stored for grid block i,j,k is the x-direction transmissibility multiplier between blocks i,j,k and i+1,j,k. If you are entering and/or modifying transmissibility and/or transmissibility multiplier values as minus-face values, then enter the keyword TRMINUS in Initial Data.

Example:

The entered or calculated value of TX for a certain grid block is 100. The entered TXF for the block is .7. A MODIFY TX entry for the grid block is * .4. The final TX and TXF values are 28 and .28, respectively.

Note:

The intended use of the multipliers is to represent the cumulative effect of many changes made to the reference array values, perhaps over the course of many runs made towards a history match. The ‘reference’ values are here defined as those entered in or calculated from Initial Data. As such, where PVF,TXF,TYF,TZF are used any non-multiplicative MODIFY PV,TX,TY,TZ entries should be strictly avoided, and for greatest simplicity, we recommend that all MODIFY entries should be avoided in favor of Initial Data entry of multiplier array values.

Gridblock initialization (equilibration) region (INITREG)

```
INITREG   MODE
 value(s)
MOD                          !  optional
 i1 i2 j1 j2 k1 k2   @  v    !  any number of lines
```

By default, all grid blocks are assigned to the Initialization Region 1. If you wish to initialize different grid regions with different parameters (e.g. woc, composition, etc.) then use the INITREG array to assign the grid blocks to your desired Initialization Regions 1,2,... . If you specify more than one Initialization Region, then each INITIAL entry (described above) must be followed by an integer specifying the desired initialization Region.

Gridblock residual saturations (SWC, SORW, SORG, SGC, SGR, SWCG)

```
SWC or SORW or SORG or SGC or SGR or SWCG   MODE
 value(s)
MOD                          !  optional
 i1 i2 j1 j2 k1 k2   @  v    !  any number of lines
```

If the endpoint saturation arrays SWC, SORW, etc. are not entered, then the grid block values of S_{wc}, S_{orw}, etc. are those of the Saturation Table assigned to the block, except as described below.

Connate water saturation Swc may optionally be specified as a function of porosity, by rock type. The input format is

```
SWC   FUNCTION
irt c1 c2
MOD                          !  optional, applied to block Swc values
 i1 i2 j1 j2 k1 k2   @  v    !  any number of lines
```

where Swc is functionally expressed as

$$Swc = MAX(c1*\Phi^{c2}, 0.)$$

and where irt is rock type and Φ is porosity.

The functions used may give unanticipated or unrealistic Swc values. In some cases, the values may cause run cpu times much larger than anticipated or necessary. We recommend that any SWC array entry (FUNCTION or not) be followed by a global MOD dataline restricting Swc to a reasonable, desired max Swc. Large values of Swc (approaching 1-Sorw) can cause large (excessive) cpu time, unless they occur in blocks having very low kx, ky, and kz. These occurrences are flagged as WARNINGs in the Sensor output file and may result in necessary changes made to input data by the program. Again, **not heeding these warnings can cause internal data changes and extremely high cpu times**.

Notes on use of SWC FUNCTION option:

1. All Saturation Tables, the POROS array, and (if entered) the ROCKTYPE array must be entered before any SWC FUNCTION entries.
2. Entered irt values must include all rocktypes assigned to gridblocks by an entered ROCKTYPE array. If the ROCKTYPE array is not entered, then data for the default rock type irt=1 are required.
3. Entered irt values must lie in the range 1 to nrtmax, where nrtmax is the highest entered Saturation Table number.
4. A Saturation Table must be entered for each entered irt.
5. If c1 and c2 are entered as zero, a fatal error occurs.
6. Rock type (irt) may be entered as a range, i.e. 1 – 5, with spaces between numbers and dash.

See example problem testfunction.dat and its output file.

Scaling of endpoint saturations (NEWSOR)

If the S_{wc} array is input, then in order to prevent negative saturation intervals, the gridblock values of any other saturation endpoints that are not input as array data will be set equal to their table value scaled by the factor

$$f = (1-S_{wc}(\text{array}))/(1-S_{wc}(\text{table}))$$

as described in Appendix 11.

If you input the S_{wc} array and do not wish this scaling to occur for the other endpoints, either enter all other endpoints as array data, or enter

```
NEWSOR 0
```

in the Initial Data before the array input. The program will then change no endpoint saturations regardless of what endpoint arrays are entered and regardless of what errors occur.

In any event, whether endpoint arrays are entered or not, Sensor flags saturation endpoint interval errors with *WARNING messages or *ERROR messages.

Gridblock relative permeability endpoints (KRWRO, KROCW, KRGRO)

The endpoint scaling includes endpoint relative permeabilities krwro, krocw, and krgro. The optional array entries in Initial Data are:

```
KRWRO or KROCW or KRGRO  MODE
 value(s)
MOD
 i1 i2 j1 j2 k1 k2   @  v   !  any number of lines
```

None, any one, any two or all three arrays may be entered. The arrays may be printed (MAPSPRINT) and written to file (MAPSFILE).

Note:

All entered kr and endpoint kr are normally less than or equal to 1.0 but values > 1.0 are permissible. If at time zero values greater than one are detected for any set of saturations, specification of KRWLIMIT, KROLIMIT, and/or KRGLIMIT indicating maximum permissible relative permeability values will be required.

You can print (MAPSPRINT) and write to file (MAPSFILE) endpoint arrays KRWRO, KROCW, KRGRO and current kr maps KRW, KRO, KRG.

Edit your outfile and search for “endpoint” and “MAP” to see useful information related to kr, krlimits, and kr maps.

Gridblock reference Leverett J function (JREF)

```
JREF   MODE (nopt)
 value(s)
MOD                          !  optional
 i1 i2 j1 j2 k1 k2   @  v    !  any number of lines
```

The Leverett J-function can optionally be used to scale entered tabular capillary pressures, and the reference value can also be entered as a global constant, as described in Section 1.4, Relative permeability and capillary pressure, Leverett J function.

Gridblock rock (formation) compressibility (CF)

```
CF   MODE
 value(s), 1/psi
MOD                          !  optional
 i1 i2 j1 j2 k1 k2   @  v    !  any number of lines
```

Grid blocks not assigned to a compaction table use a rock (formation) compressibility equal to the c_f value on the MISC dataline if the CF array is not entered. If the CF array is entered, such blocks use the c_f value in the CF array.

Gridblock rocktype (ROCKTYPE)

```
ROCKTYPE   MODE
 value(s)
MOD                          !  optional
 i1 i2 j1 j2 k1 k2   @  v    !  any number of lines
```

The ROCKTYPE integer array assigns each grid block to one of the entered Saturation Tables. Do not enter ROCKTYPE if only one Saturation Table is entered.

Gridblock compaction table (COMPACTYPE)

```
COMPACTYPE   MODE
 value(s)
MOD                          !  optional
 i1 i2 j1 j2 k1 k2   @  v    !  any number of lines
```

The COMPACTYPE integer array assigns each gridblock to one of the entered compaction tables. Do not enter COMPACTYPE if no compaction tables are entered. If no compaction tables are entered then rock compressibility c_f (entered by keyword MISC or array CF) is used for all blocks.

Example:

Two compaction tables are entered for a problem with a 44x26x12 grid. The top four layers are to use table 1 and layers 5-12 are to use table 2:

```
  COMPACTYPE  ZVAR
    4*1 8*2
or
  COMPACTYPE CON
   1
  MOD
```

```
  1 44  1 26  5 12  = 2
```

Example:

In a 44x26x12 grid problem, one compaction table is entered. Layers 1-10 are to use that table and layers 11-12 are to use the c_f value entered by MISC:

```
COMPACTYPE CON
  0
 MOD
   1 44  1 26  1 10 = 1
```

or

```
COMPACTYPE ZVAR
  10*1 2*0
```

Gridblock transmissibility stress dependence table (TMODTYPE)

```
TMODTYPE   MODE (NOPRINT)
 value(s)
MOD                          !  optional
 i1 i2 j1 j2 k1 k2   @  v    !  any number of lines
```

The TMODTYPE integer arrays assigns each grid block to one of the entered transmissibility stress dependence tables (TMODTABLE). The optional NOPRINT label suppresses printout of the entered TMODTABLEs to the output file.

Gridblock initial tracer fractions (TRACERF)

If the TRACER option is active (see Section 1.6), initial tracer fractions for each traced component are specified using the TRACERF array. Defaults are 1.0 for the last fraction and 0 for the others, for each traced component. The format is

```
TRACERF MODE
ifrac iphase  ! initialize tracer fraction ifrac in phase iphase
              ! iphase = OIL, GAS, or WATR
...data...
```

Do not specify data for the last tracer fraction in any phase. These are internally computed to insure that the fractions sum to 1.0.

Matrix-fracture exchange transmissibilities (TEX)

```
TEX   MODE
 value(s), rb-cp/d-psi
MOD                          !  optional
 i1 i2 j1 j2 k1 k2   @  v    !  any number of lines
```

Exchange transmissibilities or at least one of the matrix block sizes are required if the DUAL option (see Section 1.7) is active. Only the matrix subarray values are used – fracture subarray values are arbitrary.

Matrix-fracture exchange diffusive transmissibilities (TEXD)

```
TEXD   MODE
 value(s), ft
MOD                          !  optional
 i1 i2 j1 j2 k1 k2   @  v    !  any number of lines
```

Diffusive transmissibilities or at least one of the matrix block sizes should be input if the DUAL option (see Section 1.7) is active. Only the matrix subarray values are used – fracture subarray values are arbitrary.

Matrix block sizes (fracture spacings) (LX, LY, LZ, LZTEX)

```
LX or LY or LZ or LZTEX   MODE
 value(s), ft
MOD                          !  optional
 i1 i2 j1 j2 k1 k2   @  v    !  any number of lines
```

Either TEX or at least one of the matrix block size arrays are required if the DUAL option (see Section 1.7) is active. Only the matrix subarray values (values for the first Nz/2 layers) are used – fracture subarray values are arbitrary, except for LZ values if the LZWG option is invoked on the DUAL card. Defaults are zero, indicating absence of fracturing normal to the x, y, or z directions. A negative value, say –f, for LZ can be input to indicate that Lz = f * delx.

Tortuosity (TOR)

```
TOR   MODE
 value(s), dimensionless
MOD                          !  optional
 i1 i2 j1 j2 k1 k2   @  v    !  any number of lines
```

Tortuosity should be input for Dual Porosity cases using the DIFFUSION option in which the diffusive transmissibilities TEXD are computed internally by the program. Default is 2.

VE Thickness (THVE)

```
THVE   MODE
 value(s)
MOD                          !  optional
 i1 i2 j1 j2 k1 k2   @  v    !  any number of lines
```

Entry of THVE activates the vertical equilibrium (VE) option and specifies the total block thicknesses to be applied in computing the VE capillary pressures. The total block thickness thve should be equal to zmax – zmin, where zmax and zmin are the maximum and minimum depths of any point contained in the block.

If the THVE array is entered:

- If all values are 0, VE is turned off regardless of whether VE or VE DIP were entered or not.
- If any value > 0 then:
  - VE is activated (on) whether VE or VE DIP were entered or not.
  - Entry of VE or VE DIP or no entry of either makes no difference.

If the THVE array is not entered:

- If neither of VE or VE DIP is entered VE is off.
- If VE is entered, VE is on and THVE=gross thickness.
- If VE DIP is entered, VE is on and THVE is internally corrected for dip.

Fault connections (FAULT, AREA, FSURFACE)

All keywords discussed here are in Initial Data.

The FAULT entry specifies non-neighbor connections (transmissibilities) and modification of normal-neighbor connections (transmissibilities) across a fault. These transmissibilities may be constant over time or, if the FAULTMOD option described in the next section is used, may change with time either in a specified manner or with stress, where stress is defined as pressure difference across the fault.

The modification of normal-neighbor connections at time zero, stress-dependent or not, can be accomplished using the (optional) FSURFACE keyword under FAULT. The specified multipliers will apply in addition to any other modifications made directly to the transmissibilities through MOD and MODIFY entries, and to those made by entry of transmissibility multiplier arrays.

```
FAULT ifault wf <itab> <AREA>
FSURFACE nfseg          ! optional
I1 I2 J1 J2 K1 K2 wfs   ! modification of normal-neighbor transmissiblities
I1 I2 J1 J2 K1 K2 wfs
... nfseg of the above I1 I2 etc. lines
! Non-neighbor connection data may follow:
I1 J1 K1 I2 J2 K2 ta wfc
I1 J1 K1 I2 J2 K2 ta wfc
.. etc. any number of non-neighbor connections
```

Definitions:

ifault	fault # (positive integer)
itab	faultmod (stress) Table to which the fault is assigned (positive integer or enter 0 or omit for no stress dependence)
wf ,wfs,wfc	multiplicative factors
nfseg	number of fault surface segments

AREA	entered non-neighbor connections ta are area (sq ft) rather than transmissibility (rb-cp/d-psi) (optional, not available for radial problems)
ta	non-neighbor transmissibility (rb-cp/d-psi) or area (sq ft)

Normal-neighbor (FSURFACE) data:

The FSURFACE entry applies wf*wfs to normal-neighbor Tx,Ty,Tz throughout each segment. Enter the description of the fault surface using as many segments as necessary.

Enter I1,I2 negative (Tx between i1 and i2 are modified) if the fault segment is in the J (y) direction.
Enter J1,J2 negative (Ty between j1 and j2 are modified) if the fault segment is in the I (x) direction.
Enter K1,K2 negative (Tz between k1 and k2 are modified) if the fault segment is in the K (z) direction.

Note: Use of the term "segment" here is poor semantics. A 'segment' in the y direction is an x-z plane; a 'segment in the x direction is a y-z plane; a 'segment' in the z direction is an x-y plane.

If the I's are negative, |I2-I1| must be 1. If the J's are negative, |J2-J1| must be 1. If the K's are negative, |K2-K1| must be 1.

In absolute value, each of I2, J2, K2 must be equal to or greater than I1,J1,K1, respectively.

If the I's are negative, all normal Tx defined by the segment are multiplied by wf*wfs.
If the J's are negative, all normal Ty defined by the segment are multiplied by wf*wfs.
If the K's are negative, all normal Tz defined by the segment are multiplied by wf*wfs.

As a consequence of all the above, no normal Tx or Ty or Tz is multiplied by wf*wfs more than once, i.e. repeated modifications of the same connection are ignored and a warning is printed.

The sequence of normal-neighbor transmissibility (T) calculation is as follows:

1. T values are either entered as TX, TY, and TZ arrays and MOD's or are calculated using block dimensions and permeability arrays
2. Any entered transmissiblity multiplier arrays are applied to T
3. FSURFACE data are applied to T
4. Any MODIFY data are applied to T
5. The value of T after step 4 is stored as T_0 for subsequent recurrent modification as described under FAULTMOD, TMODTABLE, and MULTIPLY.

Non-neighbor connection data:

Block 1 (i1 j1 k1) is connected to non-neighbor block 2 (i2 j2 k2). The two blocks must not be normal neighbors. If wfc is omitted, the program internally sets it to 1.0. The ta values are multiplied by wf*wfc.

<u>If AREA does not appear on FAULT dataline:</u>

All I1,J1,K1,I2,J2,K2 must be positive integers within the grid. The ta values are positive transmissibilities (rb-cp/d-psi).

If AREA appears on the FAULT dataline:

I2,J2,K2 must be positive integers within the grid.

One and only one of I1,J1,K1 must be negative.

The ta value is area (sq ft) of interface between blocks 1 and 2.

The internally calculated non-neighbor transmissibility T_0 is

T_0 = .0011274(2*area)/(L1/k1+L2/k2) rb-cp/d-psi

where

k1,L1 = kx,delx of block 1 if i1 is negative
= ky,dely of block 1 if j1 is negative
= kz,delz of block1 if k1 is negative
k2,L2 = kx,delx of block 2 if i1 is negative
= ky,dely of block 2 if j1 is negative
= kz,delz of block 2 if k1 is negative

kx,ky,kz = absolute permeabilities in the x,y,z directions, respectively, md

delx,dely,delz = gridblock dimensions, ft

The FAULT entry may include FSURFACE entries only or non-neighbor entries only or both. The above transmissibilities specified by FAULT are stored as T_0, for possible recurrent modification as a function of stress using FAULTMOD.

Fault Connection Modifications (FAULTMOD, CHECKFAULT), optional

If entered, the FAULTMOD keyword must be entered in Initial Data but may be re-entered in Recurrent Data. FAULTMOD cannot be used with keywords RADIAL or NINEPOINT.

To view printout of FAULTMOD and other fault data, edit your outfile and search for "FAULT".

The FAULTMOD Table is used to modify all normal-neighbor and non-neighbor connections defined by FAULT by a specified multiplier or as a function of the average pressure drop ('stress') across the fault:

```
FAULTMOD itable
    stab    f    ! as many lines as desired
```

where

itable = faultmod Table #
stab = stress, psi
f = multiplier on T_0
stabi >= stabi-1, i.e. must be monotonic
no extrapolation outside of Table stab limits is used

If you enter n faultmod Tables, give them numbers (itable) 1,2,…,n. There are hard-coded maximum dimensions of 50 faultmod tables and 15 stab entries per table.

If the table has only one entry, then the entered value for stress, stab, is irrelevant and is not used, and the entered multiplier f is applied to T_0.

If the table has more than one entry, stress across the fault s is calculated at the end of each timestep as

stress = s = sum|p1-p2|/nfault (time t) – sum|p1-p2|/nfault (time 0)

where

sum is over all connections of the fault (neighbors and non-neighbors),
p1-p2 is the difference between connected blocks' 1 and 2 pressures, and
nfault = number of connections in fault ifault.

The value of s is then used to interpolate in the FAULTMOD stab vs f Table for f. The fault connection transmissibilites are then reset to $f*T_0$. Hysteresis is used as follows: no reset of the transmissibilities is done unless current stress s exceeds the historical maximum of s.

Datafile fault1.dat is a small example test problem illustrating FAULT, FSURFACE, and FAULTMOD data input. It also has a keyword CHECKFAULT entry to help you check the setting of stress-fault non-neighbor transmissibilities. See the comments in fault1.dat regarding CHECKFAULT. If the latter is entered, edit/search your outfile for "CHECKFAULT" to see the printout.

Regions and superregions (REGION, REGNAME, SUPERREGION, SREGNAME, FIELDNAME)

Regions are optional. Their use allows printout and end-of-run summaries showing initial fluids in place, recoveries and production-injection rates and cumulatives for different portions of the grid. Regions and superregions can also be used for pressure control as described in Section 3.1 (PCON). The integer array keyword REGION assigns grid blocks to Regions 0-n. The program by default assigns all blocks to Region 0, which means no Region.

Example:

This assigns layer groups as Regions 1-4 in a 28 x 26 x 10 grid. Blocks in layer 8 are not assigned to any Region.

```
REGION CON
  0
 MOD
  1 28  1 26  1 1  = 1
  1 28  1 26  2 3  = 2
  1 28  1 26  4 7  = 3
  1 28  1 26  9 10 = 4
```

A given grid block may be assigned to a number of different Regions. Pvt type within a given Region may vary and include both compositional and black oil pvt types. A Region may include blocks from different Reservoirs. Continuing with the above

Example:

```
REGION
```

```
  5  PVTTYPE 3
  6  RESERVOIR 2
  7  INITREG 2
  8  INITREG 4
```

This defines Region 5 as all blocks of pvt type 3, Region 6 as all blocks in Reservoir 2, etc. The eligible keywords are restricted to PVTTYPE, RESERVOIR, and INITREG. This entry can be used with or without the REGION array entry.

Region names, up to 10 characters, may be entered as follows, using the above Example:

```
REGNAME
  1    LYR1
  2    LYRS2-3
  3    LYRS4-7
  4    LYRS9-10
  5    PVT3
  6    RES2
  7    INITREG2
  8    INITREG4
```

The program includes total field results in the Region Table printout. The entry

```
FIELDNAME  fname
```

will label the total field results with the entered name “fname”.

A Superregion is a collection of two or more Regions, entered using the keyword SUPERREGION. A Region may be a member of any number of Superregions. Superregion names may be entered.

Example:

Continuing with the above Example,

```
SUPERREGION
  1   1 2 3    ! Superregion 1 is Regions 1,2, and 3
  2   7 8      ! Superregion 2 is Regions 7 and 8
SREGNAME
  1   UPPERZONE
  2   ZONE2
```

Do not define any Superregion as a single Region. If you define n Regions, give them numbers 1,2,3,..,n. If you define m Superregions, give them numbers 1,2,..,m.

Multiple reservoirs, multiple pvt (RESERVOIR, RESZERO, PVTTYPE)

Any number of Reservoirs and pvt types may be used. The multiple pvt types may include both black oil pvt types and compositional pvt types. Pvt type may vary within a given Reservoir but all pvt types must be compositional or all must be black oil. That is, a given Reservoir must be a compositional Reservoir or a black oil Reservoir. Any number of Initialization Regions may be used but each must contain a uniform (constant) pvt type. Normally the set of grid blocks in a given Initialization Region will all belong to the same Reservoir, but that is not necessary. Any

number of Regions may be defined for reporting purposes. Pvt type may vary throughout a given Region and may include both compositional and black oil pvt types.

The conditions defining multiple Reservoirs are as follows:

a) There can be no transmissibility communication between any pair of reservoirs.

b) No well can be completed in more than one Reservoir.

If any transmissibility connects two different Reservoirs, an error message is printed and execution is terminated. If you enter the keyword

```
RESZERO
```

in Initial Data, the program internally zeroes all transmissibilities which connect different Reservoirs.

Datafile test15.dat illustrates a case where Reservoir 2 is areally (xy) separated from Reservoir 1, with use of RESZERO to zero the connecting transmissibilities.

The array entry

```
RESERVOIR  MODE (CON or ZVAR or VALUE or ..)
  ... data ...
```

assigns grid blocks to different Reservoirs.

The array entry

```
PVTTYPE  MODE (CON or ZVAR or VALUE or ..)
  ... data ...
```

assigns grid blocks to different pvt tables.

Example:

A problem with a 23x16x26 grid has no vertical communication between the following layer pairs: 8-9, 13-14, 21-22. No well is completed in more than one of the resulting isolated layer groups. Enter the array keyword RESERVOIR:

```
RESERVOIR CON
 1
MOD
 1 23  1 16   9 13  = 2
 1 23  1 16  14 21  = 3
 1 23  1 16  22 26  = 4
```

Omit this keyword and data if there is only one Reservoir. Do not use the D4 solver with multiple Reservoirs.

Different Reservoirs do not require the same number of Newton iterations per time step. Nor do they require the same number of solver iterations per Newton iteration. Therefore, significant cpu reduction can occur if the Reservoirs are solved separately. Datafile test14.dat is a 7000-day run of a 9000-block, 23x16x26 grid having four stacked Reservoirs. When the multiple Reservoirs are not defined in the datafile, the run cpu is 60 % larger.

An option involving altered DELX, DELY formats allows significant savings in cpu and storage costs in the case of multiple Reservoirs. In a grid of (say) N_x=70, N_y=40, N_z=16, three different Reservoirs are included. Some Reservoirs require a fine grid, others have coarser grids. For purposes of discussion, Reservoir 1 is the first 5 layers and requires the maximum (fine) grid of 70 x 40. Reservoirs 2 and 3 are as follows:

Reservoir	Layers	n_x x n_y
2	6-10	24 x 11
3	11-16	32 x 30

The DELX and DELY entries for this case are:

```
DELX  LAYER
  1 5
    70 non-zero Δx values
  6 10
    70 Δx values, the last 46 are 0
 11 16
    70 Δx values, the last 38 are 0
DELY  LAYER
  1 5
    40 non-zero Δy values
  6 10
    40 Δy values, the last 29 are 0
 11 16
    40 Δy values, the last 10 are 0
```

Thus, not only can the values of Δx(i), Δy(j) differ from Reservoir to Reservoir, but the number (n_x,n_y) can also vary. For Cartesian grids, the DELX (DELY) vector should not vary from layer to layer within a given Reservoir. This option of LAYER following DELX or DELY cannot be followed with MOD.

Entry of arrays (porosity, permeability, etc) must be on the basis of the full 70 x 40 x 16 grid. This requires that your grid or geological program writes single 70x40x16 element blocks including all Reservoirs, having many zeroes for undefined blocks in Reservoirs where n_x,n_y are less than N_x,N_y. If it doesn't, you need to write a small program reading all the Reservoir arrays and rewriting them in full single-array $N_xN_yN_z$ mode.

Sensor incurs no storage or cpu costs for the many missing grid blocks arising due to the smaller (n_x,n_y) Reservoirs.

1.6 Tracer Option (TRACER, NOTRACER, ONTRACER)

The program will calculate tracer fractions for any number of traced components in Impes mode in black oil and compositional simulations. Traced components can be any of the n_c+1 fluid components, including water. An unlimited number of tracer fractions can be used for each traced component. The tracer fractions sum to 1.0 for each traced component.

Tracer calculations can be useful in equity situations. The initial oil tracer fractions might be assigned (1,0,0), (0,1,0), and (0,0,1) in three different leases or equity sectors (blocks) of the reservoir. The produced tracer fractions, current and cumulative, at the wells or groups of wells then indicate where the produced oil came from. Tracer fractions can be assigned to injected gas (water) to distinguish between produced gas (water) originating from native vs injected gas (water).

Tracers can also be used to reduce cpu in some IMPES compositional problems, as discussed in Section 1.4 under "Automatic Conversion of EOS to K-value Table".

If TRACER is entered (in the Initial Data), remember to enter tracer fractions for all injection wells in Recurrent Data under keyword WELLTRACER, as described in Section 3.1 below. Default values of initial tracer fractions in the reservoir of each traced component equal to 0 for all but the last fraction, which is defaulted to 1.0. Enter TRACERF array data to set the initial tracer fractions, for all but the last fraction, in the reservoir. The last tracer fraction is internally calculated so that all tracer fractions sum to 1.0. Do not enter TRACERF for the last tracer fraction.

Tracer calculations can optionally be delayed to begin at any specified time, by using the NOTRACER and ONTRACER keywords. The initial tracer fraction distributions specified in the TRACERF data are those desired at the time that tracer calculations are to begin. At time zero, before the first recurrent TIME or DATE entry, enter the keyword

```
NOTRACER
```

to delay tracer calculations. At any later time, enter the keyword

```
ONTRACER
```

to begin the tracer calculations at that time. WELLTRACER data may be entered before any NOTRACER entry and/or after any ONTRACER entry. Test datasets are test1c.dat, test1d.dat. The problem is 1d horizontal injection of propane into propane, 50 blocks, with about 1 block pv injected per timestep at a constant 2 day timestep, 50 days = .5 pv injected. Two tracer fractions are assigned to traced component C3. The first 25 blocks have TRACERF initial tracer fractions (1,0), and the second 25 have (0,1), and injected tracer fractions (WELLTRACER data) are (0,1). Test1c tracer calculations start at time 0 and we see in the WELL TRACER TABLE produced tracer fraction 1 change from 0 to 1 at about 50 days, and from 1 to 0 at about 100 days, as expected. Test1d tracer initialization/calculations are delayed by 50 days, as is the tracer response.

Black-oil example:

Specify three traced components (oil, gas, water) with 2 tracer fractions in oil, 3 each in gas and water:

```
TRACER
  3 GAS          !  3 tracer fractions in component GAS
  3 WATR         !  3 tracer fractions in component WATR
  2 OIL          !  2 tracer fractions in component OIL
```

To initialize the tracer fractions, use the TRACERF array entry:

```
TRACERF VALUE or CON or ZVAR or .. etc.
```

```
  1 OIL          ! initialize tracer fraction 1 of OIL
    ... data ...

TRACERF VALUE or ....
  1  WATR       !  initialize tracer fraction 1 of WATR
    ...data ...
TRACERF VALUE or ..
  2  WATR       !  initialize tracer fraction 2 of WATR
   .. data ..

TRACERF VALUE or ..
  1  GAS        !  initialize tracer fraction 1 of GAS
   .. data ..

TRACERF VALUE or ..
  2 GAS         !  initialize tracer fraction 2 of GAS
 .. data ..
```

In the multiple reservoir case, the above keyword formats are "TRACER ires" and "TRACERF VALUE ires", where ires is reservoir number. Thus tracers can be used in certain reservoirs and not in others. See example datafile test1.dat. If ires is omitted, it is internally set to 1.

Equity blocks (WELLBLOCK, BLOCKNAME)

In equity situations, it is useful to calculate the tracer fractions for groups of wells, where each group consists of all producers in a given (equity) block. The cumulative tracer fractions for traced component OIL (or a heavy component in the compositional case) then reflect the source of the total oil produced from each block.

Example:

There are three blocks, denoted CompanyA, CompanyB, and CompanyC, in a black oil reservoir. Initial Data defines OIL as a traced component, with three tracer fractions:

```
TRACER
  3 OIL
```

The TRACERF entry is used to set initial tracer fractions (1,0,0), (0,1,0), and (0,0,1) in the grid blocks representing the acreage of Companies A, B, and C, respectively. The Recurrent Data entry

```
WELLBLOCK
    W1 – W12  1    ! Company A block producers
   W13 – W20  2    ! Company B block producers
   W21 – W26  3    ! Company C block producers
```

defines the producer well groups in each block and the block names are entered as

```
BLOCKNAME
    1 COMPANYA
```

```
2 COMPANYB
3 COMPANYC
```

Tracer fractions are printed in the outfile Well Table. Cumulative tracer fractions of (say) (.12,.82.,.06) for the Company B block 2 would mean that the block 2 total oil production was 12 % block 1, 82 % block 2, and 6 % block 3 oil. See datafile test7.dat as an example.

In Recurrent Data, the tracer fractions for injected gas and water must be specfied for each injection well:

Example:

Assign GAS tracer fractions $\{f_j\}$ as {1,0,0} for all gas injection wells G12-G15, inclusive. Assign WATR tracer fractions as {0,1,0} for the single water injection well W16:

```
WELLTRACER
  G12 - G15  GAS
    1. 0. 0.
  W16    WATR
    0 1 0
```

The % increase in cpu due to the tracer calculation is generally small. For example, cpu time increased by 7 % when 12 tracer fractions were used in a 4000-block, 7-component problem.

In some cases, oscillatory tracer fractions may occur and time step reduction may be necessary. Alternatively, CFL 2 or CFL can be entered, since the stable step logic applies to tracer fractions as well as to saturations and mol fractions.

Example:

A 7000-day run showed tracer fraction oscillation in the time period from 3400 to 4000 days. This 600-day period took 53 time steps for an average time step of 11.3 days. Resubmit the run with DTMAX = 7 entered at 3400 days and with dtmax restored after 4000 days. (Entry of "DTMAX 0" restores it to default value)

The program prints a table of current and cumulative tracer fractions for each traced component for all wells and blocks. This table can be viewed by editing the outfile and searching for "TRACER TABLE". The printout frequency of this table is the same as that for the Well Table (entered using keyword WELLFREQ).

1.7 Dual Porosity Option (DUAL, DIFFUSION, MAXLZ/TH, TEXMAX, TEXDMAX)

Note:

1. This option does not apply to radial grid systems and cannot be used with the multiple RESERVOIR option.
2. Diffusion applies only to compositional runs.
3. Other Dual (array) data: TEX, TEXD, LX, LY, LZ, TOR. At least one of TEX, LX, LY, LZ is required.

4. VE and IMPLICIT are automatically invoked by the DUAL entry (you do not need to enter the keywords VE or IMPLICIT).
5. Always search the output file for 'DUAL' to review DUAL data and warnings.
6. Wells are normally completed in the fracture system (layers Nz/2+1 to Nz). However, completions in the matrix gridblocks (layers 1 to Nz/2) are allowed – see discussion of "Completions" below.
7. All keywords for dual porosity/dual permeability are in Initial Data.
8. One application of this option is to reservoirs having unfractured regions dominated by very thin, highly permeable horizontal (or bedding-plane) streaks interbedded in very tight matrix. In such regions there are no vertical fractures; LX, LY arrays specify 0 Lx and Ly and the Lz array specifies the spacings of the thin permeable streaks.

The Sensor dual porosity option simulates cases where at least some of the reservoir is either (a) naturally fractured, or (b) dominated by thin permeable streaks in tight matrix. Any portions of the reservoir may be unfractured – i.e. may contain only continuous matrix. The discussion below uses the term "fractured" to denote either case of naturally fractured or thin permeable streaks.

The dual porosity option simulates either single permeability or dual permeability fractured systems. Single permeability systems are characterized by no matrix-to-matrix interblock flow in fractured regions. Dual permeability systems allow matrix-to-matrix interblock flow in all or parts of the fractured regions. Both systems represent matrix-to-matrix interblock flow in unfractured regions, fracture-to-fracture interblock flow in the fractured regions, and matrix-to-fracture interblock flow at the interfaces between unfractured and fractured regions (these connections are automatically computed by the program, if transmissibilities are not specified). Unfractured blocks are defined as those blocks with entered fracture porosities (in the POROS array) or fracture pore volumes (in the PV array) of zero. Of course, both single- and dual-permeability systems have matrix-to-fracture transfer within each grid block in the fractured regions (fracture pore volume > 0) where matrix pore volume is present. While arguably unusual, it is permissible for a fractured gridblock to have no matrix pore volume.

The dual porosity option is invoked by the Initial Data entry

```
DUAL nperm
```

where

nperm = 1 for single permeability systems

nperm = 2 for dual permeability systems (or for mixed single and dual permeability)

If nperm is 1, then Sensor will automatically zero any input matrix-to-matrix interblock transmissiblities in fractured regions, and all calculated values will be zero.

If nperm is 2, then

Sensor will not internally change any transmissibilities (either entered using TX, TY, TZ arrays or internally calculated using arrays KX, KY, KZ, DELX, DELY, THICKNESS

The entered or calculated transmissibilities (using any desired MOD's on entered arrays and MODIFY's on calculated arrays) may specify any desired mix of single permeability vs dual

permeability in fractured regions (for the nperm=2 case). If transmissibilities are not specified (if they are internally calculated) and if there are any single permeability regions, the user must zero matrix to matrix transmissibilities at the interface between neighboring pairs of single permeability gridblocks, and at the interface between neighboring single permeability and dual permeability gridblocks.

The matrix-fracture transfer rate (transient) and equilibrium within each gridblock must both be characterized in dual porosity modeling. Usually, the equilibrium state is more important than the transient. Matrix-fracture transfer occurs by convection and optionally by diffusion. Diffusion can occur through both the liquid and gas hydrocarbon phases.

Diffusion option

Diffusion is (or may be) needed only when injected gas differs significantly in composition from in-situ gas, such as for CO2 or N2 injection. To model matrix-fracture diffusion in compositional cases, enter:

```
      DIFFUSION   (f)
C    cpt # i     Dliq(i)     Dgas(i)
           1     Dliq1       Dgas1
           2     Dliq2       Dgas2
           .       .           .
          nc     DliqNc      DgasNc
```

where

Dliqi = component i liquid-liquid molecular diffusion coefficient, cm^2/sec
Dgasi = component i gas-gas molecular diffusion coefficient, cm^2/sec
f = optional diffusion coefficient scale factor

If f is entered, all Dliqi and Dgasi are multiplied by f.

If DIFFUSION is not entered, no matrix-fracture diffusion is calculated. Gas-gas diffusion coefficients are the order of .001 cm^2/sec at reservoir pressures and temperatures (3000+ psia, 150-250 deg F) and liq-liq diffusion coefficients are 10-100 times smaller. Diffusion coefficients decrease somewhat with increasing pressure and coefficients for heavier components tend to be somewhat lower than those for lighter components.

Note: The DIFFUSION keyword with associated data can also be entered in Recurrent Data to change the diffusion coefficients at any time.

Limits on Dual data

The optional entry

MAXLZ/TH f

limits any LZ array value greater than f *Δz to f *Δz.

The optional entry

TEXMAX texmax

limits TEX array values to a maximum of texmax rb-cp/d-psi.

The optional entry

TEXDMAX texdmax

limits TEXD array values to a maximum of texdmax ft.

Completions

A wellbore traversing a physical gridblock will intersect only a small fraction of the matrix blocks within that gridblock. Therefore, it is arguably physically unrealistic to complete an active well in matrix gridblocks (layers 1 to Nz/2). An exception here is the "observation" well. An "observation" well may be a fake observation or actual observation well. A fake observation well is an artifice introduced simply for the purpose of printout monitoring of saturations and pressures in its completed gridblocks – i.e. not a physically real (drilled) well. Such a (fake) observation well may have completions in matrix and fractures and no rate (keyword RATE) should be entered for it. That will ensure that no wellbore crossflow computations will falsely affect pressures and saturations. An actual observation well is a physically real (drilled) well and it should have an entered rate of 0, reflecting the fact of wellbore crossflow. Any physical well is an (actual) observation well during every (shutin) time period when its entered rate is 0.

Grid structure and properties

Dual porosity systems are modeled by dividing each fractured reservoir gridblock into a matrix gridblock and a fracture gridblock. This is done in the model by doubling the number of reservoir layers. If the fractured reservoir has n true layers, the value of NZ input on the GRID card is 2n. The first n layers (k=1,Nz/2) represent the matrix and the second n layers (k=Nz/2+1,Nz) represent the fractures. Each matrix gridblock of dimensions Δx, Δy, Δz, contains many matrix blocks of dimensions Lx, Ly, Lz, which are separated by fractures on those spacings. The volume of those fractures is represented in the fracture gridblock.

Grid block array data specified for the first n layers (the matrix subarrays) define the matrix properties, and for the second n layers (the fracture subarrays) define the fracture properties. Matrix and fracture subarrays should be identical for values of DELX, DELY, DEPTH, THICKNESS, INITREG, and PVTTYPE. All other arrays may reflect differences in matrix and fracture properties.

Gridblock volumes and porosities are as follows.

V_m = matrix volume in a matrix gridblock = $(1-\phi_f)\Delta x\Delta y\Delta z$netm
V_{pm} = matrix pore volume in a matrix gridblock = $V_m\phi_m$ = PV(matrix)
V_f = fracture volume in a fracture gridblock = $\phi_f\Delta x\Delta y\Delta z$netf
V_{pf} = fracture pore volume in a fracture gridblock = V_f = PV(fracture)

where

$\Delta x, \Delta y, \Delta z$ = gridblock dimensions in x, y, z directions, ft (from DELX, DELY, THICKNESS arrays)
Δz = gross thickness
Δznetm = matrix gridblock net thickness, ft
Δznetf = fracture gridblock net thickness

ϕ_f = input fracture porosity, fraction
ϕ_m = input (true or physical) matrix porosity, optionally referenced to POROSBASE

Fracture porosity ϕ_f is typically the order of 100 times less than matrix porosity ϕ_m (say .001 vs .1) so initial hydrocarbon in-place in the matrix is typically the order of 100 times larger than initial hydrocarbon in-place in the fractures. As with any use of the term "typically", significant exceptions may occur. In granites, mudstones, and certain carbonates (e.g.chalk), the matrix porosity can be quite small (or zero) and the fracture porosity can be comparable to matrix porosity[35].

Matrix block dimensions

Each matrix gridblock contains many matrix blocks of dimensions Lx by Ly by Lz, ft, which are separated by fractures. Those dimensions essentially represent the average fracture spacings in the x, y, and z directions (since the fracture widths are negligible) and are input using the LX, LY, and LZ arrays. Default values are zero. Zero values of Lx (Ly, Lz) signify the absence of y-z (x-z, x-y) fractures. Lx, Ly, and Lz values are taken only from the first Nz/2 layers of their arrays (the first Nx*Ny*Nz/2 numbers) – the remaining numbers are ignored. The number of matrix blocks in a matrix gridblock is about $(1-\phi_f)\Delta x\Delta y\Delta znetm/LxLyLz$ (replace $\Delta x/Lx$ by 1.0 if Lx is 0 and similarly for $\Delta y/Ly$ if Ly is 0 and $\Delta znetm/Lz$ if Lz is 0).

You do not need to be careful to enter zero Lx, Ly, and Lz values for unfractured grid blocks – the model detects unfractured gridblocks by the PV array (PV(fracture)=0) and Lx,Ly,Lz values are ignored for such gridblocks.

The Lz value for a gridblock is used for two different purposes.

First, Lz is used to internally calculate the VE components of the matrix and fracture VE P_{cwo} and P_{cgo} curves – these determine the equilibrium matrix S_{we} and S_{ge} saturations resulting from increasing fracture water and gas saturations ("immersion" – see Reference 31). For this purpose, the Lz value used is:

If the LZ array is omitted	$Lz = \Delta z$
If entered gridblock Lz is 0	$Lz = \Delta z$
If entered gridblock Lz is > 0	Lz = entered value
If entered gridblock Lz is < 0	$Lz = \|Lz\| * \Delta z$

The VE Pc curves are approximated using Quandelle's suggestion as the sum of the rock Pc curve and the linear gravity-only curve. The latter component and the rigorous VE Pc curve are discussed in Reference 31.

Second, Lz (together with Lx and Ly) is used to internally calculate the matrix-fracture exchange coefficient (transmissibility) Tex and the diffusion transmissibility Texd if they are not entered as array data.

Calculation of matrix-fracture exchange transmissibilities Tex and Texd

If the TEX or the TEXD arrays are not input, their values are calculated internally as follows:

$T_{ex} = 4(0.0011274)\Delta x\Delta y\Delta z netm(1-\phi_f)(k_x/L_x^2+k_y/L_y^2+k_z/L_z^2)$ rb-cp/d-psi
$T_{exd} = 4\Delta x\Delta y\Delta z netm(1-\phi_f)(1/L_x^2+1/L_y^2+1/L_z^2)\phi_m/tor$ ft

where

T_{ex} = matrix-fracture exchange coefficient or transmissibility, rb-cp/d-psi
T_{exd} = matrix-fracture diffusion exchange coefficient or transmissibility, ft
k_x,k_y,k_z = matrix permerbilities in the x,y,z directions, respectively, md (from KX, KY, KZ arrays)
Lx,Ly,Lz = fracture spacings (of y-z, x-z, x-y fractures) in the x,y,z directions, respectively, ft (from LX, LY, LZ arrays)
tor = tortuosity

If Lx is 0, the term involving Lx^2 in the above equations is omitted. Similarly if Ly (Lz) is 0, the term involving Ly^2 (Lz^2) is omitted.

Tortuosity generally lies in the range 1-4 and should be entered as the TOR array. If that array is not entered or a tor value of 0 is encountered, tor is set internally to 2. Porosities should be entered using the POROS array. If that array is not entered, ϕ_m and ϕ_f are internally set to 0.1 and 0, respectively.

Tex and Texd are internally set to zero in gridblocks where PV(fracture) =0. If Tex is 0, Texd is set to 0.

At least one of the TEX, LX, LY, or LZ arrays must be input. If DIFFUSION is input, at least one of the TEXD, LX, LY, or LZ arrays should be input.

Matrix-fracture transfer by convection

The flow rate of phase j from matrix to fracture is

$q_j = T_{ex}\lambda_j\Delta p_j$ rb/d

where j = w, o, g, for water, oil, and gas, respectively, and

λ_j = phase j upstream mobility, k_{rj}/μ_j 1/cp
k_{rj} = phase j upstream relative permeability

$\Delta p_j = p_{jm} - p_{jf}$, psi

p_{jm} = phase j pressure in matrix

p_{jf} = phase j pressure in fracture

The Δp_j are

$\Delta p_w = \Delta p_o - \Delta P_{cwo}$

$\Delta p_g = \Delta p_o + \Delta P_{cgo}$

The matrix and fracture capillary pressures are

$P_{cwo} = p_o - p_w$

$$P_{cgo} = p_g - p_o$$

The P_c values should be the vertical equilibrium (VE) values in order that the matrix equilibrium saturation S_{ge} resulting from immersion in gas (S_{gf}=1.0) and matrix equilibrium saturation S_{we} resulting from immersion in water (S_{wf}=1.0) be correct. That is discussed in detail in ref 31. For given fracture saturations, at equilibrium at least one phase will have a zero Δp value and any phase having a nonzero Δp value will have zero upstream mobility.

Matrix-fracture transfer by molecular diffusion

Matrix-fracture transfer by diffusion includes gas-gas and liquid-liquid diffusion. The matrix-to-fracture diffusion rate for component i is

$$q_{di} = cT_{exd}(D_{liqi}\rho_{om}f(S_o)(x_{im}-x_{if})+D_{gasi}\rho_{gm}f(S_g)(y_{im}-y_{if}))\ \text{mols/d}$$

where

c	= units conversion constant = 16.564
D_{liqi}	= liquid-liquid molecular diffusion coefficient, cm^2/sec
D_{gasi}	= gas-gas molecular diffusion coefficient, cm^2/sec
ρ_{om}	= matrix oil phase density, mols/rb
ρ_{gm}	= matrix gas phase density, mols/rb
$f(S_o)$,f(Sg)	= functions dependent upon matrix and fracture oil and gas saturations
x_i	= oil phase mol fraction
y_i	= gas phase mol fraction

Options

Numerous opinions exist concerning the magnitude and rate of oil recovery from matrix blocks immersed in gas and/or water. Some of those opinions are discussed in detail in Reference 31. Some argue the need for two Lz values, Lzw and Lzg for matrix-fracture transfer resulting from immersion in water and gas, respectively. The supporting argument for two Lz values can involve wettability (water-wet vs mixed or oil-wet in water-oil immersion)[48] or other considerations[49]. Other opinions relate to effects of reinfiltration of oil, capillary continuity across fractures, non-zero capillary pressure in the fractures, fracture aperture (width), and lateral viscous pressure drop.

Two-Lz option

If you want different Lzw and Lzg values used in the matrix-fracture transfer, then enter the DUAL dataline as

DUAL nperm LZWG

The model will then set Lzw equal to the LZ array values of the first Nz/2 layers and Lzg to the LZ array values in layers Nz/2+1 to Nz. The Lz used in the Tex internal calculation is Lzw. If you want a different (third) Lz value used in internal Tex calculation, enter those Lz values as the first Nz/2 layers of the LZTEX array. The Lz values used in internal calculation of Texd are the same as those used in internal calculation of Tex. Note that there is no internal calculation of Tex (Texd) if you enter the TEX (TEXD) array.

Example problems

Example dual porosity and SPM check datasets are available upon request.

Summary

In summary, the recommended procedure for setting up dual porosity cases is:

1. Double the true number of reservoir layers (if n = true number, Nz = 2n).
2. In the array data, specify matrix properties for the first n layers, fracture properties for the second n layers. Fracture values for DEPTH are arbitrary – they are internally set equal to matrix values.
3. Enter the DUAL data, and DIFFUSION data for diffusion in compositional cases. Specify 1 for nperm is entire reservoir is single permeability (with or without unfractured regions), 2 otherwise.
4. Input zero Pcwo and Pcgo capillary pressures in the cap pressure data assigned to the fracture blocks (SWT and SGT or SLT, or KRANALYTICAL)
5. Input your true, physical matrix rock Pcwo and Pcgo in the tables or functions assigned to the matrix blocks.
6. Input the TEX array (and the TEXD array for diffusion), or the LX, LY, and LZ arrays, according to the fracture orientations and average spacings in the reservoir.
7. Values of the dual-specific arrays TEX, TEXD, LX, LY, LZ, TOR are taken from the first n layers only. Values for the second n layers are required but are arbitrary.
8. If nperm was entered as 2, and if transmissibilities were not input, zero the m-m interblock transmissibilities (using MODIFY TX, TY, TZ) between any neighboring pair of dual porosity (single permeability) matrix gridblocks and between any neighboring dual porosity and dual permeability matrix gridblocks.

1.8 Equilibrium Initialization (INITIAL, OBEYPSAT)

Secondary keywords - SORG, OIL, DEPTH, HWC, GOC, PINIT, ZINIT, SOAQ, SGAQ, TIRZERO

Enter all PVTEOS and/or PVTBO pvt data before entering any INITIAL data.

Any number of Initialization Regions (IR) (e.g. layer groups) may be initialized with different woc, goc, composition and pressure. An IR must have a single pvt type. By default, all blocks are assigned to IR 1. If you have more than one IR, use the array INITREG to assign blocks to the appropriate IR's.

The term “initialization” denotes the setting of initial pressure, phase saturations, and hydrocarbon fluid composition at time 0 for all grid blocks in the reservoir. The few data required to uniquely determine those quantities subject to capillary-gravity equilibrium, in addition to pvt and grid descriptions, are:

1. initial pressure at a specified reference depth, or depth to the initial saturated gas-oil contact, if one exists
2. depth to the initial water-oil contact, if one exists
3. reservoir fluid composition or composition vs depth

Normally, initializations will result in equilibrium. The user can test for violation of equilibrium by entering

```
CFL       ! enter for impes to avoid change due to instability
TIME 500
END
```

after the ENDINIT card, making the run, and observing the maximum pressure and saturation changes printed in the Timestep Table. These changes should be very small for cases initially in equilibrium. Datafile init1.dat illustrates such a test.

The first section below describes the initialization keywords and data input. The second section discusses the allowance of specified gas-oil and hydrocarbon-water contacts above, below, and within the depth interval of the Initialization Region (IR). The third section gives numerous examples, including the case where initial composition grades from a lean gas condensate at the top, through the critical point, to an oil at the bottom.

Initialization keywords and data

The reservoir is initialized or equilibrated using INITIAL followed by one or more of the keywords GOC, HWC, PINIT, ZINIT, and DEPTH, normally at the end of Initial Data:

```
INITIAL ireg (SORG) (OIL)

  GOC   goc
  HWC   hwc
  PINIT pinit
  ZINIT zinit
  (SOAQ soaq)
  (SGAQ sgaq)

C  For black oil initialization regions:
  DEPTH    PSATBP or PSATDP or GOR  ! Black Oil IR “depth table” header
  depth    psatbp or psatdp or gor  ! one or more lines as desired

C  For compositional initialization regions:
  DEPTH  ! Compositional IR “depth table” header
  depth     psat(est)   {zi}   !   one or more lines as desired
```

Definitions:

ireg Initialization Region(IR) number, if omitted set = 1

SORG	puts residual oil in gas/oil transition zone for 3 phase cases (not usual)
goc	depth of gas-oil contact (ft)
hwc	depth of hydrocarbon-water contact (ft)
pinit	initial pressure (psia) at reference depth zinit (ft)
zinit	reference depth (ft)
depth	depth (ft), measured vertically, positive downward
soaq	initial oil saturation in aquifer (not usual)
sgaq	initial gas saturation in aquifer (not usual)

For black oil IR's:

psatbp	initial bubble point pressure (psia) at depth
psatdp	initial dew point pressure (psia) at depth
gor	gas-oil ratio (scf/stb) at depth

For compositional IR's

psat(est)	estimated saturation pressure (psat, psia) of the entered composition {zi}
{zi}	composition of reservoir fluid at depth, Nc mol fractions summing to 1.0. The entered zi are normalized by: zi(new) = zi(entered)/sum(zi(entered)) if |sum(zi(entered))-1.0| < .0001. Otherwise a fatal error occurs.
OIL	optional label for first-contact miscible IR, see Section 1.3

The order of the subentries under INITIAL is immaterial.

The term "gas-oil contact" (goc) denotes a *saturated* gas-oil contact with a gas-oil capillary transition zone (Sg > 0 and So ≥ 0) above goc and no free gas (Sg=0) below the goc. Gas-oil capillary pressure Pcgo is 0 at the goc. If the DEPTH table specifies compositions grading from a 'lean' gas at the top, through the critical point, to an oil at the bottom, then the depth of the critical point composition is *not* a gas-oil contact – it is simply the depth of the *undersaturated* critical point composition. See the init2.dat example in Section 1.4.

If GOC and HWC are both entered then goc must be < hwc.

An entered hwc is a water-oil contact if the reservoir hydrocarbon at depth hwc is an oil; it is a gas-water contact if the reservoir hydrocarbon at depth hwc is a gas. The depth hwc is the free water level. If hwc is a water-oil contact, Pcwo = 0 at that depth; if hwc is a gas-water contact, Pcgw = Pcwo+Pcgo = 0 at that depth.

If GOC is entered:

A contradiction arises if capillary pressure data dictate So*Sg > 0 and composition vs depth data give Psat < p at a point (block) in the gas-oil transition zone above GOC. In default mode, the program will honor the Pcgo data, changing the composition by looking downward for the first composition which has a Psat equal to p. If you enter (before INITIAL data)

```
OBEYPSAT
```

the program will instead keep composition unchanged and zero So and set Sg to 1-Sw. Either of the above changes will result in non-equilibrium and give some response upon running with no wells.

If PINIT and ZINIT are entered (see Section 1.2), initial pressure pinit is physical pressure, i.e.:

water phase pressure if HWC is entered and zinit > hwc.
gas phase pressure if the reservoir hydrocarbon is gas at depth zinit.
oil phase pressure if the reservoir hydrocarbon is oil at depth zinit.

Pressures printed in the outfile are gas phase pressures, psia.

Enter only one dataline in the depth Table if composition does not vary with depth. The program does not extrapolate outside the Table depth limits.

In the compositional IR, the program calculates the psat of each entered composition in the depth Table and prints it out, with definition as either dew point or bubble point, (edit/search the outfile for “EQUIL”). The psat(est) values can be approximate. In general, they may differ by 100’s of psia from actual psat.

Omit (do not enter) the depth Table for black-oil two-phase problems - water-oil with Rs=0 or constant or gas-water with rs=0 or constant. See Section 1.4 (init3.dat).

If your compositional reservoir fluid is a dry gas having no dewpoint pressure, enter p_{sat} as 0 in the DEPTH table. No flash or p_{sat} calculations will be done and the reservoir hydrocarbon phase will remain all gas. For an example, enter p_{sat} as 0 in the depth table of spe3.dat; the resulting run shows the significantly larger oil recovery when liquid dropout is neglected.

The OIL label can be specified to invoke one of two methods available for modeling first contact miscibility in compositional oil reservoirs with solvent injection. See discussion of the First Contact Miscibility Option (FCM) in the Fluid and Rock Properties section. Use of the FCM option instead of OIL can significantly improve run efficiency with no loss of accuracy. If OIL appears on the INITIAL dataline (e.g. "INITIAL 1 OIL"), then the reservoir fluid will remain oil with no flash or p_{sat} calculations. Phase behavior analysis should first be performed to verify that this assumption is warranted over the range of expected operating conditions. Entering of the two-phase hydrocarbon region near the producers may or may not cause significant error, but is generally a second-order effect. Differences between test runs with and without OIL or FCM are the best measure of applicability.

The SORG option applies to 3-phase black oil and compositional IR’s only where GOC is entered under INITIAL. If gas-oil capillary pressure is non-zero, a gas-oil capillary transition zone will exist in the hydrocarbon column above the GOC depth. Oil saturation (So) values will decrease to 0 with decreasing depth in the column. If you want So >= Sorg in the gas-oil transition zone above the GOC depth, then put "SORG" on the INITIAL dataline. Oil saturation So above GOC will nevertheless be 0 at any depths (gridblocks) where psat < p. A

"WARNING" message flags blocks where So = Sorg (> 0) could not be implemented because psat < p.

The SOAQ and SGAQ keywords specify initial hydrocarbon saturation in the aquifer. Normally, the initial hydrocarbon saturations in the aquifer are 0 (initial Sw = 1.0). If they are specified, the user should insure that the relative permeability data assigned to the blocks in the region reflect the immobility of these phases, otherwise a non-equilibrium situation will occur.

Note: The user should insure that any pairs of IR's with communication between them (that is, having gridblocks with non-zero TX, TY, or TZ, or non-neighbor connections, between them) are initialized such that they are in equilibrium with each other. In the general case for differing fluid contacts in neighboring regions with communication, this requires the following:

1. Communication exists only between common gas zones, oil zones, or water zones. No two pair of zones should be in communication. There should be no communication between blocks in a phase transition zone with blocks in any other region. For example, if there is communication between the common water zones in regions 1 and 2, which includes pairs of region 1 and 2 grid blocks with depth greater than or equal to max(woc1,woc2), then there can be no communication between any pair of region 1 and region 2 grid blocks above that depth.
2. The pvt type should be the same for pairs of regions with communication in the oil or gas zones.
3. Initial pressures and reference depths should be the same for both regions and specified within the communicating zone. This is not absolutely necessary. Initial pressures can be specified at different depths and in different zones, and adjusted by the user in order to obtain identical pressures at a given depth within the communicating zone.
4. In dual porosity cases, each of the matrix cells (those in the first Nz/2 layers) should use the same equilibration data (initreg) as its corresponding fracture cell (those in the last Nz/2 layers).

In order to automatically zero all neighbor and non-neighbor connections between different IRs, enter the keyword

```
TIRZERO
```

in Initial Data. This does not affect any non-neighbor connections that are internally generated to represent angular closure in radial systems or to represent matrix-fracture transfer in dual systems.

GOC and HWC depths and their relation to the IR depth interval

The gridblocks in a given IR will have depths (at gridblock centers) ranging from zmin to zmax. This zmax-zmin interval is referred to here as the *IR depth interval;* zmin and zmax are printed in the outfile (edit/search for "EQUIL").

Each of goc and/or hwc, if entered, may be above or below or within the IR depth interval, subject to the below-mentioned requirements of pressure communication.

If GOC is entered:

- One of the depths (z) in the depth Table must equal goc.
- Do not enter ZINIT. The program internally sets reference depth zinit to goc.
- Do not enter PINIT. The program sets pinit to the psat of the composition at depth goc, entered or internally calculated from data in the depth Table. In the gas-oil transition zone above the goc, psat will equal initial pressure p and decrease with decreasing depth. In the oil column below the goc, psat will be constant at pinit. In Section 1.4 below: see the spe2.dat example and note the exception where both GOC and PINIT entries are desired and allowed to attain a saturated oil column below the goc.

Do not enter GOC if no goc exists. Do not enter GOC for an IR if a goc exists but is in another reservoir or depth interval which is not in pressure communication with the IR. That is, if GOC is entered, there must be pressure communication between the IR and the goc depth.

Do not enter HWC if no hydrocarbon-water contact exists. Entry of a non-existent hwc or of an hwc far below the IR zmax (giving no capillary transition mobile water in the IR) can cause error in the initialization. If HWC is entered, there must be pressure communication between the IR and the hwc depth.

Examples

For most of these examples, initial composition does not vary with depth and there should be and is only one entry in the depth-composition Table. Note that with only one entry in the depth Table, the value of depth entered in the Table is irrelevant if GOC is not entered. Enter any value representative of reservoir depth.

spe1.dat. Black oil. Oil reservoir.

```
INITIAL 1
  PINIT 4800          !  pinit, psia
  ZINIT 8400          !  zinit, ft

  DEPTH PSATBP
    8000      4000      !   depth, ft    psat, psia
```

spe2.dat. Black oil. Oil reservoir.

Note that GOC is entered here and therefore ZINIT must not be entered and PINIT should not be entered. Initial pressure pinit will be set equal to psat at the goc (3600 psia). The oil column below the goc will be undersaturated with a constant psat (bubble point) of 3600 psia.

```
INITIAL 1
   GOC  9035
   HWC  9209

   DEPTH  PSATBP
```

```
          9035.     3600.    !  depth, ft    psat, psia
```

The constant psat (bubble point) in the oil column below the goc is ‘normal’. However, if you want the oil column there to be saturated, enter the 3600 psatbp as a sufficiently larger value (say 3800) and enter PINIT 3600. The program can reset the saturation pressure at the goc downward from entered psatbp (3800 psia) to 3600 psia but cannot reset it upward from psatbp.

spe3.dat. Compositional, Nc=9. Retrograde gas reservoir.

```
INITIAL 1
  PINIT 3550
  ZINIT 7500
  HWC   7500

  DEPTH    !    depth   psat(est)   z1,z2,z3,….
  7000. 3550. .0121 .0194 .6599 .0869 .0591 .0967 .0472 .0153 .0034
```

spe5.dat. Compositional, Nc=6. Oil reservoir.

```
  INITIAL 1
    PINIT 4000
    ZINIT 8400

    DEPTH   !  depth, ft    psat(est)   zi, i=1,2,…
        8300.  2302.   .5 .03 .07 .2 .15 .05
```

spe9.dat. Black oil. Oil reservoir.

```
  INITIAL 1
    PINIT 3600
    ZINIT 9035
    HWC   9950
    SOAQ  .11851     !  optional

    DEPTH  PSATBP
        9000     3600     !  depth, ft   psat, psia
```

init3.dat. Black oil, two-phase gas-water (rs=0 or rs=constant).

See the datafile init3.dat for the PVTBO black oil pvt data for a two-phase gas-water problem. Omit the Depth Table for black-oil two-phase (water-oil or gas-water) problems. HWC may be entered if an initial gas-water contact exists.

```
INITIAL 1
  PINIT  7000    !       pinit = 7000 psia at zinit
  ZINIT  8400    !   zinit = 8400 ft
```

init2.dat. Compositional, Nc=6. Undersaturated hydrocarbon column with composition strongly dependent upon depth.

An undersaturated hydrocarbon column grades from a retrograde gas at the top, through the critical point, to an oil at the bottom.

The pvt data (compositions vs depth) are from test4.dat, prepared by Pera a/s Norway. Pera tuned a single equation-of-state (EOS) to match lab expansion and other data for numerous samples taken from different reservoir depths.

There are 38 entries (compositions) in the Depth Table.

Composition variation is:

‘lean’ gas at 14764 ft dew point=6252 psia gor=8394 scf/stb bbls/mmscf=119

crit. pt. at 15570 ft psat=6863 psia gor=3400 scf/stb bbls/mmscf=294

oil at 15913 ft bubble point=6624 psia gor=1923 scf/stb

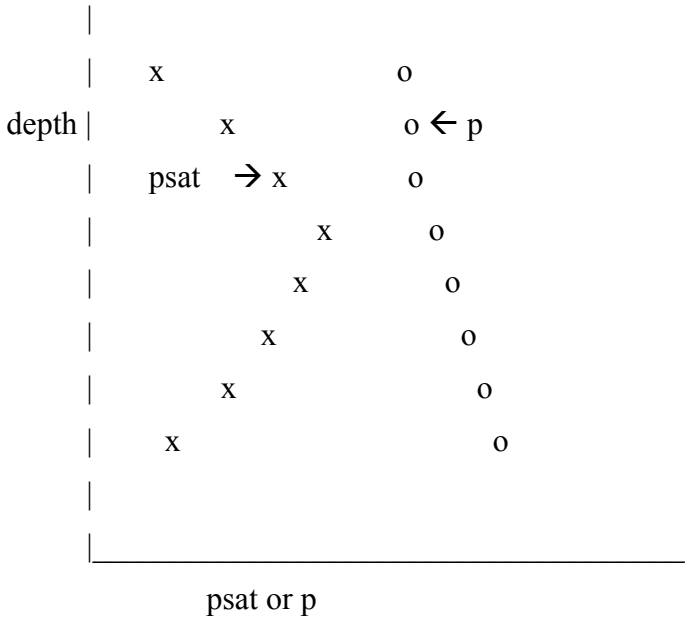
Initial pressure pinit=7000 psia at depth =15255 ft.

HWC may be entered if a hydrocarbon-water contact exists.

```
INITIAL 1
  PINIT 7000
  ZINIT  15255
  HWC 15655     !  water-oil contact

        DEPTH   ! depth, ft    psat(est), psia       composition (zi)

         14764.50   6208.90  ! dew point
         0.69568985E+00 0.15552965E+00 0.79885469E-01 0.49467518E-01
         0.18608471E-01 0.81903848E-03
          ----
         15551.94   6862.70   ! dew point
         0.63036828E+00 0.15455050E+00 0.89655011E-01 0.73376723E-01
         0.45595285E-01 0.64542006E-02

         15584.70   6860.39   ! bubble point
         0.62109317E+00 0.15393684E+00 0.90522080E-01 0.76400539E-01
         0.50024447E-01 0.80229291E-02
           ----
         15912.85   6614.76   ! bubble point
```

```
0.57138394E+00 0.14941005E+00 0.93164772E-01 0.90415277E-01
0.74500335E-01 0.21125627E-01
```

The above data specify an undersaturated hydrocarbon column. Let psat(z) and p(z) denote the saturation pressure and initial pressure, respectively, as functions of depth. Plot depth vs psat and p:



If entered pinit is sufficiently high, the entire hydrocarbon column will be undersaturated. That is normally the case. If entered pinit is sufficiently low, the entire hydrocarbon column will be saturated. For intermediate pinit, there will be saturated and undersaturated portions of the hydrocarbon column.

Let pinit* be the lowest pinit value resulting in no intersection of the above plotted psat and p vs depth curves. There is a conflict or contradiction in the entered data if entered pinit < pinit* - the compositions entered in the depth-composition Table give psat > p for some depth interval(s). For any depth, psat can be ≤ p but psat cannot be > p.

What does the program do if at some depth d it finds p < psat? It flashes the composition at depth d to pressure p to obtain equilibrium oil {xi) and gas {yi) phases. If the psat of the entered composition is a dew point, it stores/uses the gas phase composition {yi}. If the psat of the entered composition at depth d is a bubble point, it stores/uses the oil phase composition {xi}. In either case, the hydrocarbon column is saturated (psat=p) at depth d.

1.9 Coal Bed Methane (COAL)

Sensor can model coal degasification due to depletion in black oil mode. A trace immobile oil phase is used to represent the coal, while the defined porosity represents the seams, cleats and fractures. The gas adsorption isotherm is represented by the gas in solution (R_s) term in the black oil pvt tables assigned to the coal regions, as described by Seidle and Arri[46]. The authors note that in most coals, desorption and diffusion to the cleats and fractures is much faster than the convective flow through them, so the former can be assumed to be instantaneous in the model. The authors discuss the need to modify model porosity, relative permeability, and capillary pressure in order to account for the artificial oil saturation used to represent the coal.

Our recommendation is to use a small oil saturation on the order of 0.001 so that these modifications are not necessary. The R_s values in the PVTBO table are computed from:

$$R_s = 0.17525\ V_m\ d_b\ b\ p\ /\ (por\ S_o\ (1.0 + b\ p))\ scf/stb$$

or

$$R_s = 5.614468\ V_L\ b\ p\ /\ (por\ S_o\ (1.0 + b\ p))\ scf/stb$$

where

V_m	= Langmuir isotherm constant, scf gas/ton coal
V_L	= Langmuir volume, scf gas/cu ft coal = $0.031214\ V_m\ d_b$
d_b	= bulk coal density, gm/cc
b	= Langmuir pressure constant, 1/psi, = 1.0 / Langmuir pressure
p	= pressure, psia
por	= coal porosity, fraction
S_o	= artificial oil saturation, recommend 0.001

The coal gas content V, scf/ton, at any pressure p is given by:

$$V = V_m\ b\ p\ /\ (1.0 + b\ p)$$

Oil formation volume factors (BO) and viscosities (VISO) are entered in the table as 1.0. Surface oil density is immaterial, use a value of 56 lb/cu ft.

Paul et. al.[47] presented 3 benchmark cases for validating coal bed methane simulators. Datafiles for these cases are paul1.dat, paul2.dat, and paul3.dat. Coal properties for cases 1 and 2 are specified as:

Langmuir volume V_L = 28.8 scf/cu ft

Langmuir pressure = 571.0 psia = 1/b

por = 0.02

So the value of R_s in the paul1.dat PVTBO table at p=50 psia, using S_o=0.001, is

$$R_s = 5.614468\ 28.8\ 50.0\ /\ 571.0\ /\ (0.02\ 0.001\ (1.0 + 50.0/571.0))$$

$$= 650952.8\ scf/stb$$

In order to initialize the coal regions in the water column, as is usually the case, add the label COAL to the INITIAL data for those regions, as in the following example from paul1.dat:

```
PVTTYPE ZVAR
  1 2 1
INITREG ZVAR
  1 2 3
SWINIT ZVAR
  .999  .85  .999
INITIAL 1
  DEPTH  PSATBP
   1000.   1000.
  PINIT   1100.
  ZINIT  1007.5
```

```
  COAL
INITIAL 2
  PINIT 1108.
  ZINIT  1027.5
INITIAL 3
  DEPTH  PSATBP
   1000.   1115.
  PINIT   1115.
  ZINIT  1047.5
  COAL
```

In this case, an isolated sandstone layer (layer and initreg 2) exists between coal layers (and initregs) 1 and 3. Since the pvt type for initialization region 2 is black oil gas-water, and no contacts are entered, the region is assumed to lie in the gas column (PINIT is a gas phase pressure, pressures vs depth in region follows gas gradient). If it was desired to initialize the sandstone region in the water column, with a water rather than a gas pressure gradient, entry of HWC at the top of the region would be necessary. Here, it makes no difference since no pressures are being computed by gravity-capillary equilibrium (pressures for each isolated layer are specified at the layer center).

Results for the 3 Paul et. al. cases match the published results.

Note: PVT error override (RUN)

The artificial oil properties (B_o=1) will cause errors to be detected in the PVTBO data ($B_o - R_s B_g < 1$). The keyword RUN should be entered in the Initial Data to override these errors.

1.10 End of Initial Data (ENDINIT) mandatory keyword

The keyword ENDINIT must be entered to terminate the Initial Data, which is the first section of the datafile.

2 Modification Data (MODIFY)

Note: Do not set PV to 0 using the MODIFY PV entry. To set PV of block(s) to 0:
Use MOD on the PV (PVF) array if the PV (PVF) array is entered.
Use MOD on the POROS array if the PV array is not entered.

This second section of the datafile is optional. If used, it starts with the keyword MODIFY, following ENDINIT.

The keyword MOD in Section 1.5 is used to regionally alter grid block property arrays such as permeability, porosity, etc. The keyword MODIFY here is used to regionally alter calculated grid block quantities of pore volume, transmissibility (including dual matrix-fracture and dual diffusive transmissibilities), and block-center depth. If these pore volume or transmissibility arrays are entered as data, do not use MODIFY to change their values (this will result in an error) – use MOD cards (following the individual arrays) instead.

2.1 Standard MODIFY data

```
MODIFY X  f
 i1 i2  j1 j2  k1 k2  @ v  !   region of modification
```

... as many of these lines as desired ...

The X must be one of: TX TY TZ TEX TEXD PV DEPTH TXY1 TXY2 POROS. The operator @ may be any one of + * = < > CUT, as described for MOD in Section 1.5. The > is applied only to nonzero values. The modified POROS values are used only with compaction tables – they do not alter pore volumes (use MOD under the POROS array for that). The factor f is optional and is 1.0 if omitted. Values of X are multiplied by f before applying the @ modification. This can be useful in modifying communication across a fault.

Example:

In a 50x40x10 grid a fault is represented by zig-zag segments. The throw is minor and non-neighbors are neglected. Apart from structure, the only effect of the fault is areal communication across the fault:

```
MODIFY TY  .22              ! fault A3
  12 14   12 12  1 10 * 1.
  15 17   14 14  1 10 * 1.
  18 23   17 17  1 10 * 1.
MODIFY TX  .22              ! fault A3
  14 14   13 14  1 10 * 1.
  17 17   15 17  1 10 * 1.
```

For a large number of segments, only the twice-appearing factor .22 needs changing from run to run.

Transmissibility TX for block i,j,k is the x-direction transmissibility between blocks i+1,j,k and i,j,k. TY is that between i,j+1,k and i,j,k and TZ is between blocks i,j,k+1 and i,j,k (the TRMINUS keyword can be used to define TX,TY,TZ as minus-face values, but this does not affect the following definition of diagonal transmissibilities). TXY1 and TXY2 are diagonal

transmissibilities used only if NINEPOINT is entered. TXY1, stored and printed for block i,j,k, is the diagonal transmissibility between blocks i,j,k and i+1,j+1,k. TXY2, stored and printed for block i,j,k, is the diagonal transmissibility between blocks i,j,k and i+1,j-1,k.

Units are rb for PV, ft for DEPTH, and rb-cp/day-psi for transmissibility.

2.2 MODIFY TFRAC-UNFRAC (TFRAC-UNFRAC)

This option applies to dual porosity and dual permeability cases only (DUAL 1 or DUAL 2). In these cases, Sensor internally and automatically creates non-neighbor connections between any unfractured matrix block and each neighboring fracture block. Previously, there was no means for modification of these internally generated connections. They can now be modified using the TFRAC-UNFRAC label on the MODIFY card. **NOTE** that the designation of connections is different from the specifications of all other MODIFY data, that is, the connections are designated as pairs of blocks i1 j1 k1 i2 j2 k2, rather than as regions defined by the ranges i1 i2 j1 j2 k1 k2. Either k1 must represent a matrix block and k2 a fracture block (k1<= Nz/2, k2>Nz/2), or k1 must represent a fracture block and k2 a matrix block (k1>Nz/2, k2<=Nz/2). Only multiplicative operations are allowed (*):

```
MODIFY TFRAC-UNFRAC
i1 j1 k1 i2 j2 k2  * v
```

3 Recurrent Data

Recurrent Data is the third and last section of the datafile. It generally starts with the keyword WELL. It ends with the keyword END, which is the last keyword in the datafile.

3.1 Well Data

Well keywords

Well data keywords are:

WELL	SOMINPERF	SWMAXPERF	WELLRADIUS	PICALC
PIMAX	PIMULT	BETAMULT	MOBINJ	WELLTYPE
FSWAG	BHPDEFAULT	BHP	BHPDATUM	THP
DRAWDOWN	RATE	RATEMIN	WELLTRACER	INJGAS
ONTIME	WELLSEP	LIMITWELL	OPENWELL	
CONVERT	CYCLETABLE	WAGTBL	PCON	DRILL
RIGS	TDRILL	THPTABLE	FLAT	

Mandatory keywords are WELL and WELLTYPE. When you want the wells to flow, you must enter RATE. Until RATE data are specified for a well, it is shutin with no recirculation, which corresponds to an undrilled status.

Any number of wells may be perforated in a given grid block. In the Impes case, somewhat better stability results if no more than one well is active (flowing or recirculating) at any time in any given grid block. Wells may be vertical or deviated (e.g. horizontal). There are no restrictions on the ijk indices of a well's perforated grid blocks.

Well location and perforation data (WELL) mandatory keyword

Each well must be defined under keyword WELL before data are entered for it under any other keyword. The wells are numbered internally 1,2,3,... in the order they appear under the WELL keyword. This allows abbreviated data entry using a range of well numbers, discussed below. The format of WELL is:

```
WELL (SURFACE)
   h1 h2 ...hn                          ! header line
   wname                                ! wellname
    v1 v2 ... vn                        ! first perf
    v1 v2 ... vn                        ! second perf
    ......
    v1 v2 ... vn                        ! last perf
   wname
    v1 v2 ... vn                        ! first perf
    v1 v2 ... vn                        ! second perf
    .....
    v1 v2 ... vn                        ! last perf
  ... etc. .. all wells ..
```

where

wname = well name, 8 or fewer characters, or = well number, integer of 6 or fewer digits.

SURFACE = optional label, indicating the well is a planar surface well

h1 h2 ... hn = header card symbols, any of:

I J K K1 K2 PI BETA SKIN RW KX KY KZ H SHUTWC SHUTGOR SHUTWGR

Use the optional SURFACE label on the WELL dataline if you want well indices internally calculated for planar surface wells (if the well is to be represented as a planar surface, on or through which it is desired to constrain pressure or rate). The PI's will be calculated as kA/L where, for the case where the surface well is a yz plane, L is dx/2 and A=dy*dz. The normal to the plane of the surface well is the direction of flow (x,y,or z) into the reservoir. If the normal to the plane of the surface well is in the x-direction, enter i negative. If the normal to the plane of the surface well is in the y-direction, enter j negative. If the normal to the plane of the surface well is in the z-direction, enter k negative. Do not enter a positive PI under a WELL SURFACE entry. Entry of a positive PI always dominates any eligible option of PI calculation. That is, if PI is entered positive then the program sets PI to the entered value and does not calculate it.

Do not include any unperforated grid blocks.

The order of the header card symbols is immaterial but the values (columns) entered, v1 v2 ... vn, must correspond to the header line symbols. The header line is remembered - you only need to enter it once in the entire datafile, immediately after the first WELL entry, if your WELL data columns are always the same.

You can reenter a well under WELL any time to change its perf data.

Wellbore radius is used only if well indices are internally calculated or if $\beta > 0$. The r_w value is initialized (default) by the program as .25 ft for all wells. A global change of wellbore radii can be made by entering

```
WELLRADIUS
-4   rw    !  set wellbore radii to rw for all wells
```

before entering WELL. The r_w values set by a WELLRADIUS entry are not used in pi calculations of perfs entered under any previous WELL entry. Normally, for a given well, r_w and skin values are the same for all perfs.

The point of fluid withdrawal (producers) or of fluid injection (injectors) in the wellbore (where bhp applies) is presumed to be the first perf, and is referred to as the bhp perf or the pump perf. If you want some other perf to be the withdrawal or tubing point, enter k of that perf negative.

The turbulence treatment in gas injection wells and all production wells applies only to the gas phase. Appendix 7 describes the turbulent flow equations and gives the Peaceman formulae for well indices. Peaceman[30] discusses the accuracy of his formulae for horizontal wells. There is an additional option for entering turbulence, β. Engineers conduct isochronal well tests and determine a turbulence-related parameter D from the test results. Appendix 7 gives the definition of this parameter D. For Sensor, if β is entered negative then its value is that of D as opposed to a multiple of the Katz et al turbulence factor. For an example of the use of the turbulence option, see datafile turb.dat.

The header line symbols I, J, and K (or K1 K2) are mandatory. All the other header line symbols are optional. The term "omitted" used below refers to header line symbols.I J K or K1 and K2 mandatory i,j,k indices of the perforation(s)

Productivity index (PI)

If PI is omitted or if v=0:

PI is calculated internally from Peaceman[25] or Shiralkar[26-28], for the 5-point and 9-point difference schemes, respectively. The calculated value accounts for anisotropy, if present. If KX,KY,KZ,H are omitted or 0, the required values of k_x,k_y,k_z,h (wellbore length in the perforated gridblock) are taken from internally stored data. If not, k_x,k_y,k_z and h are the values entered in the KX,KY,KZ,H (md,feet) columns. The wellbore orientation within the block in the case where PI is calculated is:

i j k	direction
all positive	vertical
i negative	x-direction
j negative	y-direction

If v is < 0, PI is calculated as above and multiplied by the absolute value of v.

If v is > 0, PI = v, rb-cp/day-psi

Turbulence factor (BETA)

Omitted beta = 0

v beta is set equal to v (nonnegative)

beta = multiple of Katz et al Beta turbulence factor[29]. If BETA is entered, the program needs to know the direction of the wellbore in the block, as described above for PI.

Skin factor (SKIN)

Omitted skin is 0 (used in calculation of PI)

v skin = v

Wellbore radius (RW)

Omitted

Wellbore radius r_w (feet) is the default (.25 feet) or previously entered value. The value of r_w is used in β calculations and in calculation of PI.

v if v > 0, r_w = v, ft

if v = 0, r_w is the default or previously entered value

Shut in water cut (SHUTWC)

The integers entered in this column are denoted by nshutwc. In the event of a workover for watercut or water rate, layers (perfs) having the same value of nshutwc (> 0) are shut in as a group (see keyword LIMITWELL below).

Omitted or v=0: nshutwc is 0 for the perf.

v > 0 nshutwc = v.

Shut in gor (SHUTGOR)

The integers entered in this column are denoted by nshutgor. In the event of a workover for gor or gas rate, layers (perfs) having the same value of nshutgor (> 0) are shut in as a group (see keyword LIMITWELL below).

Omitted or v=0: nshutgor is 0 for the perf.

v > 0 nshutgor = v.

Shut in water/gas ratio (SHUTWGR)

The values entered in this column are denoted by nshutwgr. In the event of a workover for water/gas ratio, layers (perfs) having the same value of nshutwgr (> 0) are shut in as a group. (see keyword LIMITWELL below)

Omitted or v=0: nshutwgr is 0 for the perf.

v > 0 nshutwgr=v

Default values of nshutwc, nshutgor, and nshutwgr are 0. Entry of nonzero values will reduce recovery but will reduce the frequency of workovers.

Perforation Criteria (SOMINPERF, SWMAXPERF)

These keywords can optionally be used to establish criteria for completing entered perforations. A gridblock will not be completed if

```
oil saturation   < somin
water saturation > swmax
```

The recurrent data entries are

```
SOMINPERF  somin  !  (default = 0.)
SWMAXPERF  swmax  !  (default = 1.)
```

Perforations that are not completed as a result of these criteria can be observed by searching for "*NOTE" in the output file. The criteria apply only to wells defined after their entry.

Example:

Well indices J are entered and turbulence is 0.

```
WELL
  I   J  K   PI    ! header line
W1
  16  7  3  .7159   ! rb-cp/day-psi
  17  8  4   .6637
  17  8  5   2.4167
  17  8  3   11.071
W2
  25 10  3   .5996
  25 10  4    1.679
  25 10  5    2.397
```

```
  26 10  3    0.835
  26 10  4     2.338
  26 10  5    15.32
  27 10  5    23.038
W3
  21 18  1    0.114
  21 18  3    .160
  21 18  4    .526
  21 18  5    .623
  20 19  3    .168
  20 19  4    .230
  20 19  5    .402
```

... etc. ....

Example:

Well indices are calculated internally for all perfs. All skin and β values are 0 but wellbore radii are .21 feet for all wells:

```
WELLRADIUS
 -4   .21      ! wellbore radii = .21 feet for all wells
WELL
    I J K      ! header line
  wname
   i j k       !   .
    ...        !   .
   i j k       ! all perfs for well wname, calculate PI
    ...        !   .
   i j k       !   .
  ... etc., all wells and perfs
```

Note that the WELLRADIUS entry must precede the WELL entries to set the r_w prior to their use in calculating well pi's.

The header symbols K1 and K2 are useful in problems where a well has many layers and the perforation data are the same for all layers. The following example has two wells completed in all 80 layers of a 300x1x80 xz cross-section, where the well indices are to be calculated internally.

```
WELL
       I J   K1 K2     !  layers K1 - K2 inclusive
 INJR
       1 1    1 80
 PRODR
      300 1   1 80
```

Other admissible formats relating to "wname" under WELL are:

```
WELL
```

```
        h1 h2 ... hn                        ! header line
 wname  v1 v2 ... vn                        ! first perf
        v1 v2 ... vn                        ! second perf
         ......
        v1 v2 ... vn                        ! last perf
 wname  v1 v2 ... vn                        ! first perf
        v1 v2 ... vn                        ! second perf
         .....
        v1 v2 ... vn                        ! last perf
         ... etc. .. all wells ..

WELL
       h1 h2 ... hn                  ! header line
 wname v1 v2 ... vn                  ! first perf
 wname v1 v2 ... vn                  ! second perf
        ......
 wname v1 v2 ... vn                  ! last perf
C            next well
 wname v1 v2 ... vn                  ! first perf
 wname v1 v2 ,,, vn                  ! second perf
       .....
 wname v1 v2 ... vn                  ! last perf
  ... etc. .. all wells ..
```

These alternative placements of wname can be mixed under a given WELL entry.

Well Range Option and Negative Well Number Option

Throughout the following, we will speak interchangeably of "well number" and "well name" wname. This wname, entered under WELL, can be either a number or a name. Normally, all wells are designated by names or all wells are designated by numbers. A well name must start with a letter and contain 8 or fewer characters. A well number must contain 6 or fewer digits. The program internally adds a leading "W" to any well number. For example, if you enter a well number 1055, the printout will list that well as well W1055. Do not use any keywords as well names.

Most of the remaining well data keywords involve data entry in the form of "wname", "wname v", or "wname v1 v2". The well number(name) wname may be entered as a single number (name) or as a range "wname1 - wname2". The dash must have one or more blanks on each side. This is illustrated in examples below. This option cannot be used with the keyword WELL.

If you have used names for some wells and numbers for others then any well range "wname1 - wname2" must be homogeneous - i.e. both wname1 and wname2 must be names or both must be numbers.

In our examples, with wells W1-W26, entered in order W1,W2,...,W26, W3 - W7 is an admissible range, specifying wells W3 through W7, inclusive, but the range W20 - W17 is inadmissible.

For all well keywords except WELL and WELLTYPE, wname optionally can be entered as a negative integer between -1 and -4:

wname	assign the entered data to:
-1	all production wells
-2	all water injection wells
-3	all gas injection wells
-4	all wells

These options are illustrated in examples below. Note that integers -1 to -3 require knowledge of welltype. Therefore, any new WELLTYPE data should preceed any well data using this option with integers -1 to -3.

Wellbore radius (WELLRADIUS)

This keyword can be used to reset wellbore radii without having to re-enter the WELL table.

```
WELLRADIUS
   wname  rw   ! set wellbore radius to rw for well wname
   ... as many lines as desired ...
```

There is order dependence here. The r_w values set by this entry are not used in pi calculations for any perforations entered under previous WELL entries.

Alteration of well productivity indices (PICALC, PIMAX, PIMULT)

The keyword PICALC makes it easy to switch from entered well productivity indices J to internally calculated indices. It avoids a lot of retyping otherwise necessary to change from entered to calculated J when there are many perfs. The entry PICALC v entered before WELL replaces all entered J by internally calculated J * v. If v is positive, it is internally zeroed after the recurrent data frame in which it was entered. If v is negative it is retained permanently - i.e. until reentry.

With negative skin, some calculated well indices J may be very large (or even negative, which triggers an error message). The entry PIMAX v imposes a maximum value of v for all well indices J, entered or internally calculated. The default value of PIMAX is 1.E6 rb-cp/d-psi.

The entry

```
PIMULT
  wname  v
  wname1 - wname2  v
  -1  v                  ! all producers
```

multiplies all perf indices J of the specified well(s) by v. Enter as many lines as desired under the keyword. It does not apply to any perf indices entered or calculated later (after or below the PIMULT entry).

Alteration of turbulence factors (BETAMULT)

The entry "BETAMULT v" imposes a global multiplier v on all beta values entered under WELL. This is convenient in some cases where turbulence factors are important in matching history. If there are many wells, it can become laborious to retype all entered beta values from one history match run to the next. This entry will not affect beta values under any WELL entries preceeding the BETAMULT entry.

Endpoint mobility option for injectors (MOBINJ)

Calculation of injection well rates requires a grid block mobility. In default mode the model uses total mobility, i.e. the sum of grid block phase mobilities. If you wish to use end-point mobilities then enter

```
MOBINJ 0
```

Default value is 1 for total mobility. End-point mobility is the mobility of the injected phase when other phase saturations equal the lesser of their current and residual saturations.

Well type (WELLTYPE) mandatory keyword

This keyword is used to specify the welltype for each well. Well name wname can be entered as a single name or as a range.

Example:

```
WELLTYPE                !  define the welltype unit of each well
  W1 - W17   STBOIL     ! wells W1-W17 are stb/d oil producers
  W10        STBWATINJ  ! well W10 is a stb/d water injector
  W18 - W23  MCF        ! wells W18-W23 are mcf/d gas producers
  W20        STBLIQ     ! well W20 is a stbliq/d producer
  W24 - W26  MCFINJ     ! wells W24-W26 are mcf/d gas injectors
  W25        SWAG       ! well W25 is a mcf/d+stb/d SWAG injector
     .. as many lines as necessary to define all wells ...
```

There are five injector and ten producer welltype units as follows:

welltype	Integer equivalent
SWAG	-5
RBWATINJ	-4
STBWATINJ	-3
RBGASINJ	-2
MCFINJ	-1
STBOIL	1

```
MCF          2
MCFWET       3
STBLIQ       4
RBTOT        5
RBOIL        6
RBGAS        7
RBWAT        8
RBLIQ        9
STBWAT      10
```

These are also listed in Table 12. The integer equivalent may be used in place of the welltype unit in the dataline. Wetgas rate (MCFWET) is mols hydrocarbon/day, expressed as Mcf/day. The gas injection well (MCFINJ or RBGASINJ) is used to inject oil and/or gas. The injected stream composition is specified by the user (INJGAS), and that composition may be a gas or an oil or any mixture at reservoir conditions. SWAG (simultaneous water and gas) injection well rate is qg mcf/d gas + qw stb/d water, and is additionally defined by the required FSWAG data to define gas fraction for each SWAG injector.

Wells defined as water or gas injectors may essentially be retyped as WAG (alternating water and gas) injectors using the WAG option described in a following section. In that case, the well type specified here indicates the initial type for the first cycle.

If a well is multiply specified, e.g. wells W10, W20, and W25 above, the last entry applies. Welltype can be changed. Any type of well can be changed to any other type of well at any time. In general, this type change involves other data changes, such as for bottomhole pressure and rate constraints.

Well block relative permeability modification for producers

For producers, phase relative permeabilities used in the well production equations can be scaled by specified constant factors for each well. Any entries for injectors are ignored.

The welltype dataline under WELLTYPE is normally

```
wname  welltype
```

Relative permeability multipliers for the well can be optionally specified as

```
wname  welltype  fkrw fkro fkrg
```

where the three f values:

- are used for production (not injection) wells (entries for an injection well are ignored)
- are applied to all perfs of the production well
- if entered, must be ≥ 0
- if entered, may exceed 1, but the resulting production well perfs' rel perms are still limited by the max krmax values (1.0 unless entered otherwise)
- default to 1.0 if not specified

The feature requires wname to be alphanumeric (not numeric).

This feature has no effect on the phase mobilities used in interblock flow, neighbor or non-neighbor.

As usual, input data are remembered and need not be repeated unless a change is desired.

Injected gas fraction for SWAG wells (FSWAG) mandatory for SWAG injectors

The gas fraction of the injection stream for simultaneous water and gas (SWAG) injectors must be specified as

```
FSWAG
  wname  fg  ! assign well wname a SWAG gas fraction of fg
             ! as many lines as required
```

where

```
fg  = qg / (qg + qw), 0 <= fg <= 1.0
qg  = gas injection rate, mcf/d
qw  = water injection rate, stb/d
```

Maximum rates entered for SWAG injectors in the RATE data are defined as qg + qw.

Default limiting bottomhole pressure (BHPDEFAULT)

The program default values of bhp for all wells are as follows:

bhppr	=	500 psia	for producers
bhpgi	=	20000 psia	for gas injectors
bhpwi	=	20000 psia	for water injectors
bhpswag	=	20000 psia	for swag injectors

Until bhp is entered under BHP for a well,

the well's bhp value will be in accordance with its welltype and the default bhp values.

If the well's type is changed using WELLTYPE (for example, from a producer to an injector, or conversely), then the program will change the well's bhp in accordance with its new type and the default bhp values.

To change these default bhp values, use the following one-line Recurrent Data entry (example values):

```
BHPDEFAULT  3200  13000 9000 10000  ! bhppr,bhpgi,bhpwi,bhpswag
```

These values are not used for any well assigned to a tubinghead pressure table.

This keyword can be used to simplify BHP data in the following case. For the entirety of the run, producer, gas, water, and swag injector bhp values are to be (say) 2000, 13000, 10000, and 11000 psia, respectively. Enter at the beginning of Recurrent Data (before any WELL entry)

```
BHPDEFAULT  2000  13000  10000 11000
```

and DO NOT enter BHP anywhere in the datafile. You can enter WELLTYPE data which change producers to injectors, and conversely, without any need to change or enter BHP data.

If this BHPDEFAULT entry is entered later in Recurrent Data, then the bhp values for all wells which (a) have been entered previously under WELL at the time of this later BHPDEFAULT entry, and (b) have not had bhp values entered under BHP, will be changed in accordance with the values on this later BHPDEFAULT dataline.

Limiting bottomhole pressure (BHP, BHPINC)

Keyword BHP assigns limiting bottomhole pressures, bhp, to wells:

```
BHP
  wname  bhp  ! as many lines as desired
```

By default, the specified bhp applies at the bhp or pump perf (the withdrawal or tubing entry point for a producer, or the point of entry of fluids into the perforated region for injectors) defined or defaulted (to the first perf) in the WELL perforation data. If the bhp is referenced to some other datum depth, BHPDATUM data are required prior to the BHP specification.

The BHPINC keyword is used to increment values of bhp previously entered using BHP, i.e. bhp (new) = bhp (old) + bhpinc. The incrementing is performed one time at the time of BHPINC entry.

```
BHPINC
 wname  bhpinc  ! as many lines as desired
```

Example:

```
BHP                ! all pressures are psia
    -4      1200   ! bhp = 1200 for all wells
    -1      1400   ! bhp = 1400 for all producers
    -2      7000   ! bhp = 7000 for all water injectors
    -3      8000   ! bhp = 8000 for all gas injectors
 W1 - W20   1100   ! bhp = 1100 for wells W1-W20, inclusive
 W13        1200   ! bhp = 1200 for well W13
 W21 - W26  7500   ! bhp = 7500 for wells W21-W26
 W24        8000   ! bhp = 8000 for well W24
```

Default BHP values are 0 for production wells and 20000 psia for injection wells.

Datum depth for bhp (BHPDATUM)

This entry allows definition of entered well BHP pressures as datum values which apply at a datum depth. If BHPDATUM is not input, entered BHP pressures apply at the center of the bhp or pump perf, defined (by negative k) or defaulted (as the first perf) in the well perforation data. We recommend that BHPDATUM not be used, or at least that it not be used to correct pressures over significant distances (hydraulics tables are much more appropriate), as corrected pressures have questionable meaning and accuracy. If entered, BHPDATUM data specify for each well type a datum depth and a fixed gradient to use for the correction to the bhp or pump perf, as in the following example.

Example:

```
C           datum  gradient
C             ft    psi/ft
BHPDATUM
             7200   .4374   ! water injectors and water producers
             5300   .1232   ! gas injectors and gas producers
             5600   .3177   ! all other producers
             6100   .2887   ! swag injectors
```

The entered bhp value for each well is corrected by

bhp = bhp(entered) + gradient * (Z - datum)

where Z is depth to the center of the bhp or pump perf indicated by the well perforation data (default 1st perf). If entered, BHPDATUM must be entered prior to BHP entry. It will not affect bhp values of previously entered bhp data. This is an optional keyword. If not entered, no internal adjustment of entered bhp values is performed – that is, the entered bhp applies at the well's pump perf.

To deactivate previously entered well bhp datum data, use the one-line entry:

```
BHPDATUM 0
```

Note: If BHPDATUM data cause the bhp of a producer to be less than 10 psia, it is internally reset to 10 psia.

Limiting tubinghead pressure (THP)

Keyword THP assigns wells to tubinghead pressure tables and specifies limiting tubinghead pressures for the thp wells.

```
THP
  wname thp n  ! as many lines as desired
```

This assigns well wname to tubinghead pressure table # n, and specifies limiting (minimum for producers, maximum for injectors) tubinghead pressure = thp, psia, for the well. Wells can be subject to BHP control or to THP control, but not both. If you wish to change a well back to bhp control after it has been assigned to a thp table, then enter its bhp using BHP.

Normally, oil production wells would be assigned to oil thp tables and gas production wells to gas thp tables. However, the code allows a well to be assigned to either type of table. In particular, for near-critical gas condensate problems, all wells might be assigned to gas thp tables even though the initial, depth-dependent composition gives rich gas in the upper structure and volatile oil in the lower structure.

Injection wells may be assigned only to injection thp tables.

Example:

```
THP
 W11  -  W20    600  1    !  thp  =  600  psia  for  wells  W11-W20,
                  ! use thptable 1
  W2 - W10  700 2  ! thp = 700 for wells W2-W10, use thptable 2
  W1        750 3  ! thp = 750 for well W1, use thptable 3
  W4        720 3  ! thp = 720 for well W4, use thptable 3
```

If the last entered limit pressure for a well is a thp then bhp is not used and conversely.

The THP entry can be used for an additional purpose as follows. In history match periods, wells are usually placed on bhp control. However it may be desired to obtain calculated wellhead (tubinghead) pressures (p_{th}) for comparison to observed wellhead pressures. If n is entered negative, for example,

```
THP
  W7  400 -2  ! use tubinghead pressure table 2 for well W7
              ! the value 400 here is immaterial, not used.
```

then the well rate remains calculated in bhp control mode but the wellhead pressure (p_{th}) is calculated from the tubinghead pressure table. The calculated values of p_{th} are printed in the Well Table and in the eor well summaries, for all wells having n not equal to 0. The calculated p_{th} is only approximate if it exceeds the minimum or maximum thp parameter of the tubinghead pressure table, and can even be negative. Sensor does not extrapolate the pressure difference given by the tables.

For shutin (0-rate) injectors or oil production wells, p_{th} is not calculated and is printed as 0. For shutin gas production wells assigned a positive or negative n, p_{th} is calculated at 0 wcut, 0 liquid-gas ratio, and at the lowest (first) rate entered in the tubinghead pressure table n. Note that the program by default does not include 0-rate lines in the eor well summaries. If you wish to include them, use the PRINTZERO 1 entry discussed below.

Drawdown limit for a producing well (DRAWDOWN)

Drawdown for a producing well is the grid block-wellbore pressure difference. Each perf of a well has a different value of drawdown. Maximum drawdown may be specified as

```
DRAWDOWN
  wname  pd0 pdp  (PUMP) ! as many lines as desired
```

where

```
drawdown limit pd (psia) = pd0 (psia) + pdp (dimensionless) * Pi
drawdown                 = Pi – Pwi, at each perf i, psia
Pi                       = gridblock pressure for perf i, psia
Pwi                      = flowing bottomhole pressure at perf i, psia
PUMP                     = optional label, indicates to apply limit
                           only at pump perf (the perf at which bhp
                           applies)
```

The entry pdp is optional, default value is zero. Its inclusion is based on the paper SPE/ISRM 78150:

> Rhett, D.W. and R. Risnes, "Predicting Critical Borehole Pressure and Critical Reservoir Pore Pressure in Pressure Depleted and Repressurized Reservoirs", presented at the SPE/ISRM Rock Mechanics Conference, Irving, TX, 20-23 Oct. 2002.

The authors state: "In the general case, pore pressure depletion and drawdown combine to increase the effective … stress and reduce the well bore support of the bore hole wall until the shear strength of the rock is exceeded, resulting in rock failure and fragmentation."

If you have entered a drawdown limit for a well and later wish to deactivate it, enter a value of 0 for pd0, and omit pdp. The status symbol DD is used in the Well Table and end-of-run well summary printout to indicate a well is producing on drawdown limit. Drawdown pressure limits can be used for both bhp-constrained and thp-constrained production wells.

When a well is producing on drawdown constraint, drawdown will equal the drawdown limit at one of its perfs and will be less than drawdown limit at each other perf. If you put PUMP on the wname dataline, Sensor will constrain drawdown to the drawdown limit only at the pump perf - drawdown may exceed the drawdown limit at other perfs. The bhp or pump perf is the perf at the point of fluid withdrawal in the wellbore (and the perf where bhp constraints apply) and is by default the first perf defined in the WELL data. If you want some other perf to be the bhp or pump perf (withdrawal or tubing point) enter k of that perf negative in the WELL data.

Example:

```
DRAWDOWN
 -1       180     ! pd0 = 180 psi for all producers
 W3 - W9  120     ! pd0 = 120 psi for wells W3-W9, inclusive
 W12      -80     ! pd0 =  80 psi for well W12
```

The one-line entry

```
DRAWDOWN ALL pd0
```

sets pd0 for all production wells, including those entered after the time of this entry. For an example run, enter a pd0 of 25 psi in spe3.dat. It is interesting to see that this drawdown entry cuts total gas production by a factor of 2 but increases oil recovery. Simulations rarely show an adverse effect of rate on recovery.

Maximum well rates (RATE, QMINUS)

Maximum rates for injection and production wells are specified as

```
RATE
 wname  rate      ! as many lines as desired
```

or as

```
RATE wname
rate1 rate2 ...   ! where wname is a range
```

The units of rate are those associated with the well type (WELLTYPE), which when entered must precede any well's rate specification in any given time period. The second format can shorten datafiles considerably when extended-period history match data with regular (e.g. monthly) rate changes are entered.

A third format can be used for water injectors to specify a maximum water surface rate as a fraction of total field water production:

```
RATE
wname FRAC frac    ! qmax = frac*field water production rate, stb
                   ! as many lines as required
```

This constraint applies in addition to any rate constraint specified for the well using one of the first two formats.

In a fourth format that may be useful for producers in history matching runs, all three observed historical surface phase rates are specified – oil (qo, stb/d), gas (qg, mscf/d), and water (qw, stb/d):

```
RATE
 wname  qo qg qw      ! as many lines as desired
```

The rate constraint to be applied is determined by the well type (WELLTYPE), which when entered must precede any well's rate specification in any given time period. The well type must be one of STBOIL, MCF, STBLIQ, RBTOT, or STBWATER. If the producer well type is RBTOT, then Sensor converts the three surface rates to an approximate rb/d reservoir voidage rate at each iteration of each timestep. The voidage rate will be correct (equal to that in historical reality) only when the reservoir description is accurate and the history match is correct, i.e. only when predicted reservoir pressure and well surface rates are all equal to their historically observed (input) values. Entry of three rates for a RBTOT producer during a prediction period makes no sense (in that case, simply enter the desired single rb/d rate using one of the first two formats).

Until a RATE entry is specified for a well, it is shutin with no recirculation, corresponding to an undrilled status. Entering the first single-rate RATE value or the sum of any three-rate RATE values as greater than or equal to zero at a given time indicates that the well was drilled at that time, and the program will then allow the well to crossflow. If a single RATE entry of zero or a zero sum of any three-rate entry is specified for a well, by default the well will be shutin with recirculation (wellbore crossflow). If it is desired to cement the well in (to not allow crossflow when a well is shut in), enter a single rate value of -1 (or any negative value) for the rate entry in the RATE data. This option is rarely needed and should be avoided unless the well is actually cemented in, or unless multiple defined wells represent a single physical well (those that are not active at any given time should be cemented in). If you do not want any well with a specified rate of 0 to recirculate, then prior to any rate specifications of zero, enter

```
QMINUS 1
```

An entry of

```
QMINUS 0
```

will redefine any subsequently entered 0 rate for any well as shutin with recirculation.

Example:

```
RATE
      W1  1246.    ! rate = 1246 for well W1
 W2 - W6  1877.    ! rate = 1877 for wells W2-W6, inclusive
     -3   9000     ! rate = 9000 for all gas injectors
    W17    0.     ! shut in well W17 with recirculation
    W12   -1      ! shut in well W12 with no recirculation

 RATE  W4 - W12
  136  3.1  1410  712  507  893.4  0  195.6  770.9

 RATE
 W10 FRAC 0.1     ! max water injection rate =
```

```
                    ! 0.1*field water production rate, stb/d
```

Minimum well rates in RATE data (RATEMIN)

Economic limits for production wells, i.e. minimum oil or gas production well rates below which a well is to be shut in, can be specified in the data for the LIMITWELL option, which is described in a following section. The RATEMIN option applies instead to the entered single-rate RATE data and the sum of any entered three-rate RATE data for both injection and production wells.

Small historical well rates may sometimes appear as data, yet have little or no effect on the results. These small rates may be zeroed using the RATEMIN keyword:

```
RATEMIN qmin    ! default is zero (not applied)
```

All positive well single rates entered under RATE following the RATEMIN entry that are less than or equal to qmin will be set to zero. All three-rate entries entered under RATE following the RATEMIN entry are zeroed if their sum is less than or equal to qmin.

Injection well tracer fractions (WELLTRACER)

This keyword entry is required only if TRACER was entered in Initial Data.

```
WELLTRACER
   wname  cpt   !  wellname, traced component name
     {fj}       !  tracer fractions in the injected fluid
```

Example:

```
WELLTRACER          ! if tracers are used
W12 - W15  GAS      ! tracer fractions for traced component GAS
1. 0. 0.            !  = (1,0,0) in injected gas of wells W12-W15.

W16  WATR         ! tracer fractions for traced component WATR
   0 1 0          !    = (0,1,0) for water injection well W16.
-3   C1N2         ! tracer fractions for traced component C1N2
   0. 1.          !   = (0,1) for all gas injection wells.
```

Injected gas composition (INJGAS, INJECT)

Normally, the INJGAS entry is used only for compositional gas injection wells. The entered composition $\{y_i\}$ may be gas or oil or any mixture at reservoir conditions.

```
INJGAS           !  specify injection gas compositions
   wname         !  wellname
   {yi}          !  composition, nc mol fractions
   ... as many pairs of lines as desired/needed ...
```

The default injection composition is as follows: The initial hydrocarbon in place is flashed to surface and the default injection composition is that of the resulting surface gas. The compositions entered under INJGAS are used for gas injectors which are not in platforms. Gas injectors assigned to platforms do not need or use these compositions because the program internally calculates them. The injected gas composition for a gas injection well which is in a platform is:

1) calculated from the compositions of the platform surface produced gas, transfer gas, and outside gas, if YPLAT is not entered for the platform, or

2) the composition entered by YPLAT.

If

```
INJECT 1
```

is entered then the program assigns the composition of field total produced gas to all gas injection wells and INJGAS is not needed or used. The entry INJECT 0 deactivates a previous entry of 1. The default value for INJECT is 0.

The following disucussion applies only to three-phase black oil problems - i.e., it does not apply to gas-water or water-oil two-phase black oil problems. Surface oil is component 1 and surface gas is component 2. If it is desired to inject some mixture of oil and gas, rather than gas, then enter

```
INJGAS           !  specify injection gas compositions
   wname         !  wellname
   yoil          !  oil fraction
   ... as many pairs of lines as desired/needed ...
```

The composition of the injection stream is defined by yoil = qo/(qo+qg) = OGR/(OGR+1) = rs/(1000+rs), where qo STB/d and qg MCF/d are oil and gas injection rates, respectively, OGR is oil-gas ratio qo/qg, and rs is STB/MMSCF.

The Recurrent Data INJGAS entry allows all the following:

```
INJGAS
  GINJ1       !  wellname
    yoil
  GINJ5
    yoil
  GINJ7 - GINJ12
    yoil
  -4          ! all wells (entered previously or later)
    yoil
  -3          ! all gas injection wells (entered previously or later)
    yoil
```

If no Recurrent Data INJGAS entry is used, the default is yoil=0, i.e. 100 % gas.

To inject gas (i.e. no oil) enter yoil=0. (This is default and need not be entered except to change a previous entry).

To inject oil (i.e. no gas) enter yoil = 1.

For rate constrained injectors, the rate constraint introduces a second relationship between qo and qg which depends upon the units specified for the injection well. If units of MCFINJ are specified then the second relation is $q_o + q_g = q$ mcf/d, where q is the specified injection well rate. If units of RBGASINJ are used then the second relation is $q_oB_o + q_gB_g = q$ rb/d, where B_o rb/stb and B_g rb/mcf are the current grid block oil and gas formation volume factors, respectively; that is, a total (oil + gas) of q rb/d will be injected.

Example:

```
WELLTYPE
  INJ  MCFINJ
INJGAS
  INJ
    .2
RATE
  INJ 40000
```

The well will inject q_o = 8000 stb/d and q_g = 32000 mcf/d.

Well ontimes (WELLONTIME)

In some history match cases, the dataset chokes the model by specifying small time steps defining the times at which wells come onstream or go offstream. This can unnecessarily increase cpu.

The fraction of the time in which production or injection wells are onstream may be specified as

```
WELLONTIME
  wname  v     ! assign well wname an ontime fraction v
               ! as many lines as required
```

Rates for the wells in the RATE data are then entered at their physical (average) values during the time that the wells were flowing.

We believe this keyword is not needed in most cases and recommend against its use - i.e. we recommend simple entry of (e.g.) 30-day average well rates.

Notes:

1. Ontime values apply until changed by a later entry.
2. Entered ontimes should be > 0 and <= 1.0, but a value > 1.0 is accepted.
3. Any entered ontimes <= 0 are ignored.
4. Do not use ontimes of 0 to shut wells in – use RATE instead.
5. Do not specify well ontimes for wells in non-cosmetic platforms – an entered value other than 1.0 is a fatal error. WELLONTIME and Platform ONTIME cannot be used together. Use neither, or one or the other.

6. If, as normally, all entered well rates are average values for the time period, do not enter WELLONTIME.

See Appendix 9 for additional discussion of ontime factors.

Assign wells to separators (WELLSEP)

This optional entry is used only for compositional runs. If WELLSEP is not entered, the program by default assigns all wells to separator #1.

```
WELLSEP
  wname  n     ! assign well wname to surface separator n
               ! as many lines as required
```

Production well limits and workovers (LIMITWELL, BEQW)

```
LIMITWELL
  wname    wcutmax gormax wgrmax (qwmax qgmax) qomin qgmin (cement)! well
limits
  ... as many lines as desired ...

    wcutmax    = maximum water cut, fraction
    gormax     = maximum gas/oil ratio, mcf/stb
    wgrmax     = maximum water/gas ratio, stb/mmcf
    qwmax      = maximum water rate, stb/d
    qgmax      = maximum gas rate, mscf/d
    qomin      = minimum oil rate, stb/d
    qgmin      = minimum gas rate, mcf/d
    cement     = optional integer, 0 = crossflow, 1 = cement
```

A value of 0 for any limit signifies no usage of that limit. Default values of all limits are 0. The first five limits are used for well workovers. If the watercut of a well exceeds wcutmax at the end of a time step, then the following workover logic is executed (you can substitute "perf" for "layer" in the following):

Layer productivity index (PI) is zeroed for the layer having the highest watercut.

If nshutwc is > 0 for that layer, then layer productivity indices for all other layers having the same nshutwc value are also zeroed.

If the well watercut remains larger than wcutmax after the layer PI zeroing of Steps (1) and (2) then return to Step (1). Otherwise, exit.

This wcut check and workover, if necessary, is performed at the end of each time step for each well. If and when all layer PI's of a well are zeroed, the well is permanently shut in. If wcutmax is entered as a negative value, then no workover is performed; rather, the well is permanently shut in when wcut exceeds the absolute value of wcutmax. Similar workover checks and PI zeroing are performed using gormax, wgrmax, qwmax, and qgmax, including the negative-value option.

If and when a well is shut in due to one or more of these limits, its rate is internally set to 0 (shutin with recirculation) if "cement" is omitted or entered as 0. If "cement" is entered as 1, then the well is shutin with no recirculation - corresponding physically to the case where the well is actually cemented in, or its perforations are isolated by some other means.

The values of nshutwc, nshutgor, and nshutwgr are entered for each perf under the keyword WELL as described above. The default values of those integers are 0. Entry of nonzero values will reduce recovery but will also reduce the frequency of workovers. Qwmax workovers use the nshutwc integers, and qgmax workovers use the nshutgor integers.

If a well's rate falls below qomin or qgmin, the well is permanently shut in.

The BEQW option can be used to modify the qomin check to include a barrels-equivalent gas value, i.e. the check becomes IF (QO + beqw*QG .LT. QOMIN) THEN.

```
BEQW beqw ! stb/mcf
```

To deactivate all economic limits for all wells, enter

```
LIMITWELL OFF
```

at any time in recurrent data.

To see printout of workovers and shutins due to well limits, search the output file for *WELL.

Reopen shut in wells (OPENWELL)

This optional Recurrent Data entry specifies the frequency of re-opening production wells which have been shut in due to workovers, economic limits, or inability to flow.

```
OPENWELL dtopen {dtfirst} {SHUTIN}  {OFF}
 wname  qopen
 .. as many lines as desired ..
```

where

```
qopen    = well rate to be used when/if the well is re-opened
dtopen   = time, days, between time of shutin and time of re-opening
dtfirst  = first timestep, days, to be used when wells are re-opened
```

Normally, qopen > 0. If qopen=0 the well will not be re-opened. The value of dtopen must be > 0.

In many depletion cases, if a shutin well is re-opened, it will quickly shut in again. If the reservoir undergoes re-pressurization, previously shut in wells may stay open upon re-opening. In field-scale simulations a normal value of dtopen is 300 or 200, perhaps less, but normally not (say) 30 or less.

We denote the time at which wells are re-opened by topen. Successive topens will never differ by less than dtopen but they may differ by more than dtopen – see the Figure below.

In some cases, the re-opening of wells at a re-open time topen can cause very large changes in pressures and saturations in the first timestep after topen, leading to computational difficulties and timestep cuts. Entry of the optional dtfirst (say, 1 or more or less) can mitigate that. If dtfirst is not entered, the first timestep after topen will be related to the last step before topen and the auto-step logic and may be “large”.

In some cases, computational difficulties (timestep cuts) may arise in the first step after shutin of wells subject to the openwell feature. In such cases, entry of SHUTIN on the OPENWELL dataline will apply any entered dtfirst to the first step after shutin (in addition to the first step after re-opening), reducing the number of timestep cuts.

The model saves the well perf locations and indices last entered or calculated under keyword WELL. Those perf values are used upon re-opening of the well. You need not worry about perf indices set to 0 by workovers.

Edit your outfile and search for "*date" or "openwell" to see printout related to this feature. In many cases, wells shut in almost immediately after they are re-opened. If you note this behavior in your outfile, you can cancel all previous OPENWELL entries by using at any time the one-line entry:

```
OPENWELL OFF
```

If you wish only to change any of dtopen, dtfirst, SHUTIN at any time you can use the one-line entry:

```
OPENWELL dtopen {dtfirst} {SHUTIN}
```

You may enter OPENWELL at any time if you wish to remove or add production wells to the previously entered re-open lists:

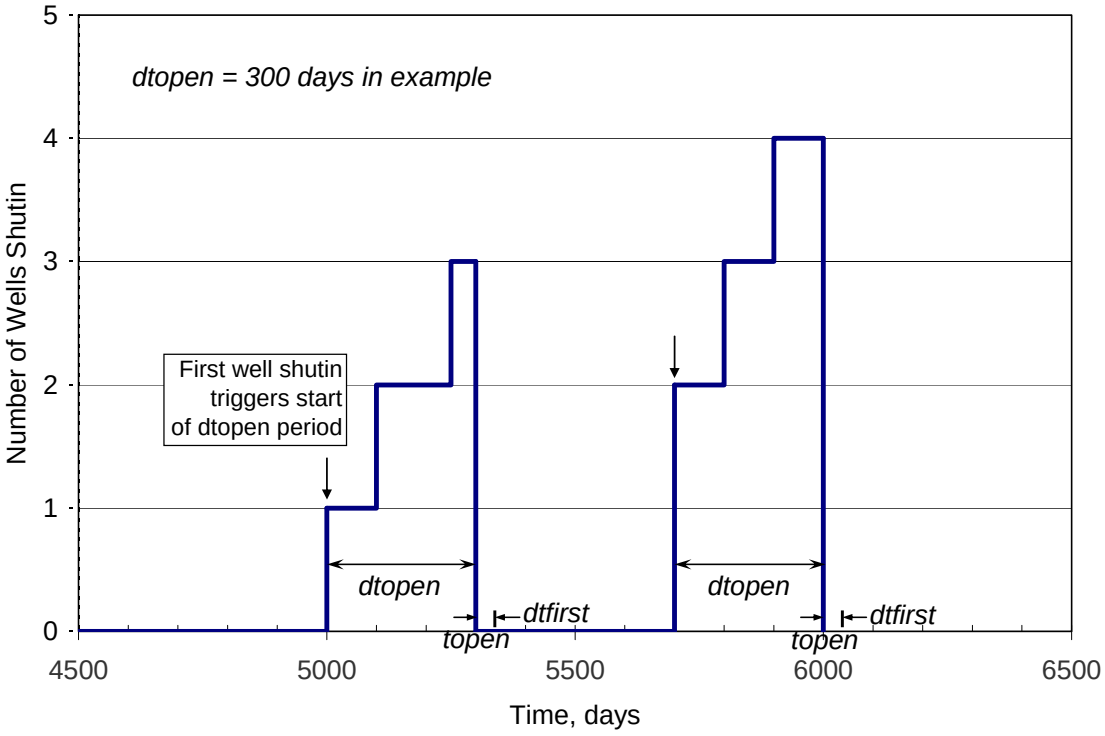
```
OPENWELL  dtopen {dtfirst} {SHUTIN}
    wella  0            ! remove
     wellc  0           ! remove
     wellf 12000    ! new
```

Example:

The entry at some Time t (say, 4500 days)

```
OPENWELL  300
  -1      30000 ! the “-1” means “all producers”
```

applies to all wells specified as producers prior to the entry. If the first producer shutin after 4500 days occurs at (say) 5000 days, then topen is set to 5300 days and all producers which shut in during the 5000-5300 day period are re-opened at 5300 days with rates of 30000. If the first producer shutin after 5300 days occurs at 5700 days then topen is set to 6000 days and all producers which shut in during the 5700-6000 day period are re-opened at 6000 days with rates of 30000.



Well conversion on shut in (CONVERT)

This option allows for automatic conversion of producers to injectors or to other producers when the former are shut in due to economic limit (LIMITWELL) or inability to flow. The recurrent data entry is:

```
CONVERT
  prwell  convwell
  ... any number of these lines ...
```

where

prwell = well name of producer to be converted automatically when it is shut in.

convwell = well name of conversion well. It may be a producer or injector.

If convwell is an injector, it may have perfs identical to those of prwell or not. The well names prwell and convwell must be different. When prwell is shut in, the program will cement well prwell (shut it in with no recirculation) and activate well convwell. After prwell is shut in and converted, it is permanently inactive - any and all later entries of it under RATE, OPENWELL, WAGTBL, DRILL are ignored.

Each conversion well convwell must be entered using all normal keywords, WELL, WELLTYPE, and BHP. The RATE of convwell MUST BE ENTERED NEGATIVE. When convwell is activated, it will produce (inject) at a constrained rate equal to the absolute value of the entered rate.

You may stack conversions, i.e.

```
CONVERT
 PW1  PW1a
 PW1a PW1b
 PW1b PW1c
 PW1c IW1c
 ... any number of similar lines ...
```

This converts producer PW1 to producer PW1a when PW1 is shut in, converts producer PW1a to producer PW1b when PW1a is shut in, etc.

To see CONVERT input data and results, edit your outfile and search for "convert". An example datafile is convert.dat.

Note:

In a rare case you may have a restart run starting from a restart record containing previous CONVERT entries. You may wish to cancel one or more of those entries in the restart run. To do so, enter:

```
CONVERT
  prwell
  ... any number of similar lines ...
```

This will cancel the previous CONVERT entry converting prwell to another well.

WAG option (CYCLETABLE, WAGTBL)

This option is provided in order to automatically cycle wells between water and gas injection in WAG (alternating water and gas) floods.

Any number of Cycletables may be entered. Each Cycletable has any number of Cycles and each of those Cycles has a gas-injection phase and a water injection phase. Values of bhp, rate, and slug size for gas injection and for water injection are entered for each cycle, to be applied to all wells assigned to the Cycletable. The last Cycle can specify continuing water injection only or continuing gas injection only. Upon completion of all Cycles, a well assigned to the Cycletable is shut in.

```
CYCLETABLE nct
  i    ioutsd  bhpgi  bhpwi    hcrate   wrate   hcslug   wslug
    ... as many of these lines as desired ....
```

where

```
nct           = Cycletable #
i             = Cycle number
ioutsd  = OUTSIDE gas # for composition of injected gas
bhpgi   = bhp limit for gas injection phase, psia
bhpwi   = bhp limit for water injection phase, psia
hcrate  = max gas injection rate, mcf/d or rbgas/d
wrate   = max water injection rate, stb/d or rbwat/d
hcslug  = hc slug size, mcf or rb
wslug   = water slug size, stb or rb
```

The WAGTBL keyword is used to assign injection wells to Cycletables.

```
WAGTBL
 wellname  nct
  ... as many of these lines as desired ...

   wellname = well name
   nct      = Cycletable #
```

Notes:

1. If you have n Cycletables, number them 1,2,3,...,n.
2. Each cycle must have water injection or gas injection or both.
3. If hcrate is entered 0 the program internally sets hcslug to 0.
4. If wrate is entered 0 the program internally sets wslug to 0.
5. If hcslug is entered 0 the program internally sets hcrate to 0.
6. If wslug is entered 0 the program internally sets wrate to 0.
7. If ioutsd is entered 0, the injected gas composition will be the INJGAS values if they are specified for the well, otherwise it will be the default surface gas composition.
8. The units of Cycletable injection rates and slug sizes are surface units (MCF and STB) or reservoir units (RB) depending upon the welltype of the injection well assigned to the Cycletable. If the injection well welltype is RB (RBWATINJ or RBGASINJ) then the Cycletable units are RB. If the injection well welltype is surface units (MCFINJ or STBWATINJ) then the Cycletable units are surface units (MCF and STB). The program printout of Cycletable data will label the columns as surface units regardless of whether they are used as such or as rb units as just described.
9. The welltype of a well assigned to a Cycletable must be entered under WELLTYPE before the WAGTBL entry assigning it to a Cycletable, as (a) MCFINJ or STBWATINJ or (b) RBGASINJ or RBWATINJ.
10. If the welltype is gas(water) then the well injects gas (water) first and alternates thereafter.
11. Any injector may be assigned to any Cycletable at any time.
12. A Cycletable must be entered before assigning any injector to it.
13. Any number of injectors may be assigned to a given Cycletable.
14. A given injector may be assigned to only one Cycletable.
15. A previously assigned wag injector may be terminated and shut in at any later time by assigning the well to Cycletable 0. To commence blowdown, all wag injectors may be terminated and shut in (see second Example below).

To view input data related to WAGTBL and CYCLETABLE entries, edit your outfile and search for "WAG".

Example:

The following Cycletable calls for 15 wag Cycles, followed by water injection.

```
CYCLETABLE 1
C                                      rb/d or  rb/d or    rb or   rb or
C                        psia   psia    mcf/d    stb/d      mcf     stb
C   cycle #  ioutsd    bhpgi  bhpwi   hcrate    wrate     hcslug  wslug
       1       3       6200    6500    15000    12000    1500000 1000000
       2       2       6000    6300    20000    15000    1600000 1100000
     3 - 15    1       6800    6700    20000    18000    2000000 1600000
      16       1       6800    6600      0.     20000       0.     1.e15
```

If the last Cycle (16) line is omitted, then a well assigned to this Cycletable will be shut in after completion of the 15 wag Cycles.

Example:

```
........
TIME 2300
WAGTBL
 INJ3  4          ! assign well INJ3 to Cycletable 4
 INJ7  2          ! assign well INJ7 to Cycletable 2
 INJ8 - INJ13  5  ! assign wells INJ8 - INJ13 to Cycletable 5

TIME 3700
WAGTBL
 INJ14 4          ! assign well INJ14 to Cycletable 4
 INJ7  0          ! terminate the well INJ7 wag injection and shut INJ7 in

TIME 5300
C  commence blowdown
WAGTBL  OFF       ! terminate all previously assigned wag injections
.........
```

Example datasets: wag1.dat wag2.dat

TIMEWAG option (TIMEWAG)

This option is a simplified version of the WAG option, applicable to WAG cases involving a single WAG injector for which the pressure constraint is fixed (previously specified in the normal manner using BHP or THP), and the duration and maximum rates for the water and gas injection phases are the same for all injection cycles. The order of injection is given by the specified welltypes. WAG cycling begins at the time at which the TIMEWAG data are specified. The format is:

```
TIMEWAG
  tstop
```

```
  wname
  dt1 rate1 welltype1
  dt2 rate2 welltype2
```

where

```
  tstop = time to stop WAG cycling, days
  wname = a single injection well name or number
  dt1 = length of first injection phase in each cycle
  rate1 = rate for first injection phase, units given by welltype1
  welltype1 = well type for first injection phase, see table below
  dt2 = length of second injection phase in each cycle
  rate2 = rate for second injection phase, units given by welltype2
  welltype2 = well type for second injection phase, see table below
```

welltype1,2	rate1,2
-1	gas, mcf/d
-2	gas, rb/d
-3	water, stb/d
-4	water, rb/d

```
If welltype1 is -1 or -2 (gas injection phase is first), then welltype2 must
be -3 or -4.  If welltype1 is -3 or -4 (water injection phase is first), then
welltype2 must be -1 or -2.  An example is spe5.dat.
```

Pressure control option (PCON)

It is sometimes desirable to simulate waterflood, gas injection, wag, or simultaneous water + gas injection operations under constant pressure conditions. Sensor pressure control logic internally calculates injection well rates vs time to maintain the desired constant average pressure.

Sensor logic allows specification of constant pressure for one or more pressure control (PC) regions for the above simulations. Any number of gas and/or water injection wells may be specified and they may be placed in platforms or not.

Use of this option places PC constraints on wells in addition to any bottomhole pressure (BHP) or tubinghead pressure (THP) or rate (RATE) constraints which may be defined. Wells that are found by the program to be limited by BHP, THP, or RATE (that is, wells that cannot inject their computed allocation of the rate required to maintain region pressure, due to these constraints) will not be on active PC. The well status indicator printed in the WELL tables will be PCON for an injector on active pressure control constraint.

In the case of multiple PC regions, ideally (a) there should be no inter-region communication (transmissibilities), and (b) an injection well completed in one PC region should not be completed in other PC region(s). Neither condition is mandatory. Communication between PC regions may make it difficult to maintain different pressures in them.

The Recurrent input entry, repeated for as many regions or superregions as desired,

```
PCON ireg p
```

specifies a hydrocarbon average pressure of p psia in region ireg. An ireg value of 0 (zero) denotes total field. Both water and gas injectors in region ireg are used. A negative ireg denotes superregion rather than region. The region of a well is the region of its first perf.

Enter

```
PCON ireg p  W
```

to maintain pressure using only water injection.

Enter

```
PCON ireg p  G
```

to maintain pressure using only gas injection.

Enter

```
PCON ireg 0
```

to deactivate a previously entered PCON.

To see printout of entered PCON data, edit your outfile and search for "PCON".

The contribution of a PC Region's injection well rate to the total Region injection rate is determined by the entered well rates Qi. If none of the region's injectors is on pressure or rate constraint, then injector i will inject the fraction Qi/sum(i)Qi of the total region injection rate. The sum(i) is over all injectors in the region.

Depending upon the bhp and rates entered for a PC region's injectors, one or more of the region's injectors may go on pressure or rate constraint. If all the injectors go on constraint, the region average pressure will fall below the desired value. If only some wells are on BHP, THP, or RATE constraint, the other wells will inject at proportionately larger rates to achieve the desired total region injection rate and pressure should be maintained.

Note:

Several factors may contribute to an average region pressure lower than its specified value:

1. Injector bhp limits may limit injection rates.
2. Entered injection well rates (keyword RATE) may limit injection rates.
3. Platform water and/or gas injection rates and constraints may limit rates of injectors in a platform.

The internally calculated region injection rate may change too rapidly, resulting in highly oscillatory injection rate and step-to-step pressure oscillation. That rate of change is related to the sum(i)Qi where the Qi are the entered rates of the PC region's injectors and sum(i) is sum over all such injectors. If the rough average of region injection rate is Q* then sum(i)Qi should not exceed two or more times Q*. If it does, calculated region injection rate Q* may show undesirable oscillation.

Another remedy to the possible oscillation just discussed is as follows. The PCON dataline has a third number - "damp" which prevents these oscillations. If omitted, this third number damp = default value of 0.05. If significant pressure oscillations occur when entering PCON then add

this third number as .03 or .02 or .01 or .. on the PCON dataline. For example, if oscillations occur with (say) the entry

PCON 0 4000 W

then try (say)

PCON 0 4000 W .03

A given injector is assigned only once as a PC well. First, all injectors are assigned on the basis of entered PC regions. Second, unassigned injectors are assigned as PC wells for entered PC superregions. Finally, all unassigned injectors are assigned as PC wells if PC is specified for the field. If there are no PC injectors for a PC region, superregion, or field, the program will print a message to that effect - edit your outfile and search for "PCON".

See examples pcon.dat, pcon1.dat, pcon2.dat, opt_pcon.dat, wag2.dat, and spe5_pcon.dat.

Drilling schedule option (DRILL, RIGS, TDRILL)

The Recurrent Data entry is

```
DRILL  ndrillperyear
  wname  rate
  wname  rate
   ... any number of wells (lines) ..
```

where

ndrillperyear = maximum number of wells allowed to be drilled per year, starting from the time of the DRILL entry.

All listed wells must be entered previously under keyword WELL. The rate unit for a given well is that defined by keyword WELLTYPE. If ndrillperyear is omitted or entered as 0, then the ndrillperyear constraint is not used. This ndrillperyear constraint has carryover; for example, if ndrillperyear were (say) 15 and only 5 wells were drilled in the first year, then 25 drilled wells would be allowed in the next year.

The listed wells must be in Platforms and may include production, gas injection, and water injection wells. They must be listed in order of decreasing drilling priority. If you have previously entered a DRILL list and wish only to change the value of ndrillperyear, then use the one-line entry "DRILL ndrillperyear".

Each time step, the program examines each listed well in the order of the list. If the well is a producer (gas injector) (water injector), the program checks whether the well's Platform production (gas injection) (water injection) target rate was met on the previous step. If not, the well is drilled and removed from the DRILL list, provided the drilling constaints (ndrillperyear and RIGS, discussed below) allow its drilling.

If a rate is entered for a well under keyword RATE, then:

(a) If the well is on the DRILL list but currently undrilled, it is removed from the DRILL list – i.e. it is considered to be drilled.

(b) If the well is entered later under DRILL, its entry under DRILL is ignored.

In the case of multiple Platforms, the placement of different Platform's wells in the DRILL list may be important, in light of the constraint(s) on the number of wells which can be drilled. If, for example, all Platform 1 wells appear in the list before any Platform 2 wells, then Platform 1 wells may be drilled in preference to Platform 2 wells over time.

The optional entry

```
RIGS  nrigs tdrill
```

specifies the number of rigs and the time (e.g. 60 days) required for a rig to drill a well. If all rigs are busy the program will not drill a well. The default is no use of the RIGS constraint. To release a previous RIGS entry and return to no RIGS constraint, enter RIGS 0.

The RIGS entry sets the tdrill value for all wells, whether entered or drilled at the time of the RIGS entry or not. For example, if the total number of wells in the run were 900 but only 170 were entered when "RIGS 7 80" is entered then tdrill is set to 80 days for all 900 wells.

The value of tdrill may be set to different values for different wells by use of the TDRILL keyword. Note that tdrill values are used only if RIGS is entered and nrigs > 0.

```
TDRILL
  wname  tdrill  ! as many lines as desired
```

The constraints on well drilling can be both or neither of ndrillperyear and RIGS, or either one alone. An additional, hard coded constraint is that not more than five wells are drilled at any given time step.

Note that the RIGS constraint alone constrains the number of wells drilled per year. For example, the entry "RIGS 5 60" restricts the number of wells drilled per year to about 30.

To view printout of entered data and drilling of wells, edit your outfile and search for the strings "RIG"and "DRILL".

Salinity option (WELLSALT)

This option calculates the salinity of produced water resulting from different salinities of original reservoir water and injected water. This is most easily discussed as the displacement of original reservoir water by source water.

All water injectors inject original water until the time of entry of WELLSALT data indicating otherwise.

```
WELLSALT
 wname    fsource      !  fraction of source water in injected water
    . . .          . . .          !  as many lines as required
```

where fsource is the fraction of source water in the injected water and will vary from 0 to 1.0. Default is 0.0. Injected water salinity may be source water or any mixture of source water and original water, and may be changed by well and with time.

The current and cumulative produced fsource values (labeled FSALT) are printed at the end of the Well Table (frequency WELLFREQ). They are printed for total field and by Reservoir, Platform, Region, Superregion, individual wells, and by well perfs. Edit your outfile and search for "FSALT" to see that printout.

To print or file write the fsource map, use the mapname CSALT on your MAPSPRINT and MAPSFILE datalines.

Examples:

If you want to start source water injection at time 0 for all water injectors initially and later active throughout the entire run, enter before the first TIME (or DATE) line of Recurrent Data

```
WELLSALT
  -4  1.
```

This specifies source water injection (fsource=1.) for all wells, whether they have been entered under WELL before this entry or not. This fsource injection will be used only for water injectors. Wells may be switched from producers to gas injectors to water injectors as desired throughout the run but any well when injecting water will inject source water. For this case this single entry is the ONLY salt data entry in the entire datafile.

Let the salinity of original water be 25000 ppm and source water have salinities of any lower values down to a minimum salinity of 5000 ppm, varying by well and with time. Define source water as the 5000 ppm water.

You can enter, for example,

```
WELLSALT
 WINJ3  1.
 WINJ6 - WINJ9  .8
 WINJ12  0.  ! if a nonzero fsource was previously entered
```

The 1. value specifies an injection salinity of 5000 ppm. The .8 value specifies an injected source-original mixture of salinity = .8*5000+.2*25000 = 9000 ppm.

Entries like the above can be entered throughout the Recurrent Data to specify injection salinities varying by well and with time.

Example Datafiles: salt1.dat, salt2a.dat, salt2b.dat.

Tubinghead pressure tables (THPTABLE)

Tubinghead pressure (THP) tables for production wells tabulate bottomhole pressure versus a primary phase surface rate, phase surface rate ratios, and tubinghead pressure. THP tables for injection wells may be specified for water, gas, and SWAG (simultaneous water and gas) injectors, and tabulate bottomhole pressure versus total injection rate, gas fraction, and tubinghead pressure. Data in a given table reflects a particular tubing size, length, roughness, configuration (including point of gas lift entry, if applicable), and PVT characterization. However, hydraulics in well tubings differing only in tubing length may be approximated by a single table through entry of the TUBINGZ and WELLHEADZ data.

THP tables for production wells are characterized as either oil THP tables, for wells which are predominantly oil or liquid producers, or as gas THP tables, for wells which are predominantly gas producers. Both oil and gas tables are capable of representing tubing hydraulics when all 3 phases are being produced.

Rates in each oil THP table may be defined as either oil surface rates in units of stb oil/d or as liquid surface rates in units of stb liquid/d. Secondary phase ratios in each oil THP table may be

defined as either gas-oil ratio (gor) in units of scf gas/stb oil or as gas-liquid ratio (glr) in units of scf gas/stb liquid. Gas lift data may be entered only for oil THP tables, in the form of optimal gor or optimal glr versus rate, wcut(wor), and thp. See the gaslift table example below for further discussion of optimal gor(glr).

Rates in gas THP tables are defined as surface gas rates in units of mscf/d. Secondary phase ratios are specified as liquid-gas ratio (lgr) in units of stb liquid/mmscf gas.

For both oil THP tables and gas THP tables, water phase ratios are entered either as water cuts (wcut, stb water/(stb liquid) or as water-oil ratios (wor, stb water/stb oil). If the maximum water phase ratio entered is greater than unity, the program interprets the entered values as wor values. **If your table expresses water phase ratio as wcut, do not input (non-physical) wcut values greater than unity. If your table expresses water phase ratio as wor, and the maximum wor entry is less than or equal to unity, you must convert your table wor entries to wcut values for input to Sensor.** This conversion is given by the equation:

wcut = wor / (1.0 + wor)

Rates in injection THP tables are defined as the sum of the rates of water (qw, stb/d) and gas (qg, mscf/d), so that a single table format can be used for all types of injectors. Bottomhole pressures are a function of rate, gas fraction fg=qg/(qw+qg), and tubinghead pressure.

The program uses bilinear interpolation to determine bottomhole pressure for given values of rate, gor (glr, lgr), wcut (wor), and thp (rate, fg, and thp for injectors). The limiting thp for each well (maximum for injectors, minimum for producers) is entered using keyword THP. Any input BHP constraints are ignored for wells with specified THP constraints.

Oil and gas THP tables for production wells are specified as

```
THPTABLE
itable iliq igl runit qunit ! omit runit and qunit for gas THP tables
nq nr nw nthp
q(1) q(2) ... q(nq)
r(1) r(2) ... r(nr)
w(1) w(2) ... w(nw)
t(1) t(2) ... t(nthp)
((((bhp(ir,iq,iw,it), ir=1,nr), iq=1,nq), iw=1,nw), it=1,nthp)
(((glgor(iw,iq,ithp), iw=1,nw), iq=1,nq), it=1,nthp) !if iliq=1,igl=1
```

and THP tables for injection wells are specified as

```
THPTABLE
itable
nq nfg nthp
 q(1)  q(2) ... q(nq)
fg(1) fg(2) ... fg(nfg)
 t(1)  t(2) ... t(nthp)
(((bhp(iq,ifg,it), iq=1,nq), ifg=1,nfg), it=1,nthp)
```

where

itable	= THP table number
iliq	= 1 for an oil table, 0 for a gas table
igl	= 1 if gaslift data are included (only for oil tables), 0 if not
runit	= definition of secondary phase ratio r for oil tables, either GOR for gas-oil ratio in scf /stb, or GLR for gas-liquid ratio in scf /stb. Also defines glgor. Omit for gas tables.
qunit	= definition of primary rate q in oil tables, as either QOIL for oil rate in stb/d, or QLIQ for liquid rate in stb/d. Omit for gas tables.
nq	= number of rate entries in the table
nr	= number of secondary phase ratio entries in the oil or gas table
nw	= number of water phase ratio entries in the oil or gas table
nfg	= number of gas fraction entries in the injection table
nthp	= number of tubinghead pressure entries in the table
q	= rate entries, stb oil/d or stb liquid/d for oil tables, mscf gas/d for gas tables, stb water/d + mscf gas/d for water injectors
r	= secondary phase ratio entries, either gor in scf/stb or glr in scf/stb for oil tables, lgr in stb/mmscf for gas tables
w	= water phase ratio entries, either wcut or wor
t	= tubinghead pressure entries, psia
fg	= gas fraction entries, defined as qg/(qw+qg), qw=water rate in stb/d, qg=gas rate in mscf/d
bhp	= table bottomhole pressure entries, psia
glgor	= optimal gor (scf/stb) or glr (scf/stb), defined by runit

Notes:

1. nq must be greater than 1
2. nr, nw, nthp, and nfg must be greater than zero
3. q, r, w, and t, and fg must be entered in monotonically increasing order
4. 0 <= wcut <= 1.0, 0 <= fg <= 1.0
5. If w(nw) <= 1.0, w must be input as wcut.

6. The lowest (first) entered rate q(1) is treated as a cutoff rate, below which the well will not flow. A well with an input THP constraint will be permanently shut-in if its rate declines below that value.
7. Rates are extrapolated beyond the maximum table entry. No extrapolation is performed on values of r, w, t, or fg.

Bottomhole pressure values obtained from the tables may be adjusted to reflect well tubing lengths that are different from the length used to generate the table, through entry of the TUBINGZ keyword:

```
TUBINGZ
 itable   tubz  ! as many lines as required
```

where tubz is the vertical component of the tubing length used to generate the table, a positive number in units of feet. By default, all wellheads are assumed to lie at zero depth. Wellhead depths may be specified as

```
WELLHEADZ
  Wname   whz   ! as many lines as required
```

where whz is the wellhead depth in feet. WELLHEADZ data are not used unless the well is assigned a THP table for which TUBINGZ data are defined. The bhp correction made by the program is

corrected bhp = thp + (bhp-thp)*(perfz-whz)/tubz

where bhp on the right hand side is the THP table value, and perfz is depth to the bhp perf of the well.

Example 1:

Entry of two tables, one for oil wells, the second for gas wells. Values are not shown for bhp.

```
THPTABLE
1  1  0                  !   itable iliq  igl
10 10 3 2                !   nq nr nw  nthp
100. 300. 500. 700. 1000. 2000. 3000. 5000.
     7000. 10000.        !   nq oil rates, q_oil, stb/d
1000. 1500. 2500. 5000. 7500. 10000. 15000.
    20000. 25000. 50000. !   nr gor values, scf/stb
0. .5 .9                 !   nw wcut values, fraction
615.  1015.              !   nthp thp values, psia
C bhp values are not shown
C ((((bhp1(ir,iq,iw,it), ir=1,nr),iq=1,nq),iw=1,nw),it=1,nthp)


2  0  0                       !  itable iliq  igl
10 10 3 2                     !  nq nr nw nthp
100. 7867. 15633. 23400. 31167. 38933.
   46700. 54467. 62233. 70000. !  nq gas rates, q_g mcf/d
10. 42.2 74.4 106.7 138.9 171.1
 203.3 235.6 268.8 300.       !  nr liquid-gas ratios, stb/million scf
```

```
0. .5 .9                        !  nw wcut values, fraction
215.  515.                      !  nthp thp values, psia
C bhp values are not shown
C ((((bhp2(ir,iq,iw,it), ir=1,nr),iq=1,nq),iw=1,nw),it=1,nthp)
```

Example 2 – Gas Lift Table:

```
     1  1  1                    itable iliq  igl
    10 10 3 2                   nq nr nw nthp
   100. 300. 500. 700. 1000. 2000. 3000. 5000.
         7000. 10000.           nq oil rates, q_oil, stb/d
   1000. 1500. 2500. 5000. 7500. 10000. 15000.
       20000. 25000. 50000.    nr gor values, scf/stb
      0. .5  .9                nw wcut values
      615.  1015.             nthp thp values, psia
   C  bhp values are not shown
   C ((((bhp1(ir,iq,iw,it), ir=1,nr),iq=1,nq),iw=1,nw),it=1,nthp)
   C (((glgor(iw,iq,ithp), iw=1,nw), iq=1,nq), it=1,nthp)
   C  gaslift data for first thp value   nw values per line
        6600.0  6600.0  6600.0    ! first line, first rate
        5900.0  6000.0  6000.0    ! second rate
        5400.0  5400.0  5400.0    ! third rate
        5000.0  5000.0  5000.0
        4200.0  4600.0  4600.0
        2400.0  3900.0  3900.0
        1800.0  2200.0  2200.0
        1500.0  2100.0  2100.0
        1400.0  2000.0  2000.0
        1300.0  1900.0  1900.0
   C   gaslift data for the second thp value
          ......
```

The first line contains optimal gor(glr) values for the first rate, the second line is for the second rate, etc. In each line, the i th value is optimal gor(glr) for the i th wcut (wor). The optimal gor(glr) values have units of scf/stboil(stbliq).

The entered BHP data allow xy plots of bhp vs gor(glr) for given fixed values of thp, rate, and wcut (wor). Theoretically, the "optimal" gaslift gor(glr) is the value at which bhp is a minimum on such a plot. Due to flat minimums, the entered gaslift data may specify gor(glr) values lower than those "optimal" values. However, there is no reason to enter a gaslift gor(glr) value larger than the "optimal" gor(glr) value. Doing so wastes gaslift gas and reduces oil rate. The program checks for this and resets any entered gaslift gor(glr) which exceeds the "optimal" value to that latter value. This reset is accompanied by a "WARNING" message.

Example 3 – Injection Table:

```
  THPTABLE
   1          ! itable = table # = 1
```

```
3 3 1      ! nq = 3 rates  nfg = 3  nthp = 1
2000.  10000  20000   ! rates q = qw stb/d + qg mcf/d
0. .5  1.  ! values of fg = qg/(qg+qw)
3000       ! thp values
6000 5800 5500 ! bhp values
5000 4950 4750
4000 3900 3700
```

Datafile test5.dat also illustrates the entry and use of the tubinghead pressure table.

Flattening of thp table outflow curve (FLAT)

There are differing opinions regarding well performance when the well is operating at an intersection of the inflow and outflow curves where the outflow curve slope (bhp vs rate) is negative. If you enter

```
FLAT 1
```

in Recurrent Data before THPTABLE then Sensor will flatten the low-rate negative-slope portions of the outflow curves to zero slope. In default mode, Sensor does not do that. If your thp table curves exhibit low-rate negative slope and you do not enter FLAT 1, then there are two possible intersections of the inflow and outflow curves on the negative slope portion of the outflow curve. The lower-rate intersection is unstable and the higher-rate intersection is stable. Sensor permanently shuts in a thp well if either no intersection is found or the unstable intersection is encountered.

Notes on Entered Well Rates and Other Well Data (WELLOFF)

When shutting a well in, enter the rate value as -1 (or any negative value, -1 is recommended) to prevent recirculation, or as 0 to allow recirculation. Use of -1 should be avoided except when using duplicate wells, or in the rare case where a well is actually cemented in. You can enter all your wells at time 0 and not worry about recirculation (wellbore crossflow) occurring before they are actually drilled at a later time. A well rate will remain at the default -1 value (corresponding to an undrilled status, shut in with no crossflow) until that time in the Recurrent Data when a rate for it is first entered.

You may on occasion use two different well names for the same physical well. The usage will be such that when one well is on, the other is off. In such a case, make sure that the well which is off is given a rate of -1, not 0.

The program remembers entered well rates from time period to time period. Thus you do not have to reenter any well rate unless it changes. If instead you want all well rates zeroed at the start of each new time period (thus all well rates not reentered at a time period will be zero rather than the remembered value), then enter:

```
WELLOFF 1
```

The default is WELLOFF 0 (program remembers entered values).

See Appendix 8 for a discussion of individual perf rates and cumulatives.

The meaning of status indicators which are printed in Well Tables is:

```
  SI    well is shut in - target rate is 0
   Q    well is on target (entered) rate
  PC    well is on pressure constraint
  NF    well cannot flow (e.g. because reservoir presssure is
        below entered BHP limit or below bhp values in the thp
        table).
  QG    well is producing entered QGMAX rate (mcf/d).
  DD    production well rate is limited by drawdown
PCON    injection well rate is on pressure control constraint
```

Numerous entries of WELL, PIMULT, WELLTYPE, BHP can make it difficult for the engineer to know or recall his many well property settings at various times through the simulation. The Recurrent Data entry PRINTWDATA causes printout of all wells and their properties at the time of that entry. The printout is easily found by editing the outfile and searching for "PRINTWDATA". The Recurrent Data entry is as follows

....

TIME 1200

... data ...

PRINTWDATA

TIME 1700

... data ...

TIME 2400

....

This activates printout of well data which are in effect for the period 1200-1700 days for all wells. To see this well data printout, edit your outfile and search for "PRINTWDATA".

3.2 Field Limits (LIMITFIELD, BEQF)

Field economic limits are entered using the keyword LIMITFIELD:

```
LIMITFIELD  wcutmax gormax wgrmax qomin qgmin (qinjmin)  !(field limits)

          wcutmax = maximum water cut, fraction
           gormax = maximum gas/oil ratio, mcf/stb
           wgrmax = maximum water/gas ratio, stb/mmcf
            qomin = minimum oil rate, stb/d
            qgmin = minimum gas rate, mcf/d
          qinjmin = minimum total injection rate, optional
```

A value of 0 for any limit signifies no usage of that limit. Default values of all limits are 0. If field watercut exceeds wcutmax, the run is terminated with printout of timestep and well and platform tables and arrays. Similar action is taken in regard to gormax and wgrmax. If field oil (gas) production rate falls below qomin (qgmin), the run is terminated with printout.

The BEQF option can be used to modify the qomin check to include a barrels-equivalent gas value, i.e. the check becomes IF (QO + beqf*QG .LT. QOMIN) THEN.

```
BEQF beqf ! stb/mcf
```

The optional minimum field injection rate terminates the run if qinj is less than qinjmin, where qinj = qwatinj (stb/d) + qoinj (stb/d) + qginj (mcf/d).

3.3 Field Target Rate (FTARG)

The entry

```
FTARG G 130000    ! qfield
PTARG 1 G 80000   ! q1
PTARG 2 G -100000 ! qswing
```

specifies a field target of 130000 mcf/d. Platform 1 is to produce a fixed rate of 80000 mcf/d, which will decline in time as capacity falls below that rate. Platform 2 is a swing platform, denoted by negative entry of its production target (maximum) rate. The program will calculate the swing platform rate so that the total production rate equals the entered field rate. There may be as many fixed-rate platforms as desired but only one swing platform. The swing platform rate is

$$q = \text{Min}\ \{\ q_{field} - \sum_{j=1}^{j=n} (q^*_j),\ q^*_{swing}\ \}$$

where q^*_j is the lesser of entered rate and capacity for fixed-rate platform j, and n is the number of fixed-rate platforms. The rate q^*_{swing} is the lesser of the entered q_{swing} and the swing platform capacity. The sum of entered rates for the fixed-rate platforms must be less than the entered qfield. To deactivate this field-level control, enter FTARG 0.

The target rate character G in the entries above can also be (letter) O (for stboil/d) or WG for wetmcf/d. Do not enter FTARG if there is only one platform.

Note: To exclude any platform from contributing to the FTARG field target, specify that platform's platform number as negative in the PTARG data. For example, the entry

```
PTARG -1 G 80000   ! exclude platform 1 from FTARG
```

results in platform number 1 not contributing to the field target.

3.4 Platform (Gathering Center) Data

The platform keywords are

```
WELLPLAT PTARG ITARG OUTSIDE LIMITPLAT
COSMETIC YPLAT
```

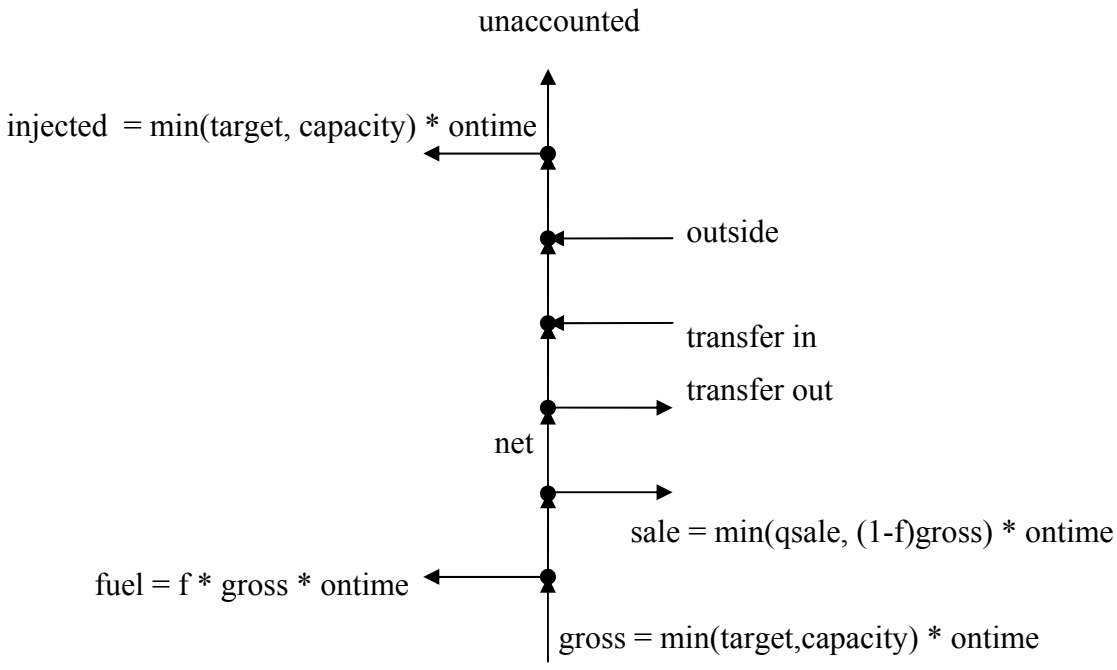
in addition to the previously discussed well keywords.

General Discussion

The term platform here denotes a group of wells. Elsewhere, the terms "gathering center" or simply "well group" are used for this. The wells in a given platform may be any mix of producers, water injectors, and gas injectors. Platforms are not mandatory - all wells or no wells may be assigned to platforms. If there are platforms, some wells may not be assigned to any platform.

Platforms may be used simply for cosmetic purposes - just to see the injection / production performance, rates and cumulatives, of groups of wells. For a “cosmetic” platform, no target rates are entered and each well in the platform operates independently just as if it were not in the platform. Any and all Platform entries PTARG, ITARG, LIMITPLAT, YPLAT, and ONTIME are ignored for a cosmetic platform.

For each active (non-cosmetic) platform, you can specify target rates and constraints such as maximum water or gas rate. A platform's gas injection can include a) gas produced from the platform, b) gas transferred from other platforms, and c) outside gas. The model blends these streams and internally calculates the default composition of the platform gas injection stream. This default composition can be reset to a specified processed gas composition (YPLAT). Platform gas streams are illustrated in the figure below.



Ontimes may be entered for each platform individually for production, gas injection, and water injection. Ontimes can differ for production and injection for a given platform, and can differ from one platform to another.

Production targets apply to the gross gas production (before it is scaled by ontime) and may be specified in any of a number of different units. Each platform's net gas production rate is calculated by the model as

net gas production rate = {min(target rate,capacity) - sales - fuel}* ontime

The total injection target for the platform is given by the sum of the number of different types of individual injection targets. Individual injection target contributions may be specified in the ITARG data as a fixed rate or as a fraction (not both) of any platform's net gas production, and as a fixed rate of outside gas. For example, the total injection target for platform 2 might be 100% of the net gas from platform 2 plus 50% (or optionally some fixed rate) of the net gas from platform 1 (this is a transfer in to platform 2, out of platform 1) plus some fixed rate of outside gas (see Example 4 at the end of this section). The "transfer in" stream is nonzero only when the platform injection target includes gas from another platform. The "transfer out" stream is nonzero only when a portion of the platform's production is specified as a portion of the injection target in another platform. Outside gas rate is nonzero only when the injection target includes some specified rate of outside gas.

A platform's gas injection well injects a rate q mcf/d and a composition y(i), i=1,nc. For a cosmetic platform:

y(i) is the composition entered under INJGAS for the well. If no INJGAS entry occurs for the well, then y(i) is the default surface gas composition

For an active (non-cosmetic) Platform:

An injected gas composition, Y(i), is calculated internally from the amounts and compositions of produced gas, transfer gas, and outside gas.

If a Platform gas injector is not assigned to a wagtable, its y(i) = Y(i). If a Platform gas injector is assigned to a wagtable, its y(i) = that of the Cycletable to which it is assigned.

The total platform injected gas composition printed in the Platform Table is

sum(qyi)/sum(q)

where sum is summation over all Platform gas injectors. The Platform Table reports as "production" the rates, compositions, and cumulatives of the gross, fuel, sale, and net production streams. The table reports as "injection" the rates, compositions, and cumulatives of the total injection stream and of the portions of the total injection stream contributed by produced, outside, and transfer gas. Also reported under "injection" is "unaccounted gas". Unaccounted gas will result if all net produced gas is not specified as being reinjected (as a target), or if a specified injection target cannot be met due to capacity limitations. The model performs a gas balance for all non-cosmetic platforms. The gas balance is

gross - sales - fuel + outside gas injected = total injected gas + unaccounted gas

or

balance=net gas produced + outside gas injected - unaccounted gas - total injected gas=0

where each term is a rate summed over all non-cosmetic platforms. Transfer terms do not appear because each transfer in is exactly canceled by a corresponding transfer out.

If one or more of the platform total gas injection targets cannot be met due to platform injection capacities, then (first) any specified outside gas will be reduced. If the remaining portion of the target cannot be met, then if that target consists of a fraction of some platform's production, that platform's production rate will be reduced to satisfy the injection target. This reduction in production rates is optional as discussed under ITARG below. To repeat, cosmetic platforms are not included in the balancing just described.

An outline of platform data (keyword) is:

a) assign wells to platforms (WELLPLAT)

b) assign maximum rates to wells (RATE)

c) define any cosmetic platforms (COSMETIC)

d) specify injected gas composition (YPLAT)

e) specify platform target rates (PTARG,ITARG)

f) specify outside gas compositions (OUTSIDE)

g) specify platform constraints (LIMITPLAT)

h) specify ontimes (ONTIME)

All input data are remembered. You do not need to reenter keywords except to change previously entered data. Throughout here, net (q_{net}) and gross (q_g) platform gas production rates are:

$$q_{net} = q_g - q_{sale} - q_{fuel} \quad \text{mcf/d}$$

Assign wells to platforms (WELLPLAT)

Keyword WELLPLAT is used to assign wells to platforms if platforms are used.

```
WELLPLAT       ! if platforms are to be used
  wname  n    ! assign well wname to platform n
  ... as many lines as desired ...
```

If you wish to remove a previously assigned well from a platform, enter

```
WELLPLAT
  wname 0      ! well is in no platform
```

Declaration of cosmetic platforms (COSMETIC)

```
COSMETIC iplat   ! platform iplat is a cosmetic platform
```

Do not enter any of PTARG,ITARG,FUEL,SALE,ONTIME for a cosmetic platform. Entry of PTARG or ITARG for a platform automatically cancels any previous cosmetic declaration.

Processed gas composition (YPLAT)

Enter the keyword YPLAT if you want injected gas to have a processed composition different from that given by the internal calculation. The entry is:

```
YPLAT iplat
 .. nc mol fractions ..  !  processed gas composition
  (iplat = platform number)
```

If you have previously entered YPLAT and later wish to change to use of the internally calculated composition for the injected gas, then enter:

```
YPLAT iplat  SURFACE  ! use surface gas composition
```

Outside gas composition (OUTSIDE)

This entry is required if your ITARG platform injection target rates specify injection of any outside gas.

```
OUTSIDE i
 .. nc mol fractions ..  ! composition of outside gas i
 i    = outside gas number
```

Platform production target (PTARG)

To specify a production target (maximum production rate) for platform iplat, enter

```
PTARG iplat unit q (NOREDUCEQ)
```

where

iplat	= platform number	
Q	= platform production target rate	
unit	= STBOIL (or O)	for oil, stb/d
	= MCF (or G)	for gas, mcf/d
	= MCFWET (or WG)	for mcf/d wet gas
	= STBLIQ	for liquid (oil+water), stb/d
	= RBTOT	for total fluids at reservoir conditions, rb/d
	= RBOIL	for oil at reservoir conditions, rb/d
	= RBGAS	for free gas at reservoir conditions, rb/d
	= RBWAT	for water at reservoir conditions, rb/d
	= RBLIQ	for liquid at reservoir conditions, rb/d
	= STBWAT	for water, stb/d

The NOREDUCEQ label optionally indicates not to reduce Platform iplat's rate in order to meet any fractional gas reinjection targets in platforms that are limited in gas injection capacity.

Wet gas, mcf/d, is the total production stream before surface separation (excluding water), expressed as (mols hydrocarbon fluid/d) * (Mcf/mol).

The default production target q is zero for a non-cosmetic platform, i.e. the platform will not produce unless a target is specified.

Platform production rate will be the minimum of the specified target and the platform capacity, which is determined by platform well constraints of RATE and BHP or THP.

To shut in platform production, enter

```
PTARG iplat SHUT    ! shut in platform iplat production
```

Note: To exclude any platform from contributing to the FTARG field target, specify that platform's platform number as negative in the PTARG data. For example, the entry

```
PTARG -1 G 80000   ! exclude platform 1 from FTARG
```

results in platform number 1 not contributing to the field target.

Platform injection target (ITARG, OUTSIDE)

Gas Target

To specify contributions to the total gas injection target for platform iplat, enter

```
ITARG iplat G  f jplat
```

If f is 1.0 or less, then

f = fraction of platform jplat net gas production to be injected in platform iplat.

If f is > 1.0, then

f = the rate of platform jplat net gas to be injected in platform iplat

All default injection targets are zero for a non-cosmetic platform, i.e. gas injection will not occur unless some target is specified.

To specify rate of an outside gas as a contribution to the injection target, enter

```
ITARG iplat OUTSIDE j  q !inject qmcf/d outside gas j in platform iplat
```

To shut in gas injection in platform iplat, enter

```
ITARG iplat G SHUT      ! shut in platform i gas  injection
```

If a gas target is specified as a fraction of production, then by default, if the fractional reinjection cannot be met due to insufficient injection capacity (after reduction of any outside gas), platform jplat production will be reduced such that the fractional reinjection target is met. If you do not wish this to occur, put NOREDUCEQ on the PTARG dataline for platform jplat. In that case, platform jplat production will not be reduced, and the specified injection target will not be met, i.e. the fractional gas reinjection target will be treated as a maximum rate that may be limited by injection capacity, as all other targets are treated.

Example:

Assume a gas PTARG q of 100000 mcf/d, no sales or fuel, and a desired injection of 100000 mcf/d. What action do you want if injection capacity is (say) 70000? If ITARG is entered as a fraction, 1.0 here, then the platform production rate will be reduced by the injection shortfall of 30000 and platform production and injection rates will be 70000 and 70000. If the NOREDUCEQ label is added to the PTARG data line, or if ITARG were entered 100000, the platform production rate would not be reduced, and production and injection rates would be 100000 and 70000.

```
ITARG i OUTSIDE j q !inject q mcf/d of outside gas j in platform i
ITARG i W q         ! inject q stb/d water in platform i
ITARG i W SHUT      ! shut in platform i water injection
ITARG i G SHUT      ! shut in platform i gas injection
```

Water target (OUTSIDEWATER)

To specify a water injection target for Platform n, enter

```
ITARG iplat W v
```

where

iplat = Platform number
v > 2 Platform water injection target rate = v stb/d
v <= 2 Platform water injection target rate = v * (Platform water production rate)

The default target v is 0, i.e. water injection will not occur unless the target is specified.

To shut in water production in platform iplat, enter

```
ITARG iplat W SHUT
```

The seldom-used OUTSIDEWATER keyword can optionally be entered to specify the maximum water rate available for injection in the platform over and above the platform water production rate.

```
OUTSIDEWATER iplat  qwout
```

where

iplat = Platform number
qwout = available external or outside water, stb/d

Let

qwinjtarg = Platform water injection target rate, entered by ITARG
qwprod = Platform water production rate, stb/d
qwinj = Platform water injection rate

If and only if qwout > 0:
qwinj will be the lesser of available water injection rate (qwout+qwprod),
target rate (qwinjtarg), and injection capacity.

If qwout = 0 (i.e. OUTSIDEWATER is never entered):
qwinj will equal the minimum of qwinjtarg and injection capacity.

The default value of qwout is 0. Therefore, "OUTSIDEWATER iplat 0." never needs to be entered (except to change a previously entered qwout > 0 value).

Example:

```
OUTSIDEWATER 3  110000  ! STB/D WATER
ITARG 3 W 150000        ! STB/D WATER
```

If unlimited by capacity, platform 3 water injection rate qwinj (stb/d) will be:

110000 + qwprod when qwprod < 40000 stb/d
150000 when qwprod > 40000 stb/d

Note: If qwout is entered > 0, it doesn't make any sense to enter the ITARG water injection rate as a fraction (<1) of produced water rate.

Target Allocations

The following applies to each target classification – i.e., production, water injection, and gas injection, except that it does not apply to production targets if the OPTPLAT option is used as described in a following subsection.

Let q(i) equal the rate of each (production, water injection, or gas injection) well assigned to the active platform, before any platform target constraints are applied. These rates are the rates each well would make if operated independently of the platform, which are determined by reservoir conditions and well constraint entries of RATE and BHP or THP. The fractional allocation of a (production, water injection, or gas injection) platform target to each platform (production, water injection, or gas injection) well, f(i), is given by:

$$f(i) = q(i) / [\text{sum}(q(j)), j=1,\text{nwells}]$$

The denominator of the right hand side is the platform capacity. A platform target is of course allocated to the wells only when the target is less than capacity (or possibly equal to capacity, for fractional gas reinjection targets). When production is reduced to meet a fractional gas reinjection target (see previous subsection), the gas injection target rate in mcf/d is effectively adjusted to equal injection capacity.

Fuel loss (FUEL)

```
FUEL iplat f  ! fraction f of platform iplat gross produced gas is
              !  burned as fuel, default 0.
```

Sale gas (SALE)

```
SALE iplat qsale ! sale of gas from platform iplat is
                 !  qsale mcf/d, default 0.
```

The program sets the sale gas rate as $\text{Min}(q_{sale},(1-f)q_g)$ where f is fuel use fraction and q_g is platform gross gas production rate. If q_{sale} is entered as a fraction f_{sale}, 1.0 or less, it is treated as a fraction of gross gas production rate. In this case the program sets $q_{sale} = f_{sale} * q_g$ mcf/d, and then applies $q_{sale} = \min(q_{sale},(1-f)q_g)$.

Platform ontimes (ONTIME)

```
ONTIME iplat  fp fwi fgi
    iplat = platform number
       fp = production ontime
      fwi = water injection ontime
      fgi = gas injection ontime
```

All ontimes are fractions, greater than 0 and less than or equal to 1.0. Default values at 0 time are 1.0. See Appendix 9 for further discussion of ontimes.

Platform limits and constraints (LIMITPLAT, BEQP)

The one-line entry (eight numbers)

```
LIMITPLAT iplat  wcutmax gormax wgrmax qomin qgmin qwmax qgmax
```

enters platform limits or constraints for platform iplat. The platform limits are:

wcutmax = maximum water cut, fraction

gormax = maximum gas/oil ratio, mcf/stb

wgrmax = maximum water/gas ratio, stb/mmcf

qomin = minimum oil rate, stb/d

qgmin = minimum gas rate, mcf/d

qwmax = maximum water rate, stb/d

qgmax = maximum gas rate, mcf/d

A value of 0 for any limit signifies no usage of that limit. Default values of all limits are 0 at 0 time. To deactivate a previously entered limit, enter 0.

If platform watercut exceeds wcutmax, the platform production and water injection are shut in. Similar action is taken in regard to gormax and wgrmax. If platform oil (gas) rate falls below qomin (qgmin), the platform production and water injection are shut in.

If platform water (gas) production rate exceeds qwmax (qgmax), the platform water (gas) production rate will equal qwmax (qgmax).

The keyword input is illustrated by examples at the end of this section. We assume 25 gas production wells, 10 gas injectors and 5 water injectors, entered under keyword WELL in the order

P1,P2,...,P25,G1,G2,...,G10,W1,W2,..,W5.

and assume that WELL, WELLTYPE, BHP or THP, and any LIMITWELL data have been entered for the wells.

The BEQP option can be used to modify the qomin check to include a barrels-equivalent gas value, i.e. the check becomes IF (QO + beqp*QG .LT. QOMIN) THEN.

```
BEQP beqp ! stb/mcf
```

To see printout of shutins due to Platform limits and constraints, search the output file for *PLAT.

Platform optimization option (OPTPLAT)

The recurrent data entry

```
OPTPLAT iplat
```

allocates any Platform iplat production targets to optimize oil production in Platform iplat using production well penalty factors proportional to (gor(i)+wor(i))**n for well i. The default value of n is 3.

Note:

1. Enter OPTPLAT iplat 0 if you wish to deactivate a previous entry of OPTPLAT for Platform iplat.
2. Enter OPTPLAT iplat n if you wish to change the value of n. The value of n should be in the range of 1 to 5. The program internally sets n to 10 if n is entered > 10.
3. If the entered Platform target rate approaches or exceeds the sum of entered Platform production well rates, there is no room for allocation or optimization and this entry will have no effect.

See example Problem opt_pcon.dat.

Platform Production Cutback Logic

Calculate net production rate q_{pi} for each platform i,

q_{pi} = (min(target,capacity)*(1-fuel)-sale)*ontime(i)

Use itarg data including transfers to calculate average gas injection rates q_{Ii} for each platform (these may exceed platform injection capacities). Store this information as

q_I(i,j) = injection rate of platform i gas into j.

The above calculated rates automatically balance but may not satisfy platform injection capacities.

If a platform cannot inject its own produced gas reinjection (due to injection capacity) then

> if ITARG is entered as fraction and NOREDUCEQ is not entered on the PTARG data line) reduce the platform's gas production accordingly. If NOREDUCEQ is entered for PTARG, the injection target will not be met, and excess gas will appear as “unaccounted gas”.

If a platform j cannot inject a transfer from platform i (due to platform j injection capacity), then

> (a) inject the transfer in platform i if i has enough injection capacity and if not,
>
> (b) if ITARG is entered as fraction and NOREDUCEQ is not entered on the platform i PTARG data line, reduce platform i production accordingly, and otherwise
>
> (c) if ITARG is entered as a fraction and NOREDUCEQ is entered for platform i PTARG, or if ITARG is entered as a rate of platform i gas, the injection target will not be met and excess gas will appear as “unaccounted gas”.

Platform hydraulics and pressure constraints (PLATTHPP, PLATTHPGI, PLATTHPWI, PLATTHPSI)

Entry of PLATTHPP or PLATTHPGI or PLATTHPWI or PLATTHPSI data for a given platform indicates that all production wells or gas injection wells or water injection wells or

SWAG injection wells assigned to the platform are connected to and flow through a common tubing, riser, or device, or any number of these in series, which is/are referred to as the platform production, gas injection, water injection, or SWAG injection tubing. Sensor can approximately apply hydraulics relationships and pressure constraints to the tubing. Although these constraints will not be exactly satisfied at the end of any timestep, they will be approximately satisfied as a function of time. A specified hydraulics table represents flow through the platform production or injection tubing. For platform production tubings (PLATTHPP), a minimum pressure constraint is specified at the outlet. For platform injection tubings (PLATTHPGI or PLATTHPWI, or PLATTHPSI), a maximum pressure constraint is specified at the inlet.

```
PLATTHPP (dmpfac)
iplat thpp np       (as many lines as desired)

PLATTHPGI (dmpfac)
iplat thpgi ngi     (as many lines as desired)

PLATTHPWI (dmpfac)
iplat thpwi nwi     (as many lines as desired)

PLATTHPSI (dmpfac)
iplat thpsi nsi     (as many lines as desired)
```

Definitions:

dmpfac damp factor, default .5

iplat platform number

thpp minimum platform production tubinghead pressure (at outlet of platform production tubing)

np THPTABLE number assigned to platform production tubing

thpgi maximum platform gas injection tubinghead pressure (at inlet of platform gas injection tubing)

ngi THPTABLE number assigned to platform gas injection tubing

thpwi maximum platform water injection tubinghead pressure (at inlet of platform water injection tubing)

nwi THPTABLE number assigned to platform water injection tubing

thpsi maximum platform SWAG injection tubinghead pressure (at inlet of platform SWAG injection tubing)

nsi THPTABLE number assigned to platform SWAG injection tubing

Notes:

1. Platform tubing pressure constraints are applied explicitly (approximately). If individual well THP constraints are violated (according to platform hydraulics, rates, and min/max tubing pressure), they are reset for use on the next timestep. In order to avoid oscillation, only dmpfac times the computed change is applied on each step. If oscillation in well thp values is observed, reduce the damp factor. When actively controlling flow (by resetting well THP constraints that are controlling) , platform tubing hydraulics will be only approximately honored in the solution of any given timestep.
2. The status of wells flowing on platform pressure constraints (flowing on well THP constraint determined by platform pressure constraint) is reported in the output tables as PP.
3. Any and all well THP constraint resets are reported in the output file in the PLATFORM THP RESET TABLE.
4. Individual well hydraulics tables assigned through THP data should represent the flow of the individual wellstreams from or to the bottomhole datum (the 'bhp' or 'pump' perf) to or from the single common junction of the well and platform tubings. If the well hydraulics tables represents only flow between the true wellheads and bottomhole (i.e. they do not include the pipes connecting the individual wellheads to the platform tubing), then the pressure drops between the true wellheads and the platform tubing junction are assumed negligible.
5. Either production or injection tubing hydraulics tables may be assigned to platform production or injection tubings. In a hydraulics table assigned to a platform production tubing, thp is the outlet pressure and bhp is the inlet pressure. In a hydraulics table assigned to a platform injection tubing, thp is the inlet pressure and bhp is the outlet pressure.
6. Gas lift is not implemented for the platform production tubings. Any gas lift data entered in the tables assigned to the platform production hydraulics tables is ignored. However, gas lift for the wells remains functional and is appropriately accounted for (the wells gas lift rates are included in the platform gas production rates used in the platform tubing hydraulics calculations).

Platform examples

Example 1:

The wells are assigned to two cosmetic platforms, representing two reservoirs or areas in the total grid. Wells W5 and G10 are excluded from the platforms.

```
WELLPLAT
  P1 - P11  1
  G1 - G4   1
  W1 - W2   1
  P12 - P25  2
  G5 - G9   2
  W3 - W4    2
YPLAT 1
   .08 .83 .07 .02 0 0    ! 6-component problem
YPLAT 2  SURFACE    ! this is default and need not be entered
RATE
```

```
  P1 - P11  80000   ! max mcf/d rate for wells P1 - P11 inclusive
  P7  40000
  P3  25000
  P12 38000
  P13 41000
   ... etc - all drilled wells ..
COSMETIC 1
COSMETIC 2
```

Default well rates at time 0 are -1 (undrilled, or shutin with no recirculation). Therefore, any wells not listed under RATE will have a "not drilled yet" status. Any wells entered under RATE with rates of 0 are drilled - shutin with recirculation.

All the above data are remembered - none needs to be reentered later except to change it. If you wish to take a well previously assigned to a platform out of the platform, enter "wname 0" under WELLPLAT.

Example 2:

Change the data in Example 1 to place the platforms on target rates beginning on January 1, 1996.

```
....
DATE 31 12 1995
PTARG 1 G  240000  ! platform 1 prodn target = 240000 mcf/d gas
PTARG 2 G  175000  !    "     2   "    "       175000    "
ITARG 1 G  1.0  1  ! inject 100 % of platform 1 net gas production
                   !   in platform 1
ITARG 1 G  .3  2   ! inject 30 % of platform 2 net gas
                   !  production in platform 1
ITARG 2 G  .7  2   ! inject 70 % of platform 2 net gas
                   !    production in platform 2
ITARG 2 W  12000   ! inject 12000 stb/d water in platform 2
ONTIME 2  .8 .7 .8 ! ontimes for platform 2
FUEL 1 .04
FUEL 2 .055
SALE 2 40000
```

You should account for all the net gas in a (non-cosmetic) platform. (e.g. the platform 2 net gas production is distributed for injection into platforms 1 and 2 by fractions which sum to 1.0). If your fractions do not sum to 1.0, the model will still run and the gas balance deficit (excess) will show up as "unaccounted gas". You do not need to reenter any of the above data, except to change them.

Example 3:

Here we specify three platforms with gas production, outside gas injection, and transfer gas injection.

```
WELLPLAT
```

```
  P1 - P7  1
  G1 - G4  1
  W1 - W2  1
  P8 - P20 2
  G5 - G9  2
  W3 - W4  2
  P21 - P25 3
  G10       3
  W5        3
YPLAT 1  SURFACE  ! not necessary if this is 0 time
YPLAT 2  SURFACE  !        "
YPLAT 3  SURFACE  !        "
PTARG 1 G 140000
PTARG 2 G 210000
PTARG 3 G  40000
FUEL 1 .04
FUEL 2 .06
FUEL 3 .07
SALE 1 40000
OUTSIDE  1
  .. nc mol fractions ..  ! composition of outside gas 1
OUTSIDE  2
  .. nc mol fractions ..  ! composition of outside gas 2

ITARG 1 OUTSIDE 1 15000  ! inject 15000 mcf/d of outside gas 1
                         !     in platform 1
ITARG 3 OUTSIDE 2 7000   ! inject 7000 mcf/d of outside gas 2
                         !     in platform 3
ITARG 1 G  .4  1 ! inject 40 % of platform 1 net gas in platform 1
ITARG 2 G  .5  1 ! inject 50 % of platform 1 net gas in platform 2
ITARG 3 G  .1  1 ! inject 10 % of platform 1 net gas in platform 3
ITARG 2 G  1. 2 ! inject 100 % of platform 2 net gas in platform 2
ITARG 3 G  1. 3 ! inject 100 % of platform 3 net gas in platform 3
```

For each platform, there should be “ITARG iplat G f jplat” fractions (f) summing to 1.0 specifying injection of its net gas. However, this summation to 1.0 is not mandatory.

Example 4:

Table 5 shows the Platform Table printout for the following 3-platform data. Table 5 should help to clarify the gas balance/platform logic.

```
RATE
  P1 - P2  20000
  P3 - P9  60000
  G1 - G4  90000
```

```
PLATFREQ 0
OUTSIDE 1
    .85 .13 .02 4*0
WELLPLAT
  P1 - P2  3
  P3 - P5  1
  P6 - P9  2
  G1  1
  G2 - G4  2
ITARG 1  OUTSIDE 1  2000
ITARG 2  OUTSIDE 1  4000
PTARG 1 G 50000
PTARG 2 G 80000
SALE 1 10000
SALE 2  20000
FUEL 1 .02
FUEL 2 .02
ITARG 2 G  .5 1  ! NOTE: only 50 % of platform 1 net gas
                 !       production is injected - the other
                 !       50 % shows up as "unaccounted" gas
                 !       in the Platform Table printout.
ITARG 2 G  1.0  2 ! inject all of plat 2 net gas in plat 2
YPLAT 1  .78 .17 .04 .01 3*0
COSMETIC 3 ! NOTE: no PTARG,ITARG, etc data are entered for
           !       a cosmetic platform.
PRINTBAL 1 ! include printout of platform gas balance in
           !  Platform Table
TIME 10
```

Example 5:

The entry ITARG 1 G .7 1 means "inject in platform 1 70% of the net gas produced from platform 1". You can enter an injection rate rather than a fraction, e.g. ITARG 1 G 70000 1 means "inject in platform 1 70000 mcf/d of the net gas production from platform 1".

Tabulated here are the platform production and injection rates which result for several cases as the platform production rate becomes capacity limited.

```
PTARG 1 G  100000
ITARG 1 G  .7 1
```

If SALE 1 20000 entered

time	Qgross	Qnet	Qinj	Qnet	Qinj
0	100000	100000	70000	80000	56000
1000	100000	100000	70000	80000	56000
1200	90000	90000	63000	70000	49000
1500	70000	70000	49000	50000	35000

2000	20000	20000	14000	0	0

```
PTARG 1 G  100000
SALE 1 20000
ITARG 1 G  1. 10
```

time	Qgross	Qnet	Qinj	Sale
0	100000	80000	80000	20000
1000	100000	80000	80000	20000
1200	90000	70000	70000	20000
1500	70000	50000	50000	20000
2000	20000	0	0	20000
2500	10000	0	0	10000

```
PTARG 1 G 100000
ITARG 1 G 70000 1
```

				If SALE 1 20000 entered	
time	Qgross	Qnet	Qinj	Qnet	Qinj
0	100000	100000	70000	80000	70000
1000	100000	100000	70000	80000	70000
1200	90000	90000	70000	70000	70000
1500	70000	70000	70000	50000	50000
2000	20000	20000	20000	0	0

Only the second of these three cases satisfies the intended usage of the platform logic: 100 % of net gas produced is injected somewhere.

3.5 Transmissibility Modifications (MULTIPLY)

Standard neighbor transmissibilities can be modified in recurrent data:

```
MULTIPLY TX or TY or TZ
i1 i2 j1 j2 k1 k2 f    ! as many lines as desired
```

Current transmissibilities at time of entry are multiplied by the factor f (multiple modifications are cumulative).

Standard neighbor transmissibilities can also be modified recurrently, along with non-neighbor (fault) connection transmissibilities, as a function of stress or not, using the FAULTMOD option described in Section 1.5.

3.6 Time or Date Specification

The Recurrent Data proceed in chronological order in "time frames", where the entry of a TIME or DATE line marks the end of each time frame.

TIME card

Example:

```
C        START OF RECURRENT DATA
C   the following well data are effective at time 0
WELL
  .. well data
  .. other well keywords and data ..
TIME 365   ! proceed to time = 365 days
C   the following well data are effective at 365 days
    .. well data changes ..
TIME 1095  ! proceed to time = 1095 days
    .. etc., time frames of well data changes
       followed by TIME entries ...
TIME 12045
END
```

The format of the TIME entry is

TIME time dtime

where dtime is optional, with the default dtime = 0 (not used). Entry of dtime>0 effectively creates a TIME entry at multiples of dtime, for output purposes. In some cases, the frequency of various printout may be specified as each TIME entry. Then, even with no well data changes, numerous TIME entries might be specified just to obtain that printout.

Example:

```
.. previous data  ..
TIME 1460
  .. well data changes at 1460 days ..
TIME 1825
TIME 2190
TIME 2555
  ... etc. many yearly TIME entries ....
TIME 12045
END
```

The use of dtime gives the same printout result with the single TIME entry,

```
TIME 1460
 .. well data changes at 1460 days ..
TIME 12045  365
END
```

DATE card

If DATE entries are used in Recurrent Data then

```
DATE iday0 imonth0 iyear0
```

must be entered in Initial Data to specify the date of 0 time. Dates may be used instead of times as follows:

```
DATE iday imonth iyear  dtime
```

iday = day (1-31)

imonth = month (1-12)

iyear = year, e.g. 1976, 1990, 2002, etc.

To specify the last day of the month you can enter 31 for iday. If there are fewer than 31 days in the month, the program will reset the iday. The dtime option (dtime in days) may be used with DATE, as described above. For example, the following

```
  .. previous data ..
DATE 31 12 1998
  .. well data changes at Dec. 31, 1998 ..
WELLFREQ 1
DATE 31 12 2020 365
END
```

would print the Well Table at the end of each year from 1999 to 2020. As described later, the frequency of printout may be specified as monthly or yearly when DATE entries are used. The iyear integer entered with DATE may be truncated to two digits, e.g. 79 for 1979, 07 for 2007, etc.

3.7 Timestep Data

Timestep size is automatically determined in Sensor. Some default parameters are:

	Impes (1)	Impes (2)	Implicit	
first step at 0 time	2	2	10	days
maximum step	31	180	60	days
dxmax	.02	.1	.07	
dsmax	.05	.1	.07	
dpmax	200	1000	4000	psi

(1) CFL not entered

(2) CFL entered

The dxmax, dsmax, dpmax values are maximum changes per time step, used in the auto timestep logic. The model may reduce the first default timestep if changes are excessive.

Recommended timestep entries and timestep keywords are now described.

For Impes runs:

Either

1) enter no timestep control data,

or

2) enter CFL 2 (once only) before the first TIME or DATE line. This activates internal calculation and usage of the Impes stable timestep size, which will vary throughout the run. If this gives poor results, remove the CFL entry (see further discussion below).

For a history match period, you might try entering DTMIN v where the value of v (say 30 days) forces one step per rate-change period. In that case, do not enter CFL. At the end of history, enter DTMIN 0.

For Implicit runs:

Enter no timestep data, except possibly "DTMAX dtmax" to change the default maximum 60-day step, and "DT dt" at time 0 if you wish to start the run with a step different from the default 10 days. Enter DTMIN v where v is a suitable value only if numerous "closely" spaced time entries are entered for reasons of rate changes or printout.

Entry of any of

DT, DTSTART, DTMAX, DTMIN, DXMAX, DSMAX, DPMAX, MINITN, MAXITN

is not recommended in general and should be viewed as an exception.

First time step specification (DT)

The DT dt entry is occasionally desirable at the beginning of a new recurrent data time period when large changes in well rates occur. The entry

```
DT dt
```

results in one timestep of dt days starting the new time frame, followed by auto timestep selection. However, if dt is entered negative then a constant timestep of |dt| days will be used. This use of negative dt is highly inadvisable in general.

Examples:

This results in one four-day step from 730 to 734 days, followed by auto timestep selection from 734 to 1095 days:

```
  .. previous data ..
TIME 730
  .. well data changes ..
DT 4
```

```
TIME                                                    1095
 …….
```

The following results in constant two-day steps from 0 to 360 days:

```
  .. previous data, 0 time ..
DT -2
TIME 360
  ...
```

Global first time step specification (DTSTART)

The entry

```
DTSTART dt
```

results in one timestep of dt days starting each new time frame, followed by auto timestep selection to the end of the time frame. This is useful in some history match datafiles or other datafiles where there are frequent TIME entries with large well rate changes each new time frame. The DTSTART entry needs entering only once and is active until it is reentered with a new value or until DTSTART 0 is entered to deactivate it. Its default value is 0 (not used).

Example:

```
   .. previous data, 0 time ..
DTSTART 5
TIME 31
  .. well data changes ..
TIME 61
  .. well data changes ..
TIME 92
  .. well data changes ..
TIME 122
  .....
```

This results in a first 5-day step from 0 to 5 days, followed by auto step selection from 5 to 31 days, a 5-day step from 31 to 36 days, followed by auto step selection from 36 to 61 days, etc. Any entry of DT dt will override the DTSTART value only in the time frame of the DT entry.

The principle here is that the DTSTART dt entry is remembered throughout all time frames, while the DT dt entry is not.

Maximum time step (DTMAX)

The entry

```
DTMAX dtmax
```

limits the timestep to a maximum of dtmax days. Default values are 31 for Impes and 60 days for Implicit.

Auto time step controls (DXMAX, DSMAX, DPMAX)

The default values of dxmax, dsmax, dpmax given previously can be changed by the entries:

```
DXMAX dxmax   ! change in mol fraction over the timestep
DSMAX dsmax   ! change in saturation, fraction
DPMAX dpmax   ! pressure change, psi
```

Impes stable timestep (CFL)

The entry

```
CFL
```

activates logic which determines and uses the Impes stable timestep size. The entry sets Δtmax to 180 days and resets larger Impes default dxmax, dsmax, and dpmax values as noted in the above Table. You can reset those values if and as desired, using the corresponding keywords. The entry sets the parameter f, which is the desired multiple of stable step size. The default f value for the above entry is 1.5 for 2D and 3D problems, and 1.0 for 1D problems. You can override these default f values by using the entry

```
CFL f
```

An f value of 2.0 is frequently found desirable in 2D and 3D problems. For 1D problems, f should not be entered different from 1.0, except for research purposes. The CFL entry is not used in Implicit problems. If this entry gives poor results, remove it.

If you have previously entered CFL and wish to deactivate it, enter NOCFL. The NOCFL restores dxmax, dsmax, dpmax, dtmax values to the values listed for Impes above.

If you enter CFL:

You may enter any of DT, DTSTART, DTMAX, and DTMIN.

Normally, no DT, DTSTART, or DTMIN entries would be used. A DTMIN v entry may force time steps larger than the calulated stable values, defeating the purpose of CFL and resulting in an unstable run.

We recommend against any entry of DXMAX, DSMAX, or DPMAX after CFL, although there are exceptions and any such entry which helps is obviously ok.

If the recommended CFL 2 entry is used, check for oscillations or unstable behavior. If such is detected, try the entry CFL or its equivalent, CFL 1.5. In rare cases, CFL 1. may be required.

To see printout of entered CFL data, edit your outfile and search for CFL.

Minimum time step (DTMIN), not recommended in general

The entry

```
DTMIN dtmin
```

sets the minimum timestep to dtmin days. With the exception discussed below, in general DTMIN should not be used in Impes runs. The default dtmin value is 0.

DTMIN can be used in Impes runs in history match periods where rates are entered at regular (say) 30-day intervals. In some such cases, the most efficient run results from entering DTMIN 30 at 0 time, forcing 30-day steps, one per rate period, throughout the history period. If entered, DTMIN 0 should be entered at the end of history to reset dtmin to 0.

Minimum and maximum Newton iterations (MINITN, MAXITN)

Each time step requires one or more Newton iterations to converge. The default parameters are:

	Impes	Implicit
minitn	1	1
maxitn	4	8

We recommend against changing minitn or maxitn. The entries are:

```
MINITN minitn
MAXITN maxitn
```

If convergence is not attained in maxitn iterations, the program will cut the timestep size and repeat the step's calculations. A moderate number of timestep cuts is of no consequence - timestep cuts do not affect results. The cumulative number of timestep cuts is printed in the Timestep Table.

Example:

Implicit datafile spe2.dat is a r-z black oil coning problem, the 2nd SPE Comparative Solution Project problem. It takes 11 steps and 31 Newton iterations to 900 days, producing an average oil rate of 610 stb/d. Replacement of the timestep data by

```
.. Initial Data ..
.. well data ..
TIME 900
END
```

represents an attempt to take one 900-day timestep. The data are incomplete for that purpose, as follows: (a) the default first step is10 days, (b) the default maximum step is 60 days, (c) the default of maxitn=8 is not enough Newton iterations to converge with the large 900-day timestep. Adding to the above data as follows

```
.. Initial Data ..
.. well data ..
DT 900
DTMAX 900
MAXITN 60
TIME 900
END
```

results in convergence in 42 Newton iterations for the single 900-day step. It is interesting that the run converges with the well on pressure constraint producing 608 stb/d.

3.8 Restart Records and Runs (RESTART, RESTARTFILE)

The program writes restart records to unit 18 (file “fort.18”) and reads restart records from unit 17 (file “fort.17”). To write a restart record from which a subsequent restart run may continue the simulation, simply insert the keyword RESTART immediately preceeding the time at which a restart record is desired.

Example:

This writes a restart record at 10220 days:

```
  .....
TIME  10128
RATE
  W11   0.
  W12 5000.0
RESTART
TIME  10220
RATE
  W11 1000.0
  W12   0.
TIME  10311
  .....
END
```

The program will print the following type message for each restart record written:

**RESTART RECORD WRITTEN AT DATE = 31 12 1995 ITIME = 348 TIME = 10220.0 DAYS

You can edit your outfile and search for "RESTART" to find these messages. For the above example, the datafile for a subsequent restart run beginning at 10220 days would be:

```
TITLE
   FIELD X RESTART RUN FROM 10220 DAYS
     .....
ENDTITLE
RESTART  348 !  time step number of desired restart record
C    RECURRENT DATA START HERE (no ENDINIT entry)
RATE
  W11 1000.0
  W12   0.
TIME  10311
  .....
END
```

The timestep number specification may optionally be replaced by the date at which the restart file is written, i.e. the restart run may specify either

```
RESTART  itime
```

or

```
RESTART  iday imonth iyear
```

where itime is the timestep number, or iday, imonth, iyear are the day, month, and year at which the restart file was written, and iyear is the 4-digit year.

The restart file written by the original run, fort.18, must be moved or copied to file fort.17 before submitting the restart run. Or, the restart file may be copied to a file of any arbitrary name and specified using the RESTARTFILE entry entered prior to entry of RESTART. For example, the entries

```
RESTARTFILE restart1.rst
RESTART  348
```

specify to restart the run using restart file restart1.rst from timestep 348. The restart file name may include no path (it is local to the execution directory) or it may include a relative or full path to the file. For example, restartfilename might be ..\restart1.rst, or C:\sensor\run1\restart1.rst. If the file name or path contains blanks, enclose restartfilename in quotes.

Restart records may be created in fort.18 by the restart run, i.e. if you forget to move or copy the original restart file before submitting the restart run, the original restart file will be lost.

A sufficiently excessive number of RESTART entries may fill or overflow disk storage. For the entry sequence

```
RESTART
TIME 10400 365
```

one restart record will be written at time = 10400, i.e. dtime entries do not apply to writing of restart records.

3.9 End of Run Card (END) mandatory

The keyword END terminates the third, Recurrent Data, section of the datafile and is the last keyword in the file.

Example:

```
.......
... Initial, Modification and Recurrent data ...
.......
TIME  5475
END
```

4 Output Control

Sensor output consists of:

1. A standard output file (named on execution line) capable of presenting virtually all aspects of Sensor results and performance.
2. Map file fort.71 for 2D and 3D visualization of fixed and time-dependent properties and results (given in Table 14) on the reservoir grid in SensorMap and 3rd party mapping packages.
3. Plot file fort.61 for creating x-y plots in SensorPlot and 3rd party plotting packages. Results are included for wells, platforms, regions, superregions, total field, and tracer fractions.
4. RFT file fort.51 reporting pressures and saturations for specified gridblocks at specified times.
5. PLT file fort.52 containing information for perforations in specified wells at specified times.
6. Platform Summary (PSM) file fort.74 containing information for platforms, wells, and perforations.
7. Potential file fort.10 containing well potential rate information.
8. Extended composition file fort.23 reporting extended composition information for EXTEND options.
9. Restart file (see Section 3.8).
10. COMP file fort.24 reporting compositional information (compositional runs only) for wells, regions, superregions, platforms, and the total field.
11. DIM file fort.29 giving dimensions that can be used for variable allocation within third-party or user-provided applications which read the Sensor binary results files.

4.1 Output File

There are two types of printed output in the printed output file. First, in-line printout, repeated at specified times, includes a Timestep Table, a Well Table, a Region Table, a Platform Table, and maps or arrays. Second, end-of-run (eor) summaries are normally printed for each well, each platform, each region, and the total field. The timestep summary is the last eor summary printed.

See Chapter 6 for a list of search strings that can be used to quickly locate particular sections of the output file.

The Timestep Table, Platform Table, and Well Table printout are shown in Tables 6, 7, and 8, respectively. Table 9 shows the end-of-run summary printout. The field total gaslift rate (mcf/d) and cumulative gaslift gas (mmcf) are printed in the Timestep Table.

The Well Table includes a column of fractional flow (rb/day basis) by perf. This perf (layer) fractional flow is q_k divided by the total well inflow over all perfs, where q_k is total rb/d rate for the perf. The bottomhole wellbore pressure is printed as equal to grid block pressure for shutin wells which are not recirculating.

Status codes of wells and platforms

It is often helpful to view the *status codes* of wells in the Well Table and of platforms in the Platform Table – and in their end-of-run summaries.

We define the term *target rate* before discussing the status codes of wells and platforms. Target rate is the rate assigned to a well or platform. It is a maximum rate and is the requested rate. A limiting bottom-hole (tubinghead) pressure bhp(thp) is also entered for each well using keywords BHP(THP). A *bhp well* is one where bhp is entered. A *thp well* is one where thp is entered.

The target rate for a well may be (a) the rate entered under keyword RATE, (b) a rate assigned by the pressure control (PCON) logic, or (c) a rate assigned by platform target rate allocation logic. In any case, the well target rate is not greater than the rate entered under keyword RATE. The actual well rate will be the target rate or a lesser pressure-constrained (PC) rate constrained by bhp or thp.

Well status code

The Well Table includes a two-character status code for each well:

SI shutin due to entered rate of 0, or LIMITWELL

Q on target rate

PC on bhp or thp pressure constraint

DD on drawdown pressure constraint

NF no-flow, the reservoir pressure is too low(high) for the producer(injector)

Active wells have rates > 0 and status codes of Q or PC or DD. *Inactive wells* have zero rates and status codes of SI or NF. The platform has separate status codes for production, gas injection, and water injection. A platform production(gas injection)(water injection) status code of SI means that all production(gas injection)(water injection) well rates are 0.

Platform status code

SI shutin due to entered rate of 0, LIMITPLAT, or reservoir pressure level

Q on target rate

PC constrained by all active wells on bhp, thp, or drawdown pressure constraint

RC constrained by all active wells on their entered maximum rates

RP constrained by some wells on PC and others on RC

QW constrained by platform water rate limit

QG constrained by platform gas rate limit

IC production rate is reduced due to gas injection capacity

The platform and well status codes are also printed in their end-of-run summaries.

In default mode, each eor summary has one line for each timestep. Thus the eor summary printout can be voluminous for a run having thousands of timesteps. This printout can be controlled by omitting certain eor summaries and/or by reducing the frequency of lines in the summaries.

The eor summaries are printed with time in days. If you have used DATE in the run and want eor summary results printed vs date then see the keyword PRINTSUM below.

Frequency of Printout (STEPFREQ, WELLFREQ, PLATFREQ, MAPSFREQ, SUMFREQ, COMPFREQ)

The Frequency Integer nn

The freqency integer nn is defined here and in Table 13 as follows:

nn	frequency
0	every time step
1	only at times (dates) entered in datafile
n > 1	every n th time step and at times (dates) entered in datafile
-n (n > 1)	every n th time step and at times (dates) where the word PRINT appears on the TIME (DATE) dataline
-1	never (off)

The frequency of printout can be controlled by the Recurrent Data entries:

```
                        default nn
STEPFREQ nn              !  1   Timestep Table printout
WELLFREQ nn              !  1   Well Table printout
COMPFREQ nn              ! -1   Surface gas composition for each well
MAPSFREQ nn              ! -1   Maps printout
PLATFREQ nn              !  1   Platform Table printout
SUMFREQ  nn              !  0   frequency of lines in eor summaries, also
                         !      controls frequency of write to plot file
```

where nn is the frequency integer of Table 13. These entries may be reentered to change printout frequencies throughout the run. You might include the above table with "C" in column 1 in each of your datafiles immediately prior to the first TIME or DATE Recurrent Data entry. It will serve as an easy reminder of default output frequencies.

(MONTH, YEAR)

An option here, if DATE entries are used, is that nn may be MONTH or YEAR. If nn is MONTH, printout occurs at DATE entries of the form:

```
DATE 1 imonth iyear
DATE iday imonth iyear
```

where iday is 31 or the last day of month imonth (if you intend the last day of the month, you can use iday=31).

If nn is YEAR, printout occurs at DATE entries of the form:

```
DATE 1 1 iyear
DATE 31 12 iyear
```

(PRINT)

Usage of the word PRINT on the TIME (DATE) dataline together with large negative nn can shorten the datafile in cases like the following example.

Example:

Assume default values of 1 for STEPFREQ and WELLFREQ, and 0 for SUMFREQ. The datalines

```
 MAPSFILEFREQ -1
TIME 3285.  365.
 MAPSFILEFREQ  1
 MAPSFREQ 1
TIME 3650.
 MAPSFILEFREQ -1
 MAPSFREQ -1
TIME 6935.  365.
 MAPSFILEFREQ  1
 MAPSFREQ 1
TIME 7300.
```

can be equivalently entered as

```
MAPSFREQ     -10000
MAPSFILEFREQ -10000
TIME 3650.  365.  PRINT
TIME 7300.  365.  PRINT
```

In this example, Timestep and Well Tables would be printed each year and the eor summaries would have one line for each time step. Maps would be printed and written to file at 10 years and at 20 years.

Reduction of Lines in eor Summaries

The Recurrent Data entry

```
SUMFREQ nn   ! nn is the frequency integer
```

controls the line frequency in eor summaries and the write frequency for the PLOT file, in accordance with the value of nn. The default value of nn is 0 (write every step). This may be reentered to change nn during the run.

Omission of eor summary printout (WELLSUM, PLATSUM, REGSUM, SREGSUM, FIELDSUM, TIMESUM, PRINTSUM)

Selected members in well, platform, region, and superregion eor summaries can be omitted, and field and timestep summaries can be omitted entirely, using the following keywords in Recurrent Data:

```
WELLSUM
  wname  m        ! eor well summaries, wname may be a range
                  ! or a negative integer between -1 and -4
PLATSUM
  iplat  m        ! eor platform summaries
REGSUM
  ireg   m        ! eor region summaries
  ireg1 - ireg2 m
SREGSUM
  isreg  m        ! eor superregion summaries

FIELDSUM m        ! eor field summary
TIMESUM  m        ! eor timestep summary
```

The value m = 0 means omit and the value m = 1 means include. Default is m = 1. As many lines as desired may appear after each of the first four keywords. An option with PLATSUM, REGSUM, and SREGSUM is entry of " -1 m" which means "assign the m value to ALL platforms (or regions or superregions)".

In addition there is a global printout control entry in Recurrent Data. It can be used to completely eliminate specific summaries from the printout, regardless of any frequencies, omissions, or inclusions specified above.

```
PRINTSUM mwell mplat mreg msreg mfield mtime! 6 integers, each 0,1 or 2
```

Each of the mwell,mplat,..etc. is either 1 or 2, to print the summary, or 0 to not print the summary. For example, if mwell is 0, no well summaries will be printed. 1 indicates to print the summary versus time in days, and 2 indicates to print the summaries versus date (applies only if dates were entered, rather than time, in the Recurrent Data). This control is convenient when the summaries are numerous and lengthy, and the files (from which they would be printed) are desired only for purposes of post processing graphics. The default values of these six integers are 1.

Inclusion of 0-rate lines in eor well summaries (PRINTZERO)

The eor well summaries exclude lines of 0 rate. This saves many printout lines of 0's. If you enter in Recurrent Data

```
PRINTZERO n
```

with n = 1, all 0-rate lines will be printed. You can enter and reenter this; n = 0 omits zero-rate lines. The default is n = 0.

Region and superregion table printout (PRINTREG)

A Region Table, including Superregions, is printed at 0 time, showing initial in-place fluids. A Region Table, including Superregions, is printed with each Time Step Table, with frequency controlled by STEPFREQ. An example is shown in Table 6. This Region Table consists of three subtables. The first table gives field rates and cumulatives. The second gives initial and current water, oil, and gas in place and region recoveries defined as (initial in place - current in place) / initial in place. The third subtable gives mols and % recovery of each component (in compositional cases only).

The Region Table printout is controlled by the Initial orRecurrent Data entry:

```
PRINTREG n
```

n = 0 omit the Region Table printout

n = 1 include the first subtable

n = 2 include the first two subtables

n = 3 include all three subtables (this is the default)

Map Printout

All data described here for keywords WINDOWS, MAPSPRINT, MAPSX, and MAPSY may be entered in Initial Data and in Recurrent Data. Any such data entered in Initial Data are "remembered" and need not be reentered, except to change, in Recurrent Data.

Printing many maps very frequently can increase cpu time appreciably. It also renders the outfile difficult to edit and view. In particular, mol fraction maps are generally of little use. Try to set your MAPSFREQ values (default -1 or "off") to eliminate unnecessary map printing.

Map windows (WINDOWS)

Maps are printed in windows. A window is a defined portion of the grid. To specify windows, enter

```
WINDOWS
  iwin  i1 i2  j1 j2  k1 k2  form
  .. as many lines as desired ..

    iwin = window #
    i1 i2 etc. define the window in terms of grid ijk indices
    form = one of XYZ,XZY,YXZ,YZX,ZXY,ZYX
```

The program by default sets window #1 as the entire grid with a map format of:

```
XYZ   for 3D problems, y printed down the page
XZY   for 2D xz cross-sections
YZX   for 2D yz cross-sections
```

If you enter no WINDOWS data, this default window #1 will be the single window. You can change this default definition of window 1.

The map is printed with the first dimension of "form" printed across the page and the second dimension printed down the page. If you want the second dimension printed increasing up the page, then put "UP" after "form".

Example:

The entry

```
WINDOWS
  2   3 10  15 27  2 7   XYZ  UP
```

will print y increasing up the page in the defined window 2.

Entered windows must be given numbers 1,2,3,..,nwin where nwin is the total number of windows entered. They can be entered in any order. For example, if you specify three windows, they should be given numbers 1,2, and 3. You can add windows and change existing windows throughout Recurrent Data.

Select maps to print (MAPSPRINT , TABLE)

This keyword is used to select which maps to print, or to deselect maps previously selected, for specific windows or for all windows. It can be specified in the Initial Data and can be specified repeatedly in the Recurrent Data.

This keyword is used to set the print status of 1 (on) or 0 (off) for the maps. The maps are printed in the order they are listed in the list of eligible mapnames given in Table 14. The one-line entry

```
MAPSPRINT  iwin   mapname1 mapname2 mapname3 ... mapnamen
 !  iwin = window number, e.g. 3, or a range of window numbers,
 !         e.g. 2 - 6 (the dash must have 1 or more blanks either
 !                     side of it)
 !  mapnamei = one of the eligible mapnames
```

causes the n listed maps to be printed in the specified window(s). If iwin is omitted, the listed maps are printed in all windows. For multiple windows, the order of printing is: window 1, all listed maps, window 2, all listed maps, etc. Do not continue the mapname list on to the next line (start the next line with MAPSPRINT iwin .. continuing mapnames if necessary).

The entry

```
MAPSPRINT OFF iwin mapname1 mapname2 ... mapnamen
```

deactivates printing of the listed maps in window(s) iwin, or in all windows if iwin is omitted. If no maps are listed, then all maps are deactivated in window(s) iwin. Thus the entry MAPSPRINT OFF deactivates all maps in all windows. These print controls are remembered; you need not reenter them except to change them.

For one-dimensional (1D) problems, we recommend the initial data entry

```
MAPSPRINT TABLE
```

which results in a single printed table that contains the most commonly needed arrays. If the table does not include certain arrays that you want, do not use the TABLE option.

Notes:

1. A given map may be printed in any number of windows.
2. The positions of active well cells are noted on the map printout by the symbols "*" for production and "#" for injection.
3. Missing (inactive) cells appear as blank as illustrated in Table 10.
4. You do not need the Recurrent Data entry "MAPSPRINT OFF iwin ..mapnames" in connection with invariant map names - the program prints them only at 0 time. Any Recurrent Data entry of invariant map names is ignored.
5. The printed value of psat is 0 for any block where kvtable is active because psat is not calculated or used in such blocks.
6. In compositional cases of gas injection, a grid block may follow a path to the right hand side of the pressure-composition (p-Z) diagram where no p_{sat} exists. In such a case, the printed p_{sat} value may be 0 or the last successfully calculated p_{sat}. In either event, it is a meaningless value which is no longer used and affects no results.
7. The printed oil (gas) phase viscosity is 0 if the oil (gas) phase mobility is 0.
8. If S_{gr} is everywhere equal to S_{gc}, S_{gmax} is not used and the SGMAX map is not printed.
9. In compositional problems where parachors are not entered or TENSION is deactivated, the TENS map values are $Max(i)|x_i-y_i|$. The values printed or written to file are equal to -0.99 for all water cells, negative in undersaturated hydrocarbon-cells (x_i is a dewpoint composition or y_i is a bubblepoint composition), and positive in 3-phase cells. When the value of this quantity is less than .05, the grid block is very close to critical.
10. If the mapnames for component mole fractions X or Y are specified, no mole fraction maps will be printed unless the components to be printed are selected using the MAPSX or MAPSY data, given in the following section.

Select component mole fractions to print (MAPSX, MAPSY)

The one-line entry

```
MAPSX iwin   i1 i2 i3 .. ! i1, i2, i3 .. = component numbers
```

activates printing of individual oil-phase component mol fractions in window iwin. The entry MAPSY iwin i1 i2 i3 .. works the same way with gas-phase mol fractions y_i. The default print status is "off" for all oil-phase and all gas-phase mol fractions. For example, for a problem with $n_c = 6$ components, the entry

```
MAPSX 1  1 6
```

specifies inclusion in window 1 of only x_1 and x_6 in the oil-phase mol fractions printed by entry of the mapname X in the MAPSPRINT 1 data list. If MAPSPRINT OFF is entered, printing is deactivated for all component mole fractions specified in the MAPSX and MAPSY data.

Map format controls (MAPSLINES, MAPSFULL, MAPSFORM)

The optional entry

```
MAPSLINES 0    !  suppress blank lines
```

suppresses blank lines. The entry MAPSLINES 1 includes them. Default is 1 (inclusion). The optional entry

```
MAPSFULL 1
```

prints full arrays with values of 0.0 for inactive grid blocks. The default MAPSFULL 0 prints blanks for inactive grid blocks. The entry MAPSFULL 2 is the same as MAPSFULL 1 except arrays of a constant value v are printed with the one-line printout "All Values = v". See the last three sections of Table 10 for examples of printout with MAPSLINES 0 and MAPSFULL 1 entered.

All maps are printed using a format of 10F10.d, i.e. 10 numbers across the page. The number of decimal places, d, varies from map to map. The default values of d are as follows:

default d	maps
1	DELX,DELY,P,PSAT
2	H,HNET,KX,KY,KZ
4	POROS, all saturations, mol fractions, and tracer fractions
4	DENO,DENG,VISO,VISG,TENS,KRW,KRO,KRG
0	ROCKTYPE,COMPACTYPE (integers)

These defaults can be changed by using one or more of the one-line entry

```
MAPSFORM  d mapname1 mapname2 mapname3 ...
```

in Initial Data and/or in Recurrent data. Entry of d does not change the field width (10) or the number of numbers printed across the page (10).

The maps PV,TX,TY,TZ,HCPV,PVSOF use G formats and should not need to be changed. If you enter d for them, their format will be changed from G to F10.d.

For one-dimensional (1D) problems, the MAPSPRINT TABLE option described in an above section can be used to consolidate printed values for the most commonly needed arrays into a single table.

Example:

Assume a 42 x 33 x 10 grid with 6 components.

```
.. in Initial Data ..
DATE 1 1 1988   ! Jan 1, 1988

MAPSX 1  1 6     ! include only x1,x6 in X mol fractions printed
MAPSY 1  1 2 6   ! include only y1,y2,y6 in Y mol fractions printed
MAPSPRINT 1  DELX DELY DEPTH H HNET POROS PV TX TY TZ ROCKTYPE
MAPSPRINT 1    REGION COMPACTYPE P PSAT SW SO SG X Y
C  (The default window 1 is 1 42  1 33  1 10 XYZ)
```

```
.. in Recurrent Data ..
C   (no map printout until Dec 31 1997)
.. first WELL and other well data
C  default MAPSFREQ is -1 so no maps will be printed
DATE 31 12 1995  ! Dec 31, 1995
.. changes in well data ..
WINDOWS
  2   1 19  1 33  1 10  XYZ UP  ! print y increasing up the page
  3  20 42  1 33  1 10  XYZ
  4    4 6  21 23  2 7  XZY
MAPSPRINT OFF    ! turn off all maps in all windows
MAPSPRINT 2 - 3  P SW SO SG  X
MAPSX 2 - 3  1 6  !  print x1 and x6 only in windows 2 and 3
MAPSPRINT 3  TRACER
MAPSPRINT 4  P SW SG PSAT
MAPSFREQ 1       ! print maps at time of each TIME (or DATE) entry
DATE 31 12 1997
   .... etc. ...
```

Print well data (PRINTWDATA)

Numerous entries of WELL, PIMULT, WELLTYPE, BHP can make it difficult for the engineer to know or recall his many well property settings at various times through the simulation. The Recurrent Data entry PRINTWDATA causes printout of all wells and their properties at the time of that entry. The printout is easily found by editing the outfile and searching for "PRINTWDATA". The Recurrent Data entry is as follows:

....

TIME 1200

... data ...

PRINTWDATA

TIME 1700

... data ...

TIME 2400

....

This activates printout of well data which are in effect for the period 1200-1700 days for all wells. To see this well data printout, edit your outfile and search for "PRINTWDATA".

Suppress printout of well data (PRINT WELL 0)

Enter PRINT WELL 0 to suppress printing of Well Input Data Tables to the output file when well-related data are input. Enter PRINT WELL 1 to re-activate printing of the tables. Default is to print the tables (n=1).

Time, rate, and cumulative scaling in eor summaries (TIMESCALE, FSCALE)

Geologic-time problems may extend to times the order of hundreds of millions of days. To avoid outfile printout of **** as time in the end-of-run (eor) summaries in such cases, you may enter

```
TIMESCALE tscale  ! where tscale = .001 or .0001 or ...)
```

in the Initial Data This will cause the times listed in the eor summaries to be printed as (actual time in days) * tscale.

Also, lab experiment problems may extend only to times the order of .01 days. In such cases you can enter a tscale value of (say) 1000 to see times listed in the eor summaries.

A related entry is FSCALE. The Recurrent Data entry

```
FSCALE fscale ! (e.g. 1000 or any value you wish)
```

prints all rates and cumulatives in the eor FIELD SUMMARY as actual values * fscale.

Miscellaneous printout controls (PRINTKR, PRINTTHP, PRINT MISSING, PRINT SHARED)

PRINTKR n ! print the first n Saturation Tables, default n = 1. If n is omitted, all
! entered tables are printed. This is an Initial Data entry.

PRINTTHP n ! print the first n tubinghead pressure tables entered,
! n = 1 default; n=0 suppresses thp table printout; omission
! of n results in printing of all tables. This is a Recurrent Data entry.

PRINT MISSING n ! if n=0, print no warnings for perforated (well) gridblocks that are missing
! (inactive), if n=1, all warnings will be printed. Default is n=1.

PRINT SHARED n ! if n=0, warnings for perforated (well) gridblocks that are shared by two or
! more active wells will not be printed, if n=1, all warnings will be printed.
! Default is n=1.

4.2 Map File (MAPSFILE, MAPSFILEFREQ)

Sensor writes maps to Unit 71 (file “fort.71”) for 2D and 3D visualization purposes. The user can write a program to read and process Unit 71 results as required for visualization purposes by the package of his choice. This fort.71 file is the file used by Map2Excel and entered by the keyword FILE of the interface program SensorMap.exe. The Fortran write statements creating fort.71 are given in the SensorMap Manual. Maps are written to this file in accordance with your entry of the keywords MAPSFILE and MAPSFILEFREQ. The keyword MAPSFILE can be entered in Initial and Recurrent Data as

```
MAPSFILE  mpname1 mpname2 mpname3 ...
```

That entry specifies maps which are to be written to Unit 71. Table 14 gives the eligible mapnames. The list cannot continue on to the following line (repeat MAPSFILE on each additional line as necesary). If you wish to deactivate the writing of certain maps previously

entered, then use the above entry with OFF appearing after MAPSFILE. For example the Recurrent Data entry

MAPSFILE P SW OFF

deactivates the writing of the maps P and SW to the Unit 71 file. The entry MAPSFILE OFF with no listed mapnames will deactivate writing of all maps.

The default status of writing is off for all maps. The frequency of map writing is controlled by the Recurrent Data entry

MAPSFILEFREQ nn

where nn is the frequency integer of Table 13 or can be MONTH or YEAR if dates are used (see pg 97). Use this frequency entry to discontinue the writing (i.e. nn = -1) or you may overflow disc capacity. The default frequency is -1.

4.3 Plot File (SUMFREQ)

Sensor writes results to Unit 61 (file "fort.61") for xy plotting purposes. All results are included for wells, platforms, regions, superregions, total field, and tracer fractions – no user input is required to specify which results to write. The user can write a program to read and process Unit 61 results as required for plotting by the plot package of his choice. This fort.61 file is the file used by Plot2Excel and listed under the keyword FILE of the interface program SensorPlot.exe. The Fortran write statements creating fort.61 are given in the SensorPlot Manual. Results are written to file in accordance with your setting of the frequency control "SUMFREQ nn", where nn is the frequency integer of Table 13. For example, the default SUMFREQ nn value of 0 gives results written each time step; an entry "SUMFREQ 10" would result in writing results to fort.61 every 10 time steps. SUMFREQ also controls the frequency of printout in the end-of-run summaries. Use a frequency integer of -1 to discontinue writing.

4.4 RFT File (RFT)

RFT reports (as many as desired) consisting of specified gridblock pressures and saturations at specified times can be written to Unit 51 (file "fort.51") by entering

```
RFT rftname ! character*12
i j k d     ! as many lines as required
```

where

i,j,k = gridblock indices

d = block center depth in ft, enter only if k=0 to automatically compute k

The report is written only at the time that the RFT data are specified.

The format of each report in the RFT file is given by the fortran write statement

```
WRITE (51) RFTNM,TIME,IDAY,IMONTH,IYEAR,NRFT,HISTNAME
DO L=1,NRFT
  WRITE (51) I(L),J(L),K(L),ZTOP(L),ZBOT(L),P(L),SW(L),SG(L)
ENDDO
```

```
where
          RFTNM     RFT test name, character*12
          TIME      time, days
          IDAY      date
          IMONTH     "
          IYEAR      "
          NRFT      number of gridblocks
          HISTNAME  name of history file containing observed data,
                    character*132
          I(L)      I index of the L th block
          J(L)      J index of the L th block
          K(L)      K index of the L th block
          ZTOP(L)   depth at the top of the block, ft
          ZBOT(L)   depth at the bottom of the block, ft
          P(L)      pressure of the block, psia
          SW(L)     water saturation in the block, fraction
          SG(L)     gas saturation in the block, fraction
```

Example:

```
        ...
        TIME 1240
        WELL
          ... data ...
        RATE
          ... data ...
        ... any other data ...
        RFT  RFT1     ! character*12
          23  13 3    ! I J K gridblock indices
          23 13 5
          23 13 0  8312
          23 24 7
          23 25 8
           ... etc. as many lines as desired ...
           Note: If any K index is entered 0, then the program
                 calculates K from the block-center grid block depth,
                 entered as the fourth number, ft.
        TIME 1310
        ...
```

4.5 PLT File (PLT)

Perforation reports for specified wells at specified times can be written to Unit 52 (file “fort.52”) by entering

```
PLT
 wname1 ! well name, character*8
```

```
    wname2
    ...    ! as many lines as required
```

The report is written only at the time that the PLT data are specified.

The format of each report in the PLT file is given by the fortran write statement

```
  WRITE (52) WNAME,TIME,IDAY,IMONTH,IYEAR,NPERF,IRATE,IWELL
  DO L=1,NPERF
    WRITE (52) IPERF(L),JPERF(L),KPERF(L),ZTOP(L),ZBOT(L),FRAC(L),PI(L)
  ENDDO
where
          WNAME     well name, character*8, left justified
          TIME      time, days
          IDAY      date
          IMONTH     "
          IYEAR      "
          NPERF     number of perfs
          IRATE     welltype integer, see Table 12
          IWELL     internal well number (1,2,3,..,nwtotal)
          IPERF(L)  I index of the L th perf
          JPERF(L)  J index of the L th perf
          KPERF(L)  K index of the L th perf
          ZTOP(L)   depth at the top of the perforated gridblock, ft
          ZBOT(L)   depth at the bottom of the perforated gridblock, ft
          FRAC(L)   % of the total well rate from perf L, sum=100.0
          PI(L)     well index of the L th perf, rb-cp/d-psi
```

Example:

```
          ...
          TIME 4114
          ... data ...
          PLT
            WELL10    ! well name, character*8
            WELL1
            WELL3 - WELL6
            WELL9
              ... etc.
          TIME 4222
          ...
```

4.6 Platform Summary File (PSM)

```
C        SENSOR FILE WRITE (VERSIONS DATED JUNE 1 2002 OR LATER)
C        WRITE PSM DATA TO FORTRAN UNIT 74
C
```

```
C      Keyword PSM (no data) must be entered in Initial Data and/or in
C       Recurrent Data for this file write to occur.
C
C      ** THIS IS A ONE-TIME ONE-LINE WRITE AT THE BEGINNING OF
C         FORTAN UNIT 74.  YOU CAN USE THESE MAX DIMS ON PLATFORMS,
C         WELLS,PERFS PER WELL, AND NC TO ALLOCATE ARRAYS SO YOU DON'T
C         HAVE TO READ THIS FILE TWICE, ONCE TO SCAN TO GET MAX DIM'S,
C         THEN AGAIN TO PROCESS.

      WRITE (74) MAXPLAT,MAXWELL,MAXPERF,MAXNC  ! ONE-TIME WRITE

C      FOLLOWING WRITE IS REPEATED AT END OF EACH TIME STEP

      WRITE (74) TIME,ITIME,IDAY,IMONTH,IYEAR,NWT,NPLAT

      DO IP=1,NPLAT
        WRITE (74) OTP(IP),OTWI(IP),OTGI(IP),FUEL(IP),SALE(IP)
      ENDDO

      DO IW=1,NWT
        WRITE (74) IW,WNAME(IW),NPERF(IW),IRATE(IW),IPVTW(IW),IPLAT(IW),
     1             NC(IW),QWELL(IW),QW(IW),QO(IW),QG(IW),L(IW),
     1             (X(I,IW),I=1,NC(IW)), (Y(I,IW),I=1,NC(IW))

        IF (QWELL(IW) .GT. 0) THEN
          DO L=1,NPERF(IW)
           WRITE (74) L,IPF(L,IW),JPF(L,IW),KPF(L,IW),IPVT(L,IW),
     1     P(L,IW),(Q(I,L,IW),I=1,NC(IW)),QWATER(L,IW),QOIL(L,IW),QGAS(L,IW)
          ENDDO

        ENDIF
      ENDDO

C     TIME       DAYS
C     ITIME      TIME STEP #
C     IDAY  IMONTH  IYEAR      DATE (E.G. 15  4  1975)
C       (NOTE: THESE 3 DATE INTEGERS WILL BE 0 IF KEYWORD "DATE"
C              WAS NOT ENTERED IN THE SENSOR DATAFILE)
C     NWT        CURRENT # OF WELLS
C     NPLAT      CURRENT # OF PLATFORMS
C     OTP        ONTIME FOR PRODUCTION (FRACTION BETWEEN 0 AND 1)
C     OTWI       ONTIME FOR WATER INJECTION
C     OTGI       ONTIME FOR GAS INJECTION
C     FUEL       MCF/D FUEL USE FOR PLATFORM
C     SALE       MCF/D SALE FOR PLATFORM
```

```
C     WNAME      WELL NAME, CHARACTER*8
C     NPERF      # OF PERFS FOR WELL
C     IRATE      WELLTYPE - SEE DOCUMENTATION
C     IPVTW      PVTTYPE FOR WELL
C     IPVT(L,IW) PVTTYPE FOR PERF #L OF WELL #IW
C     IPLAT      PLATFORM # OF WELL, 0 IF WELL IS NOT IN A PLATFORM
C     NC         # OF HYDROCARBON COMPONENTS IN WELLSTREAM
C     QWELL      WELL ENTERED RATE:  -1  SHUTIN WITH NO RECIRC
C                                     0. SHUTIN WITH RECIRC
C                                    >0  ACTIVE FLOWING WELL
C     QW QO QG   WATER (STB/D), OIL (STB/D), GAS (MCF/D)
C                WELL SURFACE RATES
C     L(IW)      L FOR WELL IW WELLSTREAM, MOLS SURFACE OIL PER MOL OF
C                WELLSTREAM (0 < L < 1, FLOATING POINT, APPLICABLE ONLY
C                FOR COMPOSITIONAL PRODUCERS)
C     X(I,IW)    MOL FRACTION CPT I IN SURFACE OIL FROM (PRODUCTION)
C                WELL IW
C     Y(I,IW)    MOL FRACTION CPT I IN SURFACE GAS FROM (PRODUCTION)
C                WELL IW
C     IPF JPF KPF   I J K INDICES OF PERF
C     P          GRID BLOCK P OF PERF, PSIA
C     QWATER     WATER INFLOW TO WELLBORE RATE, STB/D, FOR PERF
C  **FOR COMPOSITIONAL WELLS .....
C     Q(I,       TOTAL MOLS OF COMPONENT I FOR THE PERF,
C                   POSITIVE FOR INFLOW TO THE WELLBORE, NEGATIVE
C                   FOR OUTFLOW
C  **FOR BLACK OIL WELLS
C     Q(I,       Q(1)=STBOIL/D Q(2)=MCF/D INFLOW RATES FOR THE PERF
```

4.7 Well Potential File (WELCAP)

If the keyword WELCAP is entered in Recurrent Data, production well capacities (open flow potentials or rates) will be written to Unit 10 (file “fort.10”) at the beginning of each time step. The format of this write is as follows (the write statement is in a DO loop over all production wells):

```
          WRITE      (10,100)      TIME,ITIME,WNAME,IPLAT,QO,QG,QW,QO1,QG1,QW1
    100 FORMAT(F10.1,I6,1X,A8,I3,3X,3G14.6,3X,3G14.6)
```

TIME = time, days

ITIME = timestep number

WNAME = well name

IPLAT = platform number to which well is assigned (0 if none)

QO,QG,QW = the rates calculated for the well target rate equal to that entered under RATE.

QO1,QG1,QW1 = the capacity (or open flow potential) rates.

Water and oil rates are stb/day, gas rates are mcf/day. Enter WELCAP only if and when you want this write to begin. For example, generally the write is not desired during the history period. Also, the last write will be at the beginning of the last time step so no well capacities will be written at the last entered TIME of the datafile. Therefore, if WELCAP has been entered, add a timestep at the end of the datafile in order to compute capacities at end of run.

Example:

An 18000-day run has 7000 days of history followed by an 11000-day prediction period. Well capacities are desired in the prediction period. The datafile should be as follows:

```
  ...  Initial Data and 0-7000 day Recurrent Data ...
TIME 7000
WELCAP
  ... recurrent data changes ..
TIME 8000
  ... recurrent data changes ...
  ... etc. ...
TIME 18000
TIME 18010
END
```

4.8 Extended Composition File

When the EXTEND option is specified, Sensor writes total field molar rates and extended component mole fractions to Unit 23 (file “fort.23”). See Section 5.2 for description of the EXTEND option and file format.

4.9 Restart File

See Recurrent Data, Section 3.8, Restart Records and Runs.

4.10 COMP File

The binary COMP file, Unit 24 (file “fort.24”), is written only for runs (including restart runs) in which all fluid types are compositional, and contains compositional injection and production information for wells, regions, superregions, platforms, and the total field.

```
C      ** WRITE COMPOSITIONS TO FORTRAN UNIT 24
C      (COMPOSITIONAL ONLY - NOT BLACK OIL)**

C  ** ONE-TIME WRITE AT TOP OF FILE – DIMENSIONS FOR ALLOCATION,
```

```
C     AND COMPONENT NAMES **
C
        WRITE (24) NREGIONS,NSREGIONS,NPLATFORMS,NWELLS,NC
        WRITE (24) (CNAME(I),I=1,NC)  ! CHARACTER A4

C  ** FOLLOWING WRITE AT END OF STEP ON FREQUENCY OF
C     SUMFREQ (DEFAULT=0=EVERY STEP) **
C  ** WRITE IS FOR ALL REGIONS, SUPERREGIONS, PLATFORMS, WELLS, AND FIELD, IN
C     THAT ORDER, WHERE CUMI MOLS INJECTED + CUMP MOLS PRODUCED > 0 **

C   IYEAR4    4-DIGIT YEAR
C   IYEAR2    2-DIGIT YEAR
C   ITIME       TIMESTEP #
C   TIME        DAYS

C    QI(I)   MOLS/D CPT I INJECTION RATE
C    QIT     MOLS/D TOTAL INJECTION RATE
C    QP(I)   MOLS/D CPT I PRODUCTION RATE
C    QPT     MOLS/D TOTAL PRODUCTION RATE
C    CUMI(I) CUM MOLS CPT I INJECTED
C   CUMI     CUM TOTAL MOLS INJECTED
C   CUMP(I)  CUM MOLS CPT I PRODUCED
C   CUMP     CUM TOTAL MOLS PRODUCED

C  **GENERAL WRITE

    WRITE (24) CLASS,NAME,NUMBER,
      (QI(I),I=1,NC),(CUMI(I),I=1,NC),
      (QP(I),I=1,NC),(CUMP(I),I=1,NC),
      QIT,CUMI,QPT,CUMP,
      ITIME,TIME,IDAY,IMONTH,IYEAR4,IYEAR2

C    CLASS = ONE OF: REGION,SREGION,PLATFORM,WELL,FIELD
C    CLASS         NAME       NUMBER
C   REGION       RGNAME    REGION #
C   SREGION      SRGNAME   SUPERREGION #
C   PLATFORM       ' '     PLATFORM #    (' ' = BLANK)
C   WELL         WNAME     WELL #
C   FIELD          ' '     1             (' ' = BLANK)
```

4.11 DIM File

The following one-time write for each run, including runs from time zero and restart runs, is made to Unit 29 (file “fort.29”), giving dimensions that can be used for variable allocation within third-party or user-provided applications which read the Sensor binary results files. Those dimensions following NC may change on each restart, if the user has changed the number of entities defined in the data.

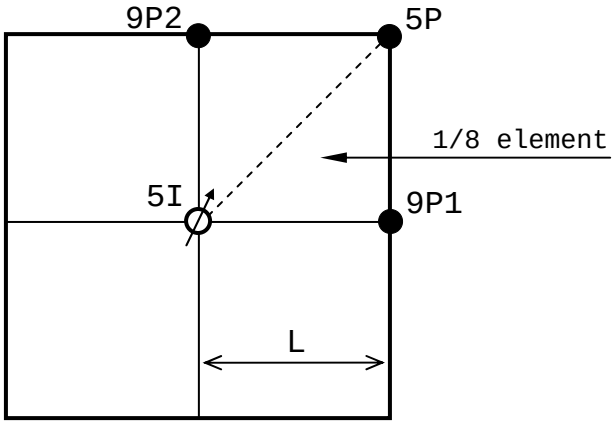
```
      WRITE (29) NX,NY,NZ,NB,NBB,NRES,NPVT,NC,NWMAX,
                 NPWMAX,NPERFMAX,NREGMAX,NSREGMAX,NPLATMAX,NDUAL

C       NX NY NZ  GRID DIMENSIONS
C       NB           # ACTIVE BLOCKS
C       NBB          # TOTAL BLOCKS (Nx * Ny * Nz)
C       NRES         # RESERVOIRS
C       NPVT         # PVT TYPES
C       NC           # COMPONENTS (EXCLUDING WATER)
C       NWMAX        # WELLS
C       NPWMAX   MAX # PERFS PER WELL
C       NPERFMAX     # TOTAL PERFS
C       NREGMAX      # REGIONS
C       NSREGMAX     # SUPERREGIONS
C       NPLATMAX     # PLATFORMS
C       NDUAL        0=SINGLE POROSITY, 1=DUAL POROSITY, 2=DUAL PERM
```

5 Special Features

5.1 Pattern Flood Simulation (EDGE, ELEMENT)

Any problem involving a symmetrical square (x-y) area can be run using a 1/8 element. For example, this applies to a 5-spot or 9-spot. The 1/8 element results and the more expensive ¼ element results are mathematically identical. The figure below illustrates the full, ¼ and 1/8 patterns. Wells 9P1 and 9P2 are present for a 9-spot, absent for a 5-spot.

5-spot or 9-spot Pattern



Simulation of symmetrical 5- or 9-spot pattern elements can be performed using either full or half blocks along the edges. Impes run cpu time will be reduced by a factor of about 2 if full blocks are used. Also, use of the minimal 1/8 symmetrical element rather than the 1/4 element will reduce run cpu time by a factor of about 2. Thus a 1/8 element run using full edge blocks will be about four times faster than a 1/4 element run using half blocks on the edges.

You may not know the correct well index (Pi or J) for a pattern element dataset, depending upon the grid and difference scheme used. A single one-layer model run can be made to determine the well indices for your dataset. This is discussed in Appendix 10.

The model calculates pore volumes and transmissibilities using full blocks. Enter

```
EDGE  (no data)
```

in Initial Data if you want them calculated for half blocks along the four xy edges of the grid. The EDGE entry does not alter any PV,TX,TY, or TZ which is entered as an array.

If you want a 1/8 pattern element with a diagonal grid, for a 5-spot or 9-spot:

(1) prepare a 1/4 pattern datafile with N_x=N_y, the 5-spot injector and producer at i,j = (1,1) and (N_x,N_y), respectively, and the same grid spacing used in the x and y directions.
(2) enter ELEMENT
(3) enter EDGE if you want half edge blocks

(4) enter NINEPOINT if you want the 9-point difference scheme
(5) you need no MODIFY data

Datafile test12.dat illustrates ELEMENT for a 1/4 9-spot. Results are identical with and without entry of ELEMENT. The run is two times faster with ELEMENT entered.

The grid recommended for simulating a pattern flood:
a) represents 1/4 or 1/8 of a 5-spot or 9-spot pattern.
b) correctly represents wells located at the corners of the grid.
c) requires no modification of edge or corner pore volumes or transmissibilities.

If the pattern is A acres then the 1/4 pattern grid is a square with side length equal to L = sqrt(A x 43560)/2. $\Delta x(i)$ and $\Delta y(j)$ are constant and equal to L/N where $N_x=N_y=N$. Note that a (say) 10x10 xy grid without EDGE entered is equivalent to an 11x11 grid with EDGE entered. Variable spacing may be used if desired as discussed in Appendix 10. The internally calculated well indices are not correct if full edge blocks or variable grid spacing is used. In those cases, one model run can be made to determine the correct indices as discussed in Appendix 10.

Entry of ELEMENT results in full blocks along the x- and y-axes of the 1/8 element. If you want half blocks on those edges, then also enter EDGE - your Impes cpu time may double if EDGE is entered. Enter all well rates as those corresponding to a 1/4 pattern. If you want a datafile which will run either a 1/4 or a 1/8 pattern then do the following. Under keyword WELL, enter all wells on or below the diagonal first, followed by all, if any, wells above the diagonal. If you want a datafile which only runs the 1/8 pattern, then enter only wells on or below the diagonal.

With the above-described datafile, a) if you run without the keyword ELEMENT in Initial Data, a 1/4 pattern run results with calculated rates and cumulatives corresponding to the 1/4 pattern., b) if you enter ELEMENT in Initial Data, identical results will be obtained with 1/2 the cpu and storage; calculated rates and cumulatives correspond to 1/8 of a pattern.

Use of ELEMENT is allowable only if the 1/4 pattern is symmetrical about its diagonal. This use of ELEMENT applies to any square-grid problem, pattern or not, which is symmetrical about its diagonal. ELEMENT can be used only with a diagonal grid - not with a parallel grid. Either unsymmetrical well placements or constraints about the diagonal or heterogeniety which is unsymmetrical about the diagonal can invalidate ELEMENT.

Any number of layers and vertical heterogeneity are permissible. Areal heterogeneity is ok for the 1/4 pattern but if ELEMENT is used, the areal heterogeneity must be symmetrical about the diagonal.

The datafiles spe79692.dat and the runs using 10x10x10 and 40x40x10 grids in misc.dat are 1/4 five-spot runs. You can reduce the cpu times of those runs by a factor of about two by adding the single entry ELEMENT after GRID. Do not change any other entries. The results will be identical except for the reduced cpu time.

5.2 Extended Component Description (EXTEND)

An extended component fluid description may be entered in black oil and in compositional problems. All fluid types must be either black oil or compositional, they cannot be mixed. If EXTEND is entered, then the Timestep Table printout includes an extended recovery table showing cumulative production and injection split out for each of the Extended Components. These results approximate the extended component recoveries which would be computed in a more expensive simulation using all the extended components as model components. Do not use this option if the EXTEND option was specified on a BLACKOIL entry.

For black oil cases, the format is

```
EXTEND BO
     cpti ! i=1,Nce  Nce 4-character extended component names on one dataline
     xi   ! Nce mol fractions, composition of stock tank oil
     yi   ! Nce mol fractions, composition of surface gas
```

The Nce xi (yi) values do not have to be on a single dataline.

The format for compositional cases is demonstrated by the following example.

Example:

The nc=6 model components are P1,P2,P3,P4,P5,P6.

```
  EXTEND
  CO2  C1  N2  C2  C3  C4  C5  C6  C7  C-10 C-15 C-20
  .02 .60 .01 .11 .08 .06 .04 .03 .02 .015  .01  .005  ! composition
    P1   N2 C1       ! nc lines defining the model
    P2   C2 CO2 C3   ! components in terms of the
    P3   C4          ! Extended Components.
    P4   C5 C6
    P5   C7 C-10
    P6   C-15 C-20
```

All component names must have 4 or fewer characters. The number of extended components must be 24 or less. The composition is that of original reservoir fluid.

If EXTEND is entered, Sensor writes a file (Fortran Unit 23) giving the molar rate and extended composition of total field production. The following is a one-time write at the beginning of the run:

```
   WRITE (23) NCE,       ! no. of extended components
  1           NC,        ! no. of model components
  1   (CNME(I),I=1,NCE), ! extended component names, character*4
  1   (CNAME(I),I=1,NC)  ! names of model component names, character*4
```

The following is written at the end of each time step:

```
        WRITE (23) ITIME,TIME,DT,QMOLF,(Z(I),I=1,NCE)

           ITIME = timestep no.
            TIME = time, days
              DT = time step, days
```

```
QMOLF = mols/day rate of total field production
 Z(I) = extended composition of total field production
```

Many compositional model studies are performed using 10-15 or more components. With proper pseudoization, most 10-15 component model runs can be reproduced more efficiently using 5-8 component runs - i.e. the same gor, wcut, pressure behavior, recoveries, etc., are obtained. This EXTEND feature offers a means of comparing (say) 7- and (say) 15-component runs. Run the 15-component model and obtain calculated recoveries of all 15 components individually (printed in the Timestep Table). Then make the 7-component run with EXTEND and compare the extended 15-component recoveries obtained with those from the first (15-component) run. Of course there will be some difference. The question is whether the difference is of engineering significance. In some cases, the difference in component recoveries may be used to argue the need for (say) 15-component simulation. However, even when this is the case, the difference in gor, wcut, etc. between 7 and 15 component runs may be small. Most of the study time and expense is spent in history matching and sensitivity runs where this reservoir performance (wcut,gor,pressure behavior,oil and gas recoveries) is the major concern. Thus, a much less expensive study might result from using few-component runs for most of the study and (say) 15-component runs for the final prediction runs, giving the plant engineers more wellstream definition for their purposes.

5.3 Prevention of Gassy Well Blowout (QGMAX)

This feature is occasionally useful in history match studies of "gassy" oil reservoirs[31]. The QGMAX entry is in Recurrent Data.

```
QGMAX
  wname  qgmax
  .. any number of lines ..
```

Example:

```
QGMAX          ! optional
 -4   25000  ! max gas production rate qg = 25000 mcf/d for all wells
3 - 9  20000  ! max gas production rate qg = 20000 mcf/d for wells 3-9
```

Default qgmax is 0 (not used) for all wells. Entry of a 0 value deactivates a previously entered value. If qgmax is positive, the well's gas rate q_g is checked against qgmax each iteration of each step. If q_g exceeds qgmax, the well is converted for that step/iteration only to a gas production well with a target rate of qgmax mcf/d. For a gas production well of type MCF (Mcf/d producer), any entered qgmax is not used. QGMAX data can be entered for injectors but it is used only for producers.

5.4 Initial Water Saturation Overread (SWINIT)

An additional array entry is SWINIT. If this array is entered in Initial Data then initial water saturations calculated by capillary- gravitational equilibrium are replaced by SWINIT array values which are 0 or positive. Enter the array value negative (e.g. -1.) for blocks where you do not want the replacement. This replacement can give inconsistencies and significant reservoir

fluid flow with no well activity. We recommend against use of this array, except in research type problems.

Swi values entered in the SWINIT array should normally lie between the block Swc value and 1.0. . If Swi < Swc, a warning message is printed near the top of the outfile and the program internally resets grid block Swc to Swi if theentered Swi is less than Swc. Any SWINIT array entries > 1.0 are internally reset to 1.0.

Also, any entered SWCG array value (Baker krw rel perm) > SWINIT array value is reset downward to the SWINIT array value.

5.5 Laboratory Experiment Simulations (LAB)

Enter

LAB

in Initial Data for lab-scale problems. Times printed in the outfile and in the end-of-run summaries will be in hours. If you prefer times printed as minutes, use the entry

LAB MINUTES

The printout will use units as follows:

CC	CUBIC CENTIMETERS
RCC	RESERVOIR CC
SCC	STD CC (GAS)
MSCC	1000s SCC
STCC	STOCK TANK CC
RATES	STCC/HR AND MSCC/HR
CUMULATIVES	STCC AND MSCC
IN-PLACE	STCC AND MSCC
GOR	SCC/STCC

Input datafile units are unchanged. The default first time step is .0001 days. See the test problem datafile test8.dat as an example.

When LAB is entered, the program internally sets default values of various parameters to much tighter values than used in field-scale problems:

DXMAX	DSMAX	DPMAX	TOLP	First Timestep	Maximum Timestep
.02	.02	50 psi	1 psi	.0001 days	10 days

Most lab-scale problems will run better with entry of IMPLICIT. You may enter D4; do not enter NF or ILU. Do not use DATE.

5.6 Fetkovich Aquifers (AQUIFER, AQAREA, CONNECTHC, PRINTAQ) (Reference 45)

We recommend that any connected aquifers be included in the reservoir grid rather than using this option.

As many aquifers as desired may be entered. The Initial Data input for a given aquifer is as follows:

```
C            AQ #       Vp       ct      Po      DEPTH      J
AQUIFER       1        9.E9    8.E-6    2200     4900    40000

   i  j  k  (DIR)  (Ti)              ! as many lines as required
or
  i1 i2  j1 j2  k1 k2   (DIR)   (Ti) ! as many lines as required
```

where

```
  Vp        = aquifer pore volume, rb, constant
  Ct        = aquifer compressibility, cw+cf, 1/psi, constant
  Po        = initial aquifer pressure, psia
  DEPTH     = aquifer depth, ft, constant
  J         = aquifer total PI, rb-cp/d-psi, constant
  i,j,k     = grid block indices
  DIR       = optional label, either X, Y, or Z
  Ti        = optional aquifer connection coefficient for block i
```

If Ti is omitted or entered 0, it is by default calulated internally as block i (kA)x or (kA)y or (kA)z depending upon whether DIR is X or Y or Z, respectively. If DIR is omitted, it is taken as X. If you wish the program to calculate the Ti based on face area, rather than on permeability-area product (for all aquifers), then enter the keyword AQAREA before any AQUIFER data (this is not recommended):

```
AQAREA
```

If Ti is entered it is taken as (kA) product (or as the face area if AQAREA is entered), and entry of DIR is not needed or used. So Ti is either entered (>0) or internally calculated (if not entered or entered 0) for all blocks i and is used as follows:

The aquifer connection transmissibility for block i is (Ti/sum(Tj)) * J rb-cp/d-psi.

Call the above transmissibility simply Ti hereafter.

Block i water influx (or efflux) is

```
  qi = Ti * (bwo/visw) * (P + gw * (Zi-DEPTH) - pi) stb/d
```

where

```
  bwo       = water formation volume factor, stb/rb, calculated at Po
              using water data on the MISC line.
  visw      = value on the MISC dataline, cp, constant.
  gw        = water density, psi/ft
```

```
Zi        = block i depth, ft
P         = current aquifer pressure, changing with time, psia
pi        = grid block pressure, changing with time, psia
```

At time 0 the following calculation is done:

If the maximum over all blocks i value of Po + gw * (Zi-DEPTH) - pi is > 0 then the value of Po is reduced by that value and the aquifer is active from time 0.

If at time 0 the maximum value of Po + gw*(Zi-DEPTH) - pi is < 0 then the aquifer is inactive at time 0 and remains inactive until the first time step that the maximum value (over all blocks i) is positive. Aquifer pressure P remains constant at Po in this case until the aquifer is activated. If the aquifer pressure is properly initialized such that it is in equilibrium with all connected blocks below the HWC, and if the aquifer is properly connected such that no connections to blocks above the HWC exist, this check will have no effect.

In either of the above cases, there will be no flow or changes at time 0 due to the aquifers in the absence of wells. Both influx and efflux of water are accounted for.

If the aquifer pressure is properly initialized such that it is in equilibrium with and only connected to blocks below the HWC, the above checks will have no effect, i.e. the aquifer will be initially in equilibrium with all connected blocks. The program default is to reject any aquifer connections to gridblocks having initial mobile hydrocarbon phases, since such connections create an equilibrium impossibility. The number of such rejected connections are noted in the output file. Other simulators may allow arbitrary aquifer connectivity. If you wish to connect the aquifer to cells without regard to equilibrium or their fluid content (we highly recommend against this), enter the CONNECTHC keyword before any AQUIFER data:

```
CONNECTHC
```

If CONNECTHC is specified, and if hydrocarbon blocks are connected to the aquifer, the calculations described above will prevent any initial non-equilibrium situation in the model, at the expense of questionable model behavior. If such a situation were possible in a real reservoir, it would represent a severe non-equilibrium situation.

The program by default prints the first 10 connected gridblocks for each input aquifer. That number can be changed with the Initial Data entry

```
PRINTAQ n
```

For an active aquifer,

```
Wen+1 = Wen + sum(qi * dt) and Pn+1 = Po - Wen+1/(Vp * bwo * ct)
```

where

```
We  = water influx, stb
n   = time step index
qi  = water influx rate for block i, stb/d
dt  = time step, days
```

Encroachable water influx is Po * Vp * bwo * ct stb, representing the stb influx corresponding to depleting the aquifer to 0 pressure.

6 Aid to Viewing Program Printout

The easiest way to view program results is to edit the printed output file and search for items of interest. The following table is meant to be of assistance in that regard.

Search for	to skip to printout of:
WARN	warning messages
ERROR	error messages
PVT	entered pvt data
SWT	entered relative permeability and capillary pressure data
SEP	separator data
EQUIL	calculated initial psat value
BLACK OIL	black oil pvt data (black oil cases)
PERME(ABILITY)	relative perm amd P_c data tables
CONCAVE	printout relating to concavity of gas rel perm
KRIFT	IFT-related data
DUAL	dual porosity data
WELL IN	well input data
PRINTW	well information printed due to PRINTWDATA directive
INJECTION GAS	compositions of injection gas entered by INJGAS
PLATFORM DATA	platform input data
TUBING	tubinghead pressure table data
COMPAC	compaction table data
IN PLACE	initial average pressure and fluids in place
MAP	maps printout
M/M	table of min, max, and average values of grid properties
(STEP	timestep table
REGION TABLE	region table
PLATFORM TABLE	platform table
WELL TA(BLE)	well table
CFL	IMPES stable step input
STABLE	stable step multiplier being used
*DATE	well events, such as workovers, shutins, drills, etc.

*WORKOVER	well workovers
WAG	WAGTBL and CYCLETABLE entries
CONVERT	CONVERT input data and results
OPENWELL	wells opened due to OPENWELL directive
PCON	pressure control data
RIG and DRILL	data and actions for drilling schedule option
AQ #	Fetkovich aquifer data/results
TRACER DATA	tracer input data
TRACER TABLE	well table showing tracer fractions
FSALT	salinity option output
SHUTIN	times of platform and/or well shutins
SURFACE GAS	surface gas composition by well, triggered by entry of COMPFREQ
RESTART	notification of writing of restart record
PLATFORM n SUM	platform n end-of-run summary
ALL PLAT	end-of-run summary for all platforms
STEP SUM	end-of-run timestep summary
FIELD SUM	end-of-run field summary
*WELL	well workovers and shutins (LIMITWELL)
*PLAT	platform shutins (LIMITPLAT)
*EOR	the beginning of the eor summaries

7 Running Hints

7.1 Run Control

With the exception of the next paragraph, Impes problems should be first tried using the entry CFL 2 (or 1 for 1D problems) in Recurrent Data, with no other timestep control entries.

If time entries in a datafile are numerous and "closely" spaced, you should try entry of DTMIN v and not use CFL. The value v is problem dependent. This forces one step per rate period and may reduce cpu time significantly. For example if the numerous time entries range from .2 to 20 or more days apart, try DTMIN 5 or 10. If the time entries are one month apart (history period), try DTMIN 30.

For Impes problems, the model attempts to automatically calculate maximum stable time step sizes either

1) using default changes in pressure, saturation, and mol fraction over the time step as a guide, if CFL is not entered,

or

2) using stability theory and (larger) default changes as a guide, if CFL is entered.

In general, DTMIN should not be used in Impes problems, except as discussed above. If an entered DTMIN is always less than the stable step size calculated by (1) or (2) above, then DTMIN never affects anything and is unnecessary. If not, then DTMIN forces a step size above the stable value and Impes problems may run poorly or not at all if step size is forced sufficiently larger than the stable value. If you enter DTMIN in an Impes problem and see poor run behavior, remove the DTMIN entry.

In some cases the NF solver may be significantly faster than the default ILU solver.

Some Impes problems, primarily gassy oil problems, will run poorly if PERC is not entered to control gas percolation. In other Impes cases, entry of PERC will not help and will just increase cpu time.

Impes problems with sufficiently high maximum T_z value will run poorly or not at all. The Initial Data entry of TZMAX v where v is a problem dependent maximum z-direction transmissibility (say 1000 or 3000 ..) may be required. Edit your outfile and search for "M/M" to see the maximum transmissibilities and the maximum non-neighbor transmissibility. Enter TRFMAX v in Initial Data, if you wish to limit the non-neighbor transmissibilities to a maximum value of v rb-cp/day-psi.

Some problems, especially Impes problems, may run poorly if the smallest block pore volume (rb) is very small. Edit your outfile and scan for "M/M" to see this value. If it is small in a normal field scale problem, say 20 or 10 or 1 or .004,..., consider using CUT in MODIFY PV data or simply enter PVCUT v in Initial Data. If the "M/M" table shows very small minimum grid block thickness (e.g. .03 ft), you might try entry of THCUT v which will zero pore volumes of any blocks having net thicknesses less than v ft.

For Implicit problems, the default dtmax is 60 days. In some implicit problems, entry of “DTMAX dtmax” where dtmax is 90,120,180, or 360 may run ok and reduce cpu time.

Occasionally, long prediction-period runs may run to large times with low or even zero total field rate. To avoid this cpu burning, always enter LIMITFIELD in long prediction-period runs.

7.2 Platforms

In problems using platforms, the platform production wells are normally the same type as the platform production rate unit, G, O, or WG. For example if the platform target rate is mcf/d then the platform production wells should be gas wells. For problems where the initial hydrocarbon column is a near-critical oil or gas-condensate (perhaps both, due to composition variation with depth), the normal production well and platform production rate unit is mcf/d. If field target rate FTARG is used then the field production rate unit would normally be the same as the platform and production well rate unit. The above notwithstanding, it is ok to have a gas production well completed in an oil leg or an oil well completed in a gas column.

Calculated platform production rates should be close to entered platform production target rate (PTARG) when the platform production is on target rate. However, when some of the platform producers are on pressure constraint, the calculated platform production rate will show some deviation from the entered rate. If in an Impes problem using platforms, you have entered CFL, then look at the maximum pressure changes per time step in the end of run Timestep Summary (at the end of the outfile). If the pressure changes per step are large and platform production rate deviation is large (when the platform is not capacity limited due to well constraints - see the eor Platform Summaries), then you might try

```
CFL
DPMAX 300   ! change in p over step, used in calculating step size
```

or some other suitable dpmax value. If CFL is not entered, the default dpmax (pressure change per time step) is 200 psi; if CFL is entered, the default dpmax is much larger.

The calculated platform production rate is correctly lower than entered rate when either

1) all producers are on pressure constraint

or

2) the sum of rates of wells on pressure constraint plus the sum of entered maximum well rates for the other producers is less than the entered platform production rate.

7.3 Slimtube Runs

Slimtube runs are performed with small pressure gradients. Normally, one well is placed on pressure constraint, the other on rate constraint. A sufficiently large permeability is used to minimize the pressure gradient. Total tube resistance changes from step to step, depending upon the extent of two-phase hydrocarbon zones and rel perm and viscosity data. This step to step resistance change may be small or large. The result is that we can obtain a smooth production rate vs time or a smooth injection rate vs time, but not both. Using CFL=1 (stable step control) in the 1D slimtube impes runs, any such oscillations in injection or production rate stem from these tube resistance changes, not from time-step instabilities.

7.4 Compositional vs Black Oil CPU Time

For the same dataset, compositional cpu time is frequently assumed or observed to be much higher than black oil cpu. If compositional results differ greatly from black oil results, this is a moot question - i.e. comparing apples to oranges. In some cases, compositional and black oil results are similar, either because no gas is injected or because the gas injection, for various reasons, causes little difference in results. We will discuss only those cases where compositional and black oil results are similar. Let R = the ratio of compositional to black oil cpu time.

Impes Runs

An upper limit on R is 1/Fs, where Fs is the fraction of total cpu time spent in the solver in the compositional run. This upper limit assumes that black-oil non-solver cpu time is a negligible fraction of compositional non-solver cpu time. For example, a recent case gave Fs=.31 in a seven-component, impes compositional run where there was no gas injection. The black oil run gave identical results. The upper limit R = 1/Fs = 3.2 compared with the actual R of 1.43. This actual R is generally much less than the theoretical 1/Fs ratio because the assumption that non-solver black oil cpu is a negligible fraction of compositional non-solver cpu is a poor assumption.

Implicit Runs

For Implicit runs, compositional/black oil cpu ratio tends to be very large because nearly all of the compositional run cpu time is spent in the solver (simultaneously solving n = 8 or 10 or .. equations per grid block). This is far more expensive than the black oil solver's solution of (say) three equations per grid block.

8 References

1. Coats, K.H. Thomas, L.K., and Pierson, R.G., "Compositional and Black Oil Reservoir Simulation", SPE Paper 29111, presented at the 13th SPE Symposium on Reservoir Simulation, San Antonio, TX, Feb. 12-15, 1995.
2. Coats, K.H., “A Note on Impes and Some Impes-Based Simulation Models”, SPEJ (September 2000) 245-251
3. Coats, K.H., “Impes Stability: The CFL Limit”, SPE 66345, presented at the 16 th SPE
   Symposium on Reservoir Simulation, Houston, TX, February 11-14, 2001
4. Coats, K.H., “Impes Stability: The Stable Step”, SPE 69225, ibid
5. Vinsome, P.K.W., "Orthomin, an Iterative Method for Solving Sparse Sets of Simultaneous Linear Equations", SPE Paper 5729, presented at the SPE Fourth Symposium of Numerical Reservoir Simulation, Los Angeles, CA, Feb 19-20, 1976.
6. Appleyard, J.R. and I.M. Cheshire, "Nested Factorization", SPE Paper 12264 presented
   at the Seventh SPE Symposium on Reservoir Simulation, San Francisco, CA, November 15-18, 1983
7. Price, H.S., and Coats, K.H., "Direct Methods in Reservoir Simulation", SPEJ (June, 1974) 295-308; Trans. AIME 257.
8. McCreary, J.G., "A Simple Method for Controlling Gas Percolation in Numerical Simulation of Solution Gas Drive Reservoirs", SPEJ(March, 1985).
9. Jhaveri, B.S., and Youngren, G.K., "Three-Parameter Modification of the Peng-Robinson Equation of State to Improve Volumetric Predictions", SPE 13118, presented
   at the SPE 59th Annual Technical Conference, Houston, TX, Sept. 16-19, 1984.
10. Peneloux, A., Rauzy, E., and R. Freze, "A Consistent Correction for Redlich-Kwong-Soave Volumes", Fluid Phase Equilibria (1982) 8, 7-23
11. Whitson, C.H., and Brule, M.R., Phase Behavoir, SPE Monograph Volume 20, 2000
12. Improved Phase Behavior Predictions of Petroleum Reservoir Fluids from a Cubic Equation of State, Ingolf Soreide, Doctoral Dissertation, The Norwegian Institute of Technology, Department of Petroleum Technology, Trondheim, Norway, April, 1989
13. Lohrenz, J., Bray, B.G., and C.R. Clark, "Calculating Viscosity of Reservoir Fluids from their Compositions", J. Pet. Tech., (Oct. 1964) 1171-76, Trans. AIME 231
14. Ahmed, Tarek H., Hydrocarbon Phase Behavior, Gulf Publishing Co., 1989, Vol 7 in the Series Contributions to Petroleum Geology and Engineering, Geo. V. Chilingar, Series Editor.
15. The Properties of Gases and Liquids, Robert C. Reid, John M. Prausnitz, and Thomas K. Sherwood, 1977, McGraw-Hill Book Company

16. Peng, D.Y., and Robinson, D.B., "A Rigorous Method for Predicting the Critical Properties of Multicomponent Systems from an Equation of State", AIChE J. (1977) 23, 137-144.

17. Soave, G., "Equilibrium Constants for a Modified Redlich-Kwong Equation of State",
Chem. Eng. Sci. (1972) 27, 1197-1203.

18. Baker, L.E., "Three-Phase Relative Permeability Correlations", SPE/DOE preprint 17369, presented at the SPE/DOE Enhanced Oil Recovery Symposium, Tulsa, OK, April 17-20, 1988

19. Stone, H.L., "Probability Model for Estimating Three-Phase Relative Permeability", J. Pet. Tech., Feb. 1970, 214-218, Trans. AIME 249.

20. Fayers, F.J., "Extension of Stone's Method 1 and Conditions for Real Characteristics in Three-Phase Flow", SPERE, Nov. 1989 (437-445).

21. Stone, H.L., "Estimation of Three-Phase Relative Permeability and Residual Oil Data", J. Canadian Pet. Tech., Oct-Dec. 1973, 53-61.

22. Land, C.S., "Comparison of Calculated with Experimental Imbibition Relative Permeability", Trans. AIME (1971) Vol. 251, 419-425.

23. Carlson, F.M., "Simulation of Relative Permeability Hysteresis to the Nonwetting Phase", Paper SPE 10157, presented at the SPE 56th Annual Fall Technical Conference, San Antonio, TX, Oct. 5-7, 1981.

24. Sulak, R.M., Thomas, L.K., and Baode, R.R., "3D Reservoir Simulation of Ekofisk Compaction Drive", JPT (Oct. 1991) 1272-1278

25. Peaceman, D.W., "Interpretation of Well-Block Pressures in Numerical Reservoir Simulation with Nonsquare Grid Blocks and Anisotropic Permeability", SPEJ (June 1983) 531-43

26. Shiralkar, G.S., "Calculation of Flowing Well Pressures in Reservoir Simulation Using Nine-Point Differencing", J. Cdn. Pet. Tech. (Nov.1989) 28, No. 6, 73-82

27. Shiralkar, G.S. and R.E. Stephenson, "A General Formulation for Simulating Physical
Dispersion and a New Nine-Point Scheme", paper SPE 16975 presented at the 1987 Annual Technical Conference and Exhibition, Dallas, TX, Sept. 27-30

28. Shiralkar, G.S., "Reservoir Simulation of Generally Anisotropic Systems", SPERE (Aug. 1990) 409-414

29. Handbook of Natural Gas Engineering, Katz, D.L. et al, McGraw-Hill Book Co., Inc.,
NY (1959)

30. Peaceman, D.W., "Representation of a Horizontal Well in Numerical Reservoir Simulation", SPE 21217, Presented at the 11th SPE Symposium on Reservoir Simulation, Anaheim, CA, Feb. 17-20, 1991.

31. Coats, K.H., "Implicit Compositional Simulation of Single-Porosity and Dual-Porosity Reservoirs", SPE 18427, Presented at the 10th SPE Symposium on Reservoir
Simulation, Houston, TX, Feb. 6-8, 1989.

32. Thomas, L.K., and Coats, K.H., "Stone's k_{ro} Methods and Modifications", SPERE Forum submittal, SPE 25289, June, 1992.

33. Firoozabadi, A., and Katz, D.L., "An Analysis of High Velocity Gas Flow Through Porous Media", JPT (Feb. 1979) 211

34. Coats, K.H., and Ramesh, A.B., "Effects of Grid Type and Difference Scheme on Pattern Steamflood Simulation Results", JPT (May 1986) 557.

35. Muskat, M., Flow of Homogeneous Fluids Through Porous Media, J.W. Edwards, Inc., Ann Arbor, Michigan, 1946

36. Coats, K.H., "Engineering and Simulation", Fourth International Forum on Reservoir Simulation, Salzburg, Austria, Aug. 31 - Sept. 4, 1992

37. Odeh, A.S., "Comparison of Solutions to a Three-Dimensional Black-Oil Reservoir Simulation Problem", JPT (Jan. 1981) Vol. 33 p 13025

38. Weinstein, H.G., Chappalear, J.E., and Nolen, J.S., "Second Comparative Solution Project: A Three-Phase Coning Study", JPT (Mar. 1986) Vol. 38 345-353

39. Kenyon, D.E., and Behie, G.A., "Third SPE Comparative Solution Project: Gas Cycling of Retrograde Condensate Reservoirs", JPT (Aug. 1987) Vol. 39 981-998

40. Killough, J., and Kossack, C., "Fifth SPE Comparative Solution Project: Evaluation of Miscible Flood Simulators", SPE 16000, presented at the 9th SPE Symposium on Reservoir Simulation, San Antonio, TX, Feb. 1-4, 1987.

41. Nghiem, Long, Collins, D.A., and Sharma, R., "Seventh SPE Comparative Solution Project: Modelling of Horizontal Wells in Reservoir Simulation", SPE 21221, presented at the 11th SPE Symposium on Reservoir Simulation, Anaheim, CA, Feb. 17-20, 1991.

42. Killough, J.H., "Ninth SPE Comparative Solution Project: An Expanded Three-Dimensional Problem with a Geostatistical Distribution of Permeability", SPE 29110, presented at the 13th SPE Symposium on Reservoir Simulation, San Antonio, TX, Feb. 12-15, 1995.

43. Christie, M.A., and Blunt, M.J., "Tenth SPE Comparative Solution Project: A Comparison of Upscaling Techniques", SPE 66599, presented at the SPE Reservoir Simulation Symposium, Houston, TX, Feb. 11-14, 2001.

44. Whitson, C.H. and Torp, S.B.,"Evaulating Constant Volume Depletion Data", JPT, (March 1983) 610; Trans., AIME, 275.

45. Fetkovich, M.J., “A Simplified Approach to Water Influx Calculations – Finite Aquifer Systems”, JPT, (July 1971) 814-828.

46. Seidle, J.P., and Arri, L.E., “Use of Conventional Reservoir Models for Coalbed Methane Simulation”, paper CIM/SPE 90118, presented at the Canadian Institute of

Mining (CIM)/Society of Petroleum Engineers (SPE) International Technical Meeting, Calgary, Alberta, Canada, Oct. 10-13, 1990.

47. Paul, G.W., Sawyer, W.K., and Dean, R.H., "Validation of 3D Coalbed Simulators", SPE 20733 presented at the 65th Annual Technical Conference and Exhibition of the Society of Petroleum Engineers, New Orleans, LA, September 23-26, 1990.
48. Personal communication, Chet Ozgen, Nitec LLC, July, 2006.
49. Personal communication, Kent Thomas, Consultant, July 2006.

9 Appendices

Appendix 1. Linear Solver Parameters

In general you should not need to change any linear solver parameters, except possibly to enter NF in Initial Data if the default ILU solver is slower. The default solver is red-black ILU(0) (RBILU(0)). Enter "NF" in Initial Data to use Nested Factorization. Enter "D4" in Initial Data to use reduced band width direct solution for small problems. Do not enter any solver data for 1D problems. If you enter "D4", do not enter any other solver data.

The eligible solver keyword entries are:

```
ILU n n1 n2
NF
D4
TOLERANCE  tol
ITERATIONS  maxitn
NORTH  north
ORDER  order
```

n = the ILU order, ILU(n)
n1 = 1 to use residual constraint
= 0 to use no residual constraint
n2 = 1 to use red-black ILU
= 0 to not use red-black

tol = closure tolerance for solver iterations, psi
maxitn = maximum number of solver iterations
north = maximum number of orthogonal vectors in orthomin
order = ordering scheme for grid blocks, one of

`XYZ,XZY,YXZ,YZX,ZXY,ZYX`

Default Values

	ILU	NF
n	0	
n1	1*	
n2	1	
tol	.01	.1**
maxitn	60	30
north	30	30

* If n1 is 1, then residual constraint is deactivated if the solver fails to converge in maxitn iterations more than once. If n1 is entered -1, then residual constraint is permanently used.

** If the number of components, n_c, exceeds 6, default tol is .01

The default order is internally determined by the program at time of execution and is printed in the outfile.

The value north must be less than or equal to maxitn. If maxitn > north and the solver iterations exceed north, then:

restarted orthomin is used if north is positive
truncated orthomin is used if north is negative

NF convergence can depend strongly upon the ordering used to number the grid blocks. There are six orderings: xyz xzy yxz yzx zxy zyx. The program follows rules of Cheshire et al to determine the optimal ordering using N_x,N_y,N_z and average values of transmissibilities in the three directions, T_{xavg},T_{yavg},T_{zavg}. If any two or more of these average values are roughly equal, an ordering different from the default program-determined ordering may be better. The general principle is to order first in the direction of largest transmissibility, second in the direction of next-largest transmissibility. Ordering first in the direction of largest number of blocks is an additional consideration. You can override or change the default ordering determined by the program by entering

```
ORDER order
```

This entry assigns the ordering "order" to all Reservoirs - i.e. to the entire grid. If in a multiple reservoir case you wish to assign different orderings for different reservoirs then use the entry

```
ORDER order ires
```

This entry assigns ordering "order" only to Reservoir ires, where ires is an integer between 1 and nres, inclusive.

The number of solver iterations is printed in each Timestep Table. The program prints the three average transmissibilities and the ordering used at the beginning of the printout.

The program prints the maximum number of iterations for either NF or ILUat the beginning of the printout. The variable "north" is the maximum number of iterations for which Orthomin constructs orthogonal vectors. As north increases, storage and possibly cpu may increase significantly for both NF and ILU. Therefore, if you enter a larger north and find no benefit, then remove the entry in subsequent runs. Usually, the best remedy when NF performs poorly is use of the default ILU.

The closure tolerance (tol) for the linear solver is printed at the beginning of the printout. Additional closure tolerances for the linear solver are used and some are tied to the value of tol. In general:

1) the value of tol should not be changed.
2) if tol is reduced, cpu time usually will increase with no benefits - results will be unchanged.
3) in rare cases, a lower tol value is required.

To see whether the solver is taking the maximum number of iterations, edit your outfile and search for "(STEP" to locate the Timestep Table printout. That printout shows the global (all steps, all Newton iterations) maximum number of iterations taken by the linear solver and the number of Newton iterations in which the solver took the maximum number of solver iterations.

In the majority of cases, you should find that no solver parameter changes are necessary. In some cases, a change to NF will reduce cpu.

The default RBILU(0), with residual constraint (equivalent to the entry "ILU 0 1 1"), is the best of the ILU choices for most problems. Some experts have concluded that residual constraint is undesirable. However, residual constraint helps convergence in some problems. Datafile test14.dat (with the multiple Reservoir entry omitted) runs in 128 seconds cpu with the default residual-constraint RBILU(0). The cpu time increases to 336 seconds if residual constraint is off (entry "ILU 0 0 1").

In 2D problems, the optimal ILU order frequently is higher than in 3D problems, e.g. RBILU(1) or ILU(2).

The ILU options are listed here in order of increasing robustness:

```
          entry
ILU(0)       ILU  0 1 0     ! least robust
ILU(1)       ILU  1 1 0
RBILU(0)     ILU  0 1 1      ! default
ILU(2)       ILU  2 1 0
RBILU(1)     ILU  1 1 1
ILU(3)       ILU  3 1 0
  ... etc. ...
```

With increasing robustness:

(1) The storage requirement and cpu time per solver iteration increase.
(2) The number of solver iterations required usually decreases.

Thus for a given problem there will be an optimal ILU option giving the least total solver cpu time. This optimal option is frequently the default RBILU(0).

The major effect of solver failure is numerous time step cuts. The cumulative number of cuts is printed in the Timestep Table and in the end of run Timestep Summary. The cumulative number of solver failures (number of Newton iterations where the solver failed) is printed in the Timestep Table. A few time step cuts are ok - they do not affect results. Numerous step cuts are a cause for concern. Similarly, a few solver failures are ok while numerous solver failures are a cause for concern. When many time step cuts occur, check the solver failures. If the latter are numerous, then change to another solver option. In one Impes field study datafile, the NF solver failed badly and caused many timestep cuts. The default ILU solver gave no solver failures and no timestep cuts. For that same problem run in Implicit mode, the NF solver worked well and was faster than ILU.

Incompressible problems and 2D problems, such as SPE10, may require tighter solver tolerance and/or higher ILU. Example for spe10_case1.dat:

	n RBILU(n)	tol	steps	itns	cuts	solver itns/newt	cpu sec	final gor
default	0	.01	***bomb***					
	0	.001	1024	1148	1	22.55	504	267
	0	.0001	1022	1098	0	28.94	620	259

1	.0001	1026	1092	1	23.3	538	257
2	.0001	1023	1097	1	23.1	557	258
3	.0001	1028	1086	0	17.1	528	257
3*	.0001	1022	1070	0	20.26	570	258
4	.0001	1035	1107	2	17.5	576	258
5	.0001	1041	1119	2	14.2	622	258
5*	.0001	1026	1084	0	17.1	652	258

ILU n1 n2 n3

3 1 0	.0001	1026	1093	1	28.7	1186	256
6 1 0	.0001	1031	1089	0	32.2	1542	259
9 1 0	.0001	1025	1096	1	18.9	1097	258
15 1 0	.0001	1024	1095	1	17.3	1552	258

* resid constraint off, i.e. ILU n 0 1

Appendix 2. Example Ninepoint Results

Enter

```
NINEPOINT
```

in Initial Data to use the nine-point difference scheme in the xy plane. No other entries or changes are necessary. If you enter NINEPOINT and need to zero certain pore volumes for symmetry, use MODIFY PV rather than a MOD on POROS or PV. The nine-point difference scheme reduces grid orientation effects in adverse-mobility ratio pattern floods. However, it is not needed in full field studies where significant heterogeneity and gravity effects dominate grid orientation effects.

The nine-point difference scheme is a modification of the Shiralkar method[26-28]. The modifications provide accuracy or consistency in the following sense:

1) For unidirectional flow, where permeability varies in the flow direction, the method gives the correct ("series resistance") relationship between flow rate and pressure drop.

2) For unidirectional flow in a striped system, the method gives strip flow rates which are proportional to their permeability*width*thickness product values.

A 5-spot (or 9-spot) adverse-mobility ratio pattern flood is a test of the nine-point logic. There is no need to use more than one layer and it doesn't matter whether a gas/oil or water/oil problem is used. Datafile test13.dat is a one-layer grid for 1/4 or 1/8 of a 5-spot waterflood of an undersaturated oil. The oil/water viscosity ratio is 30 (unfavorable) and k_{rw}, k_{ro} are equal to S_n^2 where S_n is normalized saturation. There are four Impes, black oil datafiles in test13.dat:

test13.dat diagonal grid 5-point 11x11x1 1/8 5-spot
test13a.dat parallel grid 5-point 15x8x1 1/8 5-spot
test13b.dat 11x11x1 kx=kx(x) ky=ky(y) unidirectional x-direction flow . There should be no y-direction flow with 9-point.
test13c.dat Unidirectional flow rate in striped system with arbitrary k variation in transverse direction. Flow in strips should be proportional to their kh*width products with 9-point.

The test13.dat datafiles illustrate use of the keywords EDGE, ELEMENT, NINEPOINT, and KRANALYTICAL. Each file can be run with 9-point by inserting the keyword NINEPOINT. The 15x8 parallel grid is equivalent to the 11x11 diagonal grid. Grid block sizes are nearly identical in the two cases, with 66 active blocks (diagonal) vs 64 active blocks (parallel) on a 1/8 5-spot basis. ELEMENT can be used only for diagonal grids.

The 9-point scheme reduces the spread between parallel-grid and diagonal-grid watercut curves.

Appendix 3 Foamy Oil Black Oil Option

The logic of the foamy oil option is as follows. As gas appears due to gas injection or pressure decline, it first exists as gas entrained in the oil. It later appears as free gas as the gas saturation exceeds a specified maximum value, sgemax. The entrained gas flows with the oil in a miscible fashion, as an emulsion. This foamy oil treatment bears some resemblance to conventional black oil treatment with a high critical gas saturation, S_{gc}. However in that case the first S_{gc} of gas saturation is trapped and does not flow. The phase behavior treats the oil and entrained gas as immiscible phases in the normal fashion.

The foamy oil treatment has two effects on recovery which are absent in conventional black oil treatment. First, the retention of gas in entrained form lowers gor and retains reservoir energy for production, thus increasing recovery. Second, the effect of entrained gas on the oleic phase viscosity may increase or reduce recovery, depending upon whether the emulsion viscosity decreases or increases as entrained gas saturation increases.

The foamy oil option is coded for black oil for both Impes and Implicit. Fault (non-neighbor) connections and nine-point are permissible but the multiple reservoir, multiple pvt options are not permissible.

To activate the foamy oil option, enter in Initial Data

```
FOAM sgemax vismult
```

sgemax = maximum value of entrained gas saturation, fraction

vismult = coefficient for emulsion viscosity

The entrained gas is limited by oil saturation as well as by sgemax as follows:

$$sge = \text{Min}(sgemax, S_o, S_g)$$

This limits the emulsion to 50% entrained gas by volume; that is, the maximum value of $f_g = sge/(sge+S_o)$ is .5, where f_g is the volume fraction of entrained gas in the emulsion. The viscosity of the oleic or oil-gas emulsion phase may increase or decrease with an increase in entrained gas saturation:

$$\mu_{oleic} = \mu_o (1 + vismult * f_g)$$

where μ_o is the pressure-dependent oil viscosity from the black oil pvt table. The entered value of vismult must be greater than -2 since f_g may increase to .5, regardless of the sgemax value.

In the Implicit foamy oil case, the model internally sets S_{gc} to .001 in all grid blocks where the data S_{gc} is less than .001. This low value has a negligible effect on results but helps Implicit stability.

For foamy oil problems we recommend you use the Implicit option – enter the keyword IMPLICIT in Initial Data. If the default Impes is used, the CFL entry in Recurrent Data is required and run cpu times may be 10 to 100 times larger than Implicit runs.

Datafiles test11.dat and test11a.dat are xz cross-section and 3D test problems for the foamy oil case. Both are Implicit. If the test11.dat problem is run with Impes, the cpu time is 100 times greater than Implicit, due to the small stable step of Impes.

Appendix 4. Three-Phase Oil Relative Permeability

Stone's Methods 1 and 2

In default mode, Sensor calculates three-phase k_{ro} Stone's Method 1[19] with variable Som[20]. Stone's Method 1 equation can be written as [32].

$$k_{ro} = R * k_{rocw} * K_{row} * K_{rog}$$

k_{rocw} = rel perm to oil at $S_w=S_{wc}$, $S_g=0$

k_{row} = rel perm to oil at $S_g=0$, a function of S_w

k_{rog} = rel perm to oil at $S_w=S_{wc}$, a function of S_g

K_{row} = k_{row}/k_{rocw}

K_{rog} = k_{rog}/k_{rocw}

S_w^* = $(S_w - S_{wc})/(1 -S_{wc} -S_{om})$

S_g^* = $S_g/(1 -S_{wc} -S_{om})$

R = $(1 - S_w^* - S_g^*)/((1 - S_w^*)*(1 - S_g^*))$

S_{om} = minimum oil saturation, below which k_{ro} is 0

If S_{om} is a constant, it must obey $S_{om} < \text{Min}(S_{orw},S_{org})$[32]. Here we use Fayer's method which treats S_{om} as variable, varying with S_g with positive ($a>1$) or negative ($a<1$) curvature. This is discussed by Fayers[20] and Coats[36].

$$S_{om} = S_{org} * S_{gn}^{a} + S_{orw} * (1-S_{gn})^{a}$$

$$S_{gn} = S_g/(1-S_{wc}-S_{org})$$

Sensor uses a default value of a=1.0. If you want to use Stone's second method, then enter STONE 2 in Initial Data.

Numerous modifications of Stone's second method appear in the literature, most addressing a supposed need to alter Stone's method for the case where k_{rocw} is not equal to 1. No such need exists. Stone gives his second method as

$$k_{ro} = (k_{rw} + k_{row})(k_{rg} + k_{rog}) - k_{rw} - k_{rg}$$

He states that $k_{ro}=k_{row}$ when $S_g=0$ and $k_{ro}=k_{rog}$ when $S_w=S_{wc}$ which makes clear that he writes his equation with a basis of k_{ocw} for all relative permeabilities. Thus his equation is

$$k_o /k_{ocw} = (k_w/k_{ocw} + k_{ow}/k_{ocw}) (k_g/ k_{ocw} + k_{og} /k_{ocw}) - k_w /k_{ocw} - k_g /k_{ocw}$$

where k_{ocw} is effective permeability to oil at $S_g=0$ and $S_w = S_{wc}$ and $k_o,k_w,k_g,k_{ow},k_{og}$ are effective permeabilities, independent of the arbitrary basis selected to define relative permeability. Selecting an arbitrary basis k* for relative permeability, and multiplying both sides of the last equation by k_{ocw}/k^* ($=k_{rocw}$) gives

$$k_{ro} = (1/k_{rocw}) (k_{rw} + k_{row})(k_{rg} + k_{rog}) - k_{rw} - k_{rg}$$

which is Stone's Method 2 equation (unmodified) for any arbitrary basis k*.

Baker's Saturation-Weighted Linear Interpolation (SWLI)

As discussed above in Section 1.4, enter in Initial data

```
KROINT
  n    1
```

to use the SWLI method for three-phase k_{ro} in Saturation Table n. Enter the two-column table

```
SGWT n
  Sw  krwg   ! any number of lines
```

to use the SWLI method for k_{rw} in Saturation Table n.

Baker[18] presents his linear interpolation method as one which is more appropriate for mixed wettability rock, since both k_{rw} and k_{ro} are functions of both S_w and S_g saturations.

$$k_{ro} = ((S_w-S_{wc})*k_{row} + S_g*k_{rog}) / (S_w-S_{wc}+S_g) \quad (1)$$

$$k_{rw} = (S_o*k_{rwo} + S_g*k_{rwg}) / (S_o+S_g) \quad (2)$$

The k_{rwo} and k_{row} are water and oil relative permeabilities, respectively, measured in a water-oil system with S_g=0. They are the k_{rw} and k_{row} values entered under SWT. The k_{rog} is oil relative permeability measured in a gas-oil system with S_w=S_{wc}, entered under SGT or SLT. The k_{rwg} is water relative permeability measured in a gas-water system with S_o=0, entered under SGWT.

Appendix 5. Discussion of Trapped Gas, k_{rg} Hysteresis, and Land's Method

The effect of trapped gas saturation with k_{rg} hysteresis is significant in some simulations, insignificant in others. When the effect is significant, different ways of treating the hysteresis can give different results even for the same value of maximum trapped gas saturation.

The following defined grid block saturations are data and do not change with time:

S_{gr} = maximum trapped gas saturation, the value entered as data.

S_{gmax} = maximum gas saturation for which S_{gr} is defined, either1-S_{wc} or 1-S_{wc}-S_{org}.

S_{gc} = critical gas saturation

S_{org} = residual oil saturation to gas

S_{wc} = connate water saturation.

The following grid block saturations change with time:

S_g = grid block gas saturation.

s_{gmax} = historical maximum value of grid block gas saturation.

s_{gr} = current residual or trapped gas saturation, a function of s_{gmax}, $s_{gr} = S_{gr}$ when $s_{gmax} = S_{gmax}$.

If a grid block is initially at $S_g = S_{gmax}$ and then flooded by water and/or oil, then the gas phase will cease to flow at a saturation $S_g=S_{gr}$. An obvious question is "What would the trapped gas saturation s_{gr} be if initial $S_g = s_{gmax} < S_{gmax}$?". The simplest (linear) relation is

$$s_{gr} = S_{gc} + C\,(s_{gmax} - S_{gc}) \qquad (1)$$

Land proposed the equation

$$s_{gr} = s_{gmax}/(1 + C\,s_{gmax}) \qquad (2)$$

The constraint that s_{gr} must equal or exceed S_{gc} applies to each of Eqns (1) and (2). Whenever the calculated s_{gr} is less than or equal to S_{gc}, there is no hysteresis and k_{rg} is simply calculated from the entered k_{rg} vs S_g data.

Hysteresis occurs when $s_{gr} > S_{gc}$ and $S_g < s_{gmax}$. Note that s_{gr} and s_{gmax} are set at the end of each time step so that S_g can exceed s_{gmax} in the Newton iterations of a given step. If $S_g > s_{gmax}$ there is no hysteresis - k_{rg} is simply calculated from the entered k_{rg} vs S_g data.

Since $s_{gr} = S_{gr}$ for $s_{gmax} = S_{gmax}$, Eqns (1) and (2) give respective values of C as

$$C = (S_{gr}-S_{gc})/(S_{gmax}-S_{gc}) \qquad (1a)$$

$$C = Cland = (S_{gmax}-S_{gr})/(S_{gmax}S_{gr}) \qquad (2a)$$

These equations relate the constant C to the maximum trapped gas saturation S_{gr}. Thus we can specify C or S_{gr}, but not both. We have not defined S_{gmax} here except for the requirement that $s_{gr} = S_{gr}$ when $s_{gmax} = S_{gmax}$. In black oil simulation, oil saturation in a grid block is generally at least S_{org}. Thus in frequent use is the definition

$$S_{gmax} = 1 - S_{wc} - S_{org} \quad (3)$$

However, the definition

$$S_{gmax} = 1 - S_{wc} \quad (4)$$

is arguably better in compositional simulation since oil can be and frequently is vaporized to zero or very low oil saturation. Even in black oil simulation, grid blocks having $S_o = 0$ occur in the initial gas cap region, and in black-oil gas-water problems, S_o is of course 0.

If S_g initially increases, with corresponding increases in s_{gmax} and s_{gr}, then decreases toward 0, S_g may decrease to values much less than s_{gr}. Subsequent increase of S_g would then result in gas phase immobility until S_g increased to the "large" s_{gr} value. This is a physically untenable situation. For example, if S_g decreased to 0, the whole process should start over with $s_{gr} = S_{gc}$. Thus the condition

$$s_{gr} < S_g + \Delta S \quad (5)$$

is imposed, with s_{gr} reset, if necessary, to satisfy it as S_g decreases. (If s_{gr} is reset, s_{gmax} is also reset in accordance with the relationship between s_{gr} and s_{gmax}, Eqn (1) or (2).)

In default mode, Sensor uses:

s_{gr} from linear option Eqn (1)

$S_{gmax} = 1 - S_{wc}$

$\Delta S = 0$

You can use the following Initial Data entry to change those defaults:

LAND n ΔS kw

n = 1 s_{gr} from linear option Eqn (1)

= 2 s_{gr} from Land option Eqn (2)

ΔS = value desired in Eqn (5)

kw = SWC for $S_{gmax} = 1-S_{wc}$

= SORG for $S_{gmax} = 1-S_{wc}-S_{org}$

Sensor uses geometric similarity to obtain the imbibition k_{rg} curve used when $s_{gr} > S_{gc}$ and $s_{gr} < S_g < s_{gmax}$. The imbibition k_{rg} curve from s_{gr} to s_{gmax} has the same shape as the drainage k_{rg} curve has between S_{gc} and s_{gmax}. This imbibition curve is similar to the curve derived from the drainage curve described by Land[22] and Carlson[23], but requires no additional input or decisions from the user.

The Land option 2 (Eqn (2)) can cause oscillations and instability, reflected in more time steps, iterations, and higher cpu time. Sensor contains logic to minimize that. For the same S_{gmax} and S_{gr} values, Eqn (1) gives lower s_{gr} than does Eqn (2). Increasing s_{gr} tends to lower average gor and increase average pressure.

Appendix 6. Normalized Relative Permeability

The normalization method for the tabular relative permeability functions described here is one proposed by Phillips (L. K. Thomas, 8/93). One or more Saturation Tables are entered as data and each grid block is assigned to one of them. Each Table contains two tables, a water-oil table giving k_{rw}, k_{row}, P_{cwo}, P_{cwoi} vs S_w, and a gas-oil table giving k_{rg}, k_{rog}, P_{cgo}, vs S_{liq} (= 1-S_g) or vs S_g. Normalization is done (a one-time calculation at the beginning of the run) to represent each of the quantities as a function of a normalized saturation. The five normalized saturations and variables which are functions of them are:

S_n	quantities
$S_{w1} = (S_w - S_{wc})/(1 - S_{wc})$	k_{rw}, P_{cwo}
$S_{w2} = (S_w - S_{wcg})/(1 - S_{wcg})$	k_{rwg}
$S_{w3} = (S_w - S_{wc})/(1 - S_{wc} - S_{orw})$	k_{row}, P_{cwoi}
$S_{g1} = (S_g - S_{gc})/(1 - S_{wc} - S_{gc})$	k_{rg}
$S_{g2} = S_g/(1 - S_{wc} - S_{org})$	k_{rog}
$S_{g3} = S_g/(1 - S_{wc})$	P_{cgo}

In effect, the normalization calculation stores k_{rw} as a tabular function of S_{w1}, k_{row} as a function of S_{w3}, k_{rog} as a function of S_{g2}, etc.

Endpoint saturations are stored in full-grid arrays. They are S_{wc}, S_{orw}, S_{org}, S_{gc}, S_{gr}, and, if SWLI is used, S_{wcg}. The model then computes k_{rw}, etc., as follows, using k_{rw} with no SWLI as an example. The current grid block S_w saturation and grid block S_{wc} value are used to calculate S_{w1}. k_{rw} is then calculated by table lookup in the normalized k_{rw} vs S_{w1} table formed at the beginning of the run. Gridblock relative permeabilities are additionally functions of any entered relative permeability endpoint arrays (KRWRO, KROCW, KRGRO)

Any grid block endpoint saturation not entered as an array is defaulted to its value in the Saturation Table assigned to the grid block.

Appendix 7. Turbulence and Well Index Equations (Reference 36)

Consider a grid block of dimensions $\Delta x,\Delta y$ and net thickness h penetrated by a vertical wellbore of radius r_w at its (xy) center. We want a relationship between gridblock pressure p, flowing wellbore pressure p_w, and gas phase production rate q, reflecting non-Darcy (turbulent) flow effects.

Definitions:

p	average gridblock $p = 2\pi\, I(r\, p(r)dr) / \pi(r_e^2-r_w^2)$
q	free gas production rate, mols/day
ρ	average gas phase density $r_w < r < r_e$, mols/rb
λ	gas phase mobility = k_{rg} /μ_g, 1/cp
M	gas molecular weight
r_w	wellbore radius, ft
r_d	radius of damaged zone
k	permeability in $r_d < r < r_e$, md
h	net thickness, ft
b	Katz Beta factor in $r_d < r < r_e$, 1/ft
k1	permeability in damaged zone $r_w < r < r_d$
b1	Katz Beta factor in damaged zone, 1/ft
$\Delta x,\Delta y,h$	gridblock dimensions (h is net thickness)
s	skin factor
J	layer productivity index, rb-cp/d-psi
r_e	effective exterior radius of gridblock, ft
p_e	pressure at $r = r_e$
b_{eff}	effective Katz Beta factor in region $r_w < r < r_e$

Assumptions are: a) the flow in the grid block is approximated as steady-state radial flow of free gas, b) gas properties ρ,λ,M are given average (constant) values in $r_w < r < r_e$ for integration purposes.

Forscheimer's equation, ignoring unit conversion constants, is

$$dp/dr = (1/k\lambda)v + b\,\rho v^2 \qquad (1)$$

where v is superficial gas phase velocity in toward the wellbore. Integrating this equation from r_w to r_e twice (first to get p(r), second to get average pressure p) gives

$$p-p_w = q/(\lambda J\rho) + (41.1E\text{-}16)b_{eff}M/(h^2 r_w\rho)\, q^2 \qquad (2)$$

where

$J = .00708kh/(\ln(r_e/r_w)-1/2+s)$ rb-cp/d-psi

$s = [k(r_d-r_w)/(k1 r_w)]$

$b_{eff} = b[1 + b1(r_d-r_w)/(br_w)]$ 1/ft

p_w = flowing wellbore pressure, psia

and a $1/g_c$ is included in the constant 41.1E-16, where g_c is the acceleration of gravity, 32.2 (ft-lb mass)/(lb force - sec^2).

The well index J is calculated using Peaceman's formulae[25]. The term $\ln(r_e/r_w)$-1/2 is replaced by $\ln(r_0/r_w)$ where r_0 is Peaceman's effective well radius. We generalize to the case of a wellbore penetrating the center of a grid block in any direction - x or y or, as above, z directions. We label the wellbore direction direction 3 and the directions in the plane perpendicular to the wellbore directions 1 and 2. Block permeabilities and dimensions in directions 1,2,3 are k_1,k_2,k_3 and h_1,h_2,h_3, respectively. Peaceman's formula thus generalized, for the anisotropic case, is[25]

$$J = .00708kh_3/(\ln(r_0/r_w)+s) \qquad (3)$$

where $k = (k_1k_2)^{.5}$, k and h are in units of md and ft, and

$$r_0 = .28[\ (k_2/k1)^{.5}\ h_1^2\ + (k_1/k_2)^{.5}h_2^2\ ]^{.5} / [\ (k_2/k_1)^{.25} + (k_1/k_2)^{.25}\ ] \qquad (4)$$

For the isotropic case ($k_1 = k_2$), this reduces to

$$r_0 = .14[h_1^2 + h_2^2]^{.5}$$

Eq (2) can be written in a form useful for interpreting isochronal well tests:

$$q = .0070838\rho\lambda kh(p-p_w)/(\ \ln(r_0/r_w) + s + Dq\ ) \qquad (5)$$

where

$$D = (2.911E-17)b_{eff}M\lambda k/(hr_w) \qquad (6)$$

The value of D can be determined from isochronal well test data for entry into the Sensor datafile. Sensor accepts these D values entered as layer beta values. To signify that the entered value is D and not the multiple of Katz et al Beta factor, the D value must be entered negative.

Sensor extracts and stores the invariant part of D (M and λ vary with time) by using the values (at the time D is entered) of M and λ internally available. If λ is 0, a warning message is printed and turbulence for that layer is zeroed.

If beta is entered > 0, then Sensor calculates the Katz turbulence factor Beta from the Firoozabadi-Katz correlation[33]

$$\text{Beta (1/ft)} = \text{beta} * 2.6E10/(k)^{1.2} \qquad (7)$$

where k is grid block permeability calculated as $\text{sqrt}(k_1k_2)$.

Here we discuss the entry of beta as a negative number, signifying that |beta| is the D of Manual Eqs. 5 and 6. We emphasize the pitfalls of misuse related to this usage of D (we recommend entry of positive beta layer values, which is not prone to these pitfalls).

Note:

Enter layer D negative in units of day/mcf (e.g. if D=.001 day/mols, enter layer beta as -.001/.379 = -.002638 day/mcf). Note that in Eqs 5 and 6, q is layer free gas flow rate in mols/day and D has units of day/mols - use (say) .379 mcf/mol conversion, assuming standard conditions of 60 deg F and 14.7 psia.

We consider a well completed in n layers (perfs) j=1,2,..,n. For a given layer j during steady-state flow at well rate Q,

Sj = sj + Djqj

Sj = total skin of layer j

sj = mechanical skin of layer j

Dj = D of layer j, day/mols

qj = layer j free gas flow rate, mols/d at reservoir conditions

Q = total well gas production rate, mcf/day of surface gas

Note that if rs > 0 and/or there exists two-phase gas-oil flow, Q is NOT equal to sum(j)qj, where sum(j) denotes summation over j=1 to n and common units of mols/d or mcf/d are used. Rather than belaboring this point, we proceed, assuming rs=0 and single-phase gas flow. In this case Q = sum(j)qj, provided common units of mols/d or mcf/d are used.

We use the four-layer datafiles in turb.dat to illustrate the difficulty or misuse of D. Isochronal well tests are performed at rates Q1,Q2, etc. These tests give total well skins S1, S2, etc. Any pair, say tests 1 and 2, gives two equations in the two unknowns s and D,

S1 = s + DQ1

S2 = s + DQ2

where s and D are TOTAL WELL mechanical skin and D values. The datafile turb.dat uses Sensor (with beta=10 and s=2 for all four layers) to generate these well tests. The Q1=10000 and Q2=20000 mcf/d test rates give S1=5.9 and S2=13.7, which in turn give from the two equations s=-1.9 and D=.00078 day/mcf.

Using these WELL s,D values as LAYER values, Sensor gives extremely erroneous pw (flowing bottomhole presures) for the Q1=10000 and Q2=20000 mcf/d (and other rate) well tests. The problem here is that the well tests give WELL s,D values but give no insight into the needed LAYER sj,Dj values.

An engineer described to us the following method for deducing layer sj,Dj values from welltest well s,D values. He said other simulation models use this method or a variant thereof. We are unaware of a reference for it. While it works rather well for our example, we question its generality.

Method

enter layer sj for all layers equal to well s.

enter layer Dj as well D * kh(well)/kh(layer)

The resulting sj and Dj values are entered in turba.dat (which is included in the file turb.dat). The resulting pw vs Q agree well with the welltest pw vs Q values. The layer gas flow rates differ from the correct (welltest) values by 30-40+ % in the lower-rate layers.

Kent Thomas of ConocoPhillips (January, 2005) derived the above method using the assumptions:

1. layer mechanical skin sj in all layers = s

2. the quadratic term in the Forscheimer equation dominates the Darcy term p-pw is constant for all layers qj is directly proportional to (kh)j each of gas phase mobility and molar density is the same in all layers.
3. These assumptions are in addition to those of radial, steady-state, single-phase gas flow.

Appendix 8. Discussion of Perforation Rates and Cumulatives

In the compositional case, production well perf rates and cumulatives are non-physical quantities. An individual perf rate is calculated by flashing that layer's inflow stream through the surface multistage separation. Obviously that is non-physical. The actual well rate is obtained by flashing the total wellstream (sum of all perfs' inflow streams) through the surface multistage separation. The oil (gas) obtained by a flash of the sum of the individual perf streams is not equal to the sum of the oil (gas) amounts obtained by separately flashing the individual perf streams. In general, the two are close but become more different as the compositions of the individual perf streams become more disparate. In the case of a gas condensate reservoir, little to none of the well's oil rate may come from oil phase flow in the reservoir. Thus saturations do not affect gor - only composition affects gor. Therefore, gor difference among the perfs is a measure of the composition differences in the individual perf inflow streams.

The Well Table printout in Table 11 illustrates the above. The perf oil rates sum to 130 stb/d but the well oil rate is 118 stb/d. The perf gas rates sum to 1318 mcf/d but the well gas rate is 1348 mcf/d. These differences are relatively large but so is the difference in compositions of individual perf streams, as indicated by gor ranging from 2322 to 36654 scf/stb.

Perf rates and cumulatives have only a qualitative meaning. They closely reflect the relative contributions of different layers to total well rate. But they are not quantitative (or "real" or "physical").

Also, the model at any time requires storage and cpu only for active perfs at each step. The well table prints out these active perfs. Due to past reperforation history, there may be other perfs (other perforated grid blocks) active previously but not at the present time. In that case the sum of the printed (currently active) perf cumulatives may be far different from the well cumulatives.

If perf cumulatives are of interest, the REGION feature of the model might be more relevant. For example, you might define N_z regions, each region being a layer of the reservoir. Or the regions might be a number of layer groups. The printed region summary then shows the rates and cumulatives for each region and does not suffer from the "missing perfs" problem. The region rates and cumulatives are still only qualitative or approximate due to the first item discussed above.

Appendix 9. Additional Discussion of Ontime

Let tp, tw, and tg denote production ontime, water injection ontime, and gas injection ontime, respectively, for a platform. Each is physically the fraction of time that the respective process is on or active. For discussion, consider a one-day time step. If tp is .4 then production is at full or target production rate for .4 days and is shut in for .6 days. If tg is .8 then gas injection operates at full or target rate for .8 days and is shut in for .2 days. Similarly for tw.

For clarity of discussion, consider a case where:

outside gas = sale = fuel = 0

q_g = platform target production rate = 100

q_{gi} = platform target injection rate = net produced gas rate

no wells are restricted by capacity.

Let entered ontimes for production and gas injection, tp and tg, both be < 1. Then production and injection are simultaneously on for an ontime t = min(tp,tg). We do have a choice here. For example, if tp = tg = .5, we could say that means that production is active the first half-day and injection is active the second half-day. However that wouldn't make much sense since there would be no produced gas to inject during the period that gas injection is on. We therefore define ontimes to be "coincident", i.e. on simultaneously for the ontime t=min(tp,tg).

Example:

tp = .8 tg = .8

production is active at q_g=100 for .8 days

injection is active at q_{gi}=100 for the same .8 days

Actual (average) rates for the one-day time step:

q_{gavg} = 80 q_{giavg} = 80

tp = .8 tg = .4

q_g = 100 for .8 days q_{gavg} = 80

q_{gi} = 100 for .4 days q_{giavg} = 40

tp = .4 tg = .8

q_g = 100 for .4 days q_{gavg} = 40

q_{qi} = 100 for the .4 days when production is on,

= 0 for the other .6 days

q_{giavg} = 40

For this case, if outside gas were available, gas injection would continue (using outside gas only) for the period from .4 to .8 days and q_{giavg} would be > 40.

Water injection is not tied to "available" water as gas injection is tied to net produced gas. Thus water injection rate is a function only of water injection target rate and tw, in the same sense that production rate is a function only of production target rate and tp.

Consider the case where you wish to inject a fraction of net produced gas rather than all of it. Let that fraction = .7. For the three examples above, the average rates are now:

tp = tg = .8 $q_{gavg} = 80$ $q_{giavg} = 56$

tp = .8 tg = .4 $q_{gavg} = 80$ $q_{giavg} = 28$

tp = .4 tg = .8 $q_{gavg} = 40$ $q_{giavg} = 28$

The average rates in real field cases can become considerably complicated by platform constraints, well constraints, wells going on pressure constraint (capacity), available gas dependencies upon fuel and sales and outside gas, and ontimes. The platform table (search for PLATFORM TABLE) is perhaps the best printout to view the various platform rates.

WELLONTIME Example:

Consider the following example where rates are entered at 30-day intervals through history. Between 1440 and 1470 days, well W4 comes onstream at 1441 days and well W12 goes offstream at 1460 days. For such wells, you can enter their physical rates - i.e. their rates during the period they were flowing. Then enter WELLONTIME reflecting the fraction of the 30 days when they were flowing. This example shows the avoidance of TIME entries at 1441 and 1460 days.

```
....
TIME 1440
RATE
  ....
  W4  13000
  W7  17000
  W12 11000
  ....
WELLONTIME
  W4   .9667
  W12  .6667

TIME  1470
RATE
```

If, as normally, all entered well rates are average values for the time period, do not enter wellontime. For example, the above example data could be

```
....
TIME 1440
RATE
```

```
    ....
    W4  12567.1
    W7  17000
    W12  7333.7
    ....

  TIME  1470
  RATE
....
```

leaving only a question of whether the larger W4 and W12 physical rates can be met without pressure constraint. The keyword WELLONTIME should be seldom or never used.

Appendix 10. Well Index J for a Homogeneous 5- or 9-spot Pattern Element

If the diagonal or parallel pattern grid has half blocks on the edges and constant, equal or unequal, Δx and Δy, the well indices J calculated internally in Sensor are correct for:

1) the isotropic or anisotropic case,

2) the 5-point or 9-point difference scheme

3) the 5-spot or 9-spot case

In those cases the procedure described below is not required. Anisotropy is restricted to the case where the grid x and y axes coincide with the principal axes.

For both isotropic and anisotropic cases, the Sensor internal calculation of J uses the Peaceman[25] and Shiralkur[26-28] equations for the 5-point and 9-point difference schemes, respectively. Those equations apply for isolated wells with constant grid spacings and half edge blocks.

The method described here can be used to determine well index J for any of the above three cases when variable grid spacing and/or full edge blocks are used. The use of full vs half edge blocks can reduce Impes cpu time by a factor of about two.

The method requires a model run using the desired xy grid in a dataset representing single-phase flow in one layer of constant kh of a 1/4 5-spot. The well index for any other kh is easily obtained by noting that J is proportional to kh. There are few restrictions on the desired xy grid. It may be diagonal or parallel and have half blocks or full blocks along the edges. If grid blocks have constant $\Delta x,\Delta y$, the Δx and Δy may be equal or unequal. If variable spacing is used in one direction, the spacing in the other direction may be the same variable spacing, a different variable spacing, or constant. The required model run can use any layer properties k,h, and ϕ, any rate, any single-phase fluid mobility $\lambda = k_r/\mu$, and any well index J values. Run the dataset to steady-state, a short time of (say) 50 days.

The full well index J is given by

$$J = \frac{2q/\lambda}{\Delta p_{wb} - \Delta p} \qquad (1)$$

and

$$\Delta p_{wb} = qa/kh\lambda \qquad (2)$$

where

q = injection rate for a full pattern

λ = fluid mobility, k_r/μ

Δp_{wb} = pressure difference between flowing bottomhole pressures

Δp = pressure difference between wellblock pressures

$a = (\ln(d/r_w) - .61739)/\pi$

d = distance between injector and producer

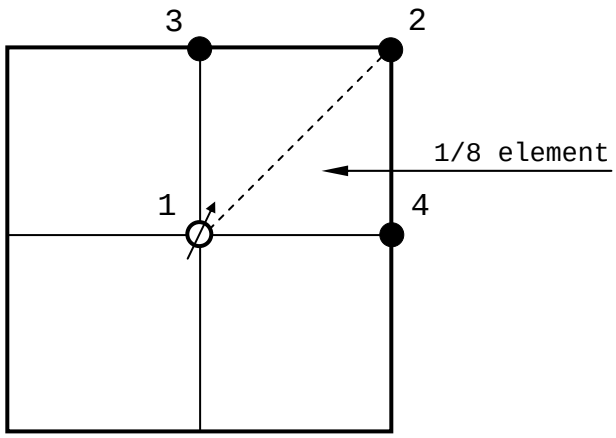
r_w = wellbore radius

The value of Δp is obtained from the outfile printout. Eqn (2) is Muskat's equation[35] for the isotropic 5-spot with the more accurate constant .61739[25] replacing the .6190 value given by Muskat. For English units of md, ft, cp, psi and rb/d, the units conversion constant on k is .0011274. Units of J are then rb-cp/d-psi. It is simplest to use $\lambda = 1.0$, as in the dataset listed in Example 1 below. J is not dependent upon λ.

Eqn (2) serves as a check on the value of J determined from the model run. That is, entering the determined J value and rerunning the dataset should give a Δp_{wb} value equal to that given by Eqn (2).

Eqn (2) applies only for the isotropic case. Eqn (1) applies only when all wells in the 1/4 5-spot or 9-spot have the same J value. The figure here shows the 1/4 5-spot injection and production wells 1 and 2 and 9-spot production wells 3 and 4.

5-spot or 9-spot Pattern



All wells have the same J if each of the $\Delta x(i)$ and $\Delta y(j)$ spacings is symmetric. The spacing $\Delta x(i)$ is symmetric if $\Delta x(i) = \Delta x(N_x+1-i)$ for $i = 1,2,..,N_x/2$ (round down if N_x is odd). And similarly for symmetric $\Delta y(j)$ for $j=1,2,..,N_y/2$. Clearly, any constant spacing is symmetric. More detail on unequal J values is given in Section II below.

All examples here are for a 40-acre 5-spot or 9-spot, and use $r_w = .25$ ft, $k_x = 800$ md, $h = 50$ ft, $\lambda = 1.0$, $q = 4000$ rb/d, and a 1/4 pattern grid. For the isotropic case, $a = (\ln(d/r_w) - .61739)/\pi = 2.4216$, and from Eqn (2), $\Delta p_{wb} = 4000*2.4216/800*50*.0011274 = 214.796$.

I. The Isotropic Case - All Well J Equal

Example 1

Determine the correct 5-point well index J when half edge blocks are used with a diagonal 11 x 11 grid and square gridblocks.

The simplest dataset required is as follows, representing single-phase incompressible water flow with mobility $\lambda = 1$, in a 1/4 5-spot:

```
TITLE
  Example 1
  1/4 of a 40-acre 5-spot, 11 x 11 diagonal grid, half edge blocks,
  5-point difference scheme
ENDTITLE
GRID 11 11 1
TOLERANCE .001 ! tighten solver tolerance for extra digit accuracy
EDGE             ! use half blocks on the xy edges
MISC  1  0  62.4  1.0  4e-6  3000  ! Bwi cw  denw  visw  cr pref
MAPSPRINT  1  P TX TY TXY1 TXY2  PV

PVTBO      ! any blackoil water-oil pvt table
 DENSITY  49.1   .792  1.3687316E-5  9.02E-5  deno deng  coil cvoil
 PRESSURES 1 2
       1000   8000
 PSIA       BO        RS      VISO   ! Saturated Table
  1000.00   1.29500   371.00   0.8300

SWT 1      ! Saturation Table 1
  0 0 1    ! Sw  krw  krow
  1 1 0
SGT 1
  0 0 1    ! Sg  krg  krog
  1 1 0

DELX  CON
    66
DELY  CON
    66
POROS CON
  .2
DEPTH CON
  1000
THICKNESS CON
  50
KX CON
  800
KY EQUALS KX

INITIAL
   PINIT 3000
   ZINIT 1025
   HWC 1000
```

```
ENDINIT

WELL
     I J K  PI      ! header line
 INJ
     1 1 1   -.25   ! calculate J internally and multiply it by .25
 PR
    11 11 1  -.25

MAPSFREQ 1      ! print maps each TIME
WELLTYPE
  INJ  STBWATINJ
  PR   STBWAT
RATE
  INJ 1000
   PR 1000
TIME 50
END
```

For this example (half edge blocks, constant spacings), the internally calculated J are correct. From the run printout, this internally calculated 1/4 well J is 17.89 rb-cp/d-psi, and Δp and Δp_{wb} are 103.08 and 214.86 psi, respectively. From Eqn (1), full well J is

$$J = 2(4000)/(214.8 - 103.08) = 71.6$$

and 1/4 well J is 17.9.

Example 2

Determine the correct 5-point well index J for a diagonal 10x10 grid with square gridblocks and full edge blocks.

In the base dataset above, remove EDGE and change the grid and producer location from 11 11 1 to 10 10 1. The run printout gives $\Delta p = 66.78$ psi, Eqn (1) gives full well J as

$$J = 8000/(214.8 - 66.78) = 54$$

and 1/4 well J is 13.51. Rerunning with this J value gives $\Delta p_{wb} = 214.82$ psi.

Example 3

Determine the 9-point well index J for Example 2.

Running the Example 2 dataset with the keyword NINEPOINT added gives $\Delta p = 67.82$. Eqn (1) gives J = 54.43 or 13.6 for the 1/4 wells. Rerunning with this J value gives $\Delta p_{wb} = 214.88$ psi.

Example 4

Determine the 5-point well index J for $\Delta x = 33$ ft, $\Delta y = 66$ ft, and half blocks on the edges.

Change the Example 1 dataset grid to 21x11 and Δx from 66 ft to 33 ft; change the producer location from 11,11 to 21,11. Running the resulting dataset gives an internally calculated (1/4 well) J of 19.02, and Δp_{wb} = 214.82. These results are correct with no need for Eqn (1).

Example 5

Determine the 5-point and 9-point well indices J when half blocks and the following symmetric variable spacings are used:

```
DELX  XVAR
    10.0000   20.4945   42.0025   86.0821  176.4209
   176.4209   86.0821   42.0025   20.4945   10.0000
DELY  YVAR
    10.0000   20.4945   42.0025   86.0821  176.4209
   176.4209   86.0821   42.0025   20.4945   10.0000
```

These spacings preserve the 40-acre 5-spot size when half blocks are used on the edges. Use them in the Example 2 dataset with the keyword EDGE. Running the dataset gives an internally calculated (1/4 well) 5-point J of 34.19 and Δp = 174.14 psi. Using Eqn (1) gives full well J = 196.75 or 49.19 for the 1/4 wells. Rerunning with this J value gives the correct Δp_{wb} = 214.8 psi.

Running with NINEPOINT inserted in the dataset gives dp = 172.14. Eqn (1) gives full 9-point J as 187.53 or 1/4 well J as 46.88. Inserting this J value and rerunning gives the correct Δp_{wb} = 214.8 psi.

II. The Anisotropic Case - All Well J Equal

The x,y axes are assumed to be aligned with the principal axes. Because Eqn (2) does not apply for the anisotropic case, an additional model run must be made to determine Δp_{wb}. That is, two one-layer 1/4 5-spot runs are required. Both datasets must use the same k_x,k_y,h,λ, and both must use diagonal grids or both must use parallel grids. The grid spacings must be symmetric in order that all well J are equal.

For the first of these two datasets, use:

(1) half blocks along the edges and constant Δx,Δy,

(2) the 5-point difference scheme, and

(3) internal calculation of the well J values, using the appropriate PI multiplier, e.g. - .25 for a 1/4 element.

Run this first dataset and note the resulting Δp_{wb} value. For the second dataset, use the desired grid - it can have any constant or symmetric variable spacings and half or full edge blocks. Use the difference scheme for which you desire the J value.

Run the second dataset to obtain Δp and use Eqn (1) to calculate the full well J for the grid of the second dataset. If the two dataset rates are different then adjust Δp_{wb} by the multiplicative factor q2/q1.

Example 6

Determine the 9-point well index J for the anisotropic case where k_x = 800 md and k_y = 100 md. Use full edge blocks and the variable spacing given in dataset 2 below. That grid preserves the 40-acre 5-spot size when full edge blocks are used.

Dataset 1 is the 5-point dataset of Example 1, altered for the anisotropy. Only changed entries are listed here:

```
TITLE
   Example 6  Dataset 1
      11x11 diagonal grid, half edge blocks, square blocks, 5-
      point, anisotropic, 1/4 5-spot
ENDTITLE
KY CON
   100
```

Dataset 2 is that of Example 1 with the following changes:

```
TITLE
 Example 6   Dataset 2
 10x10 diagonal grid, full edge blocks, variable spacing, 9-point,
        anisotropic, 1/4 5-spot
ENDTITLE
    GRID 10 10 1
    C  EDGE         ! remove EDGE
    NINEPOINT
    DELX  XVAR
        10.0000   20.4945   42.0025   86.0821  171.4209
       171.4209   86.0821   42.0025   20.4945   10.0000
    DELY  YVAR
        10.0000   20.4945   42.0025   86.0821  171.4209
       171.4209   86.0821   42.0025   20.4945   10.0000
    KY CON
      100

    PR
       10 10 1  -.25   ! producer in wellblock 10,10,1
```

Running dataset 1 gives Δp_{wb} = 663.94 psi. Running dataset 2 gives Δp = 407.94 psi. Eqn (1) gives full well J as

$$J = 8000/(663.94 - 407.94) = 31.25$$

and the correct 9-point 1/4 well index J is 7.8125 for the variable grid of dataset 2.

III. The Isotropic Case - Unequal Well J Values

If the x and y grid spacings are not both symmetric, then Wells 1,2,3,4 noted in the above figure will not all have the same index J. Let a pair of vectors X and Y be associated with each well. The vector X is the vector of $\Delta x(i)$ values with i starting at 1 at the well and increasing 2,3,..,N_x

away from the well. The vector Y is defined similarly using Δy(j) values. Any two wells have the same index J if and only if their X vectors are identical and their Y vectors are identical.

If Wells 1 and 2 have different indices then

$$J1 = \frac{2q/\lambda}{\Delta p_{wb} + 2*(p0-p1)} \qquad (3)$$

$$J2 = \frac{2q/\lambda}{\Delta p_{wb} + 2*(p2-p0)} \qquad (4)$$

where p0 is the grid volumetric average pressure, p1 and p2 are pressures in wellblocks 1 and 2, and Δp_{wb} is given by Eqn (2). The same equations apply when Wells 3 and 4 are used as the 5-spot wells (change subscripts 1 and 2 to 3 and 4). Again, one model run is required to obtain the p1, p2 values.

Example 7

Determine the well indices for the 40-acre, isotropic, 9-spot when the 9-point difference scheme and the following 3x3 asymmetric grid spacings with full edge blocks are used:

Δx(i) = 40, 200, 420 Δy(j) = 80, 250, 330

The X,Y vectors for the four wells are:

Well	X	Y
1	40 200 420	80 250 330
2	420 200 40	330 250 80
3	40 200 420	330 250 80
4	420 200 40	80 250 330

No two wells have the same X,Y vector set so the well indices are different for each well. The dataset required to obtain J1 and J2 is:

```
TITLE
  Example 7   dataset to obtain J1, J2 for Wells 1 and 2
    1/4 of a 40-acre 5-spot, 3 x 3 diagonal grid, full edge blocks,
    isotropic, 9-point difference scheme
    *********************************
    Omitted entries are identical to
    those in the dataset of Example 1.
    *********************************
ENDTITLE
GRID 3 3 1
NINEPOINT
C    EDGE
```

```
DELX  XVAR
    40 200 420
DELY  YVAR
    80 250 330
 PR
    3 3 1  -.25
```

Running this dataset gives p1 = 3039.24, p2 = 2991.28, and p0 = 3000 psia. Eqns (3) and (4) give 1/4 well indices:

$$J1 = \frac{1}{4} * \frac{8000}{214.8 + 2*(3000-3039.24)} = 14.67$$

$$J2 = \frac{1}{4} * \frac{8000}{214.8 + 2*(2991.28-3000)} = 10.134$$

Rerunning with these J1,J2 values gives the correct Δp_{wb} = 214.8 psi. The incorrect, internally calculated 1/4 well J1,J2 indices are 18.66 and 12.85, respectively.

To determine J3 and J4, use Wells 3 and 4 as the 5-spot wells with the change in the dataset:

```
WELL
     I J K  PI      ! header line
 INJ
     1 3 1   -.25   ! calculate J internally and multiply it by .25
 PR
     3 1 1   -.25
```

Running that dataset gives p3 = 3019.12, p4 = 2985.49, and p0 = 3000 psia. Eqns (3) and (4) give 1/4 well indices:

$$J3 = \frac{1}{4} * \frac{8000}{214.8 + 2*(3000-3019.12)} = 11.328$$

$$J4 = \frac{1}{4} * \frac{8000}{214.8 + 2*(2985.49-3000)} = 10.765$$

Rerunning with these J3,J4 values gives the correct Δp_{wb} = 214.8 psi. The incorrect, internally calculated 1/4 well J3,J4 indices are 13.57 and 12.98, respectively.

IV. The Anisotropic Case - Unequal Well J Values

The procedure here is the same as in Section III except that the Δp_{wb} value in Eqns (3) and (4) must be determined from an additional model run as described in Section II.

Appendix 11 Relative Permeability Endpoint Checking

Sensor allows different relative permeability endpoint saturations for each grid block. These endpoint saturations are Swc, Sorw, Sorg, Sgc, Sgr, and (for Baker relative permeability) Swcg. The Saturation Table relative permeabilities are normalized with respect to the Table endpoint saturations. For each grid block those Table-normalized rel perms are denormalized using the grid block endpoint saturations to obtain rel perm, as described in Appendix 6.

Gridblock endpoint saturations are set equal to those of the Saturation Table assigned to the block, unless they are overridden by entry of the endpoint saturation arrays SWC, SORW, SORG, SGC, SGR, SWCG. A final check and reset of Sgr is: if Sgr < Sgc, Sgr = Sgc.

If the Swc array is entered, then erroneous negative saturation intervals can arise for kro and krg. The term "negative saturation interval" denotes the case where there is no saturation interval where kr > 0.

The NEWSOR feature is provided to protect against this error. In default mode this NEWSOR feature is ON (active) and may result in internal change of an endpoint saturation.

If the array SWC is not entered there is no need for or effect of the entry NEWSOR.

As an example, consider a grid block assigned to a Table having Swc=.2 and Sorw=.4. If an entered SWC array assigns an Swc of .7 and no SORW array is entered, the block will have Swc=.7 and Sorw=.4, giving a negative saturation interval of -.1 for the krow rel perm.

IF AND ONLY IF THE ARRAY SWC IS ENTERED:

The default NEWSOR causes the following changes in endpoint saturations.

Definitions:

Swc*, Sorw*, Sorg*, Sgc*, Sgr*, Swcg* = endpoints from the Saturation Table

Swc**, Sorw**, Sorg**, Sgc**, Sgr**, Swcg** = endpoints entered by arrays SWC, SORW, etc.

Swc, Sorw, Sorg, Sgc, Sgr, Swcg = final endpoint saturations

Settings:

Swc = Swc**

Each of Sorw, Sorg, etc. is equal to Sorw**, Sorg**, etc. if SORW, SORG, etc. is entered.

For endpoint saturations where the corresponding endpoint array is NOT entered:

f = (1-Swc**)/(1-Swc*)

Sorw = f x Sorw* if Sorw** array SORW is not entered

Sorg = f x Sorg* if Sorg** array SORG is not entered

Sgc = f x Sgc* if Sgc** array SGC is not entered

Sgr = f x Sgr* if Sgr** array SGR is not entered

Swcg = f x Swcg* if Swcg** array SWCG is not entered

To deactivate this NEWSOR feature, insert the Initial Data entry NEWSOR 0, and the program will change no endpoint saturations regardless of what endpoint arrays are entered and regardless of what errors occur.

If the array SWC is not entered but some other endpoint arrays are entered, saturation interval errors may occur.

In any event, whether endpoint arrays are entered or not, Sensor flags saturation endpoint interval errors with *WARNING messages or *ERROR messages.

Capillary pressure curves are treated as functions of normalized saturation, Pcwo = Pcwo(Sw/(1-Swc)) and Pcgo = Pcgo(Sg/(1-Swc)).

10 Tables

Table 1 The Datafile Structure

```
C           *** START OF INITIAL DATA ***
TITLE
  ... any number of 132-column alphanumeric lines ..
ENDTITLE
GRID  Nx Ny Nz
C            default solver is RBILU(0)
C            default formulation is Impes
D4          !  if the direct-solution D4 solver is to be used
NF          !  if the Nested Factorization solver is desired
IMPLICIT    !  if the fully Implicit formulation is to be used
MISC  ... 6 numbers ...   !  water properties, pref
PVTEOS      !  if this is a compositional problem
  ... equation of state data
PVTBO       !  if this is a black oil problem
  ... black oil pvt table
SWT n (SGT n)
  ... tables giving kr and Pc vs saturation
COMPACTABLE  !  if compaction is used for rock compressibility
  ... compaction tables ..
DEPTH    VALUE or CON or ZVAR or.. etc.
  ... array data ...
THICKNESS  VALUE or ...
  ... array data ...
KX (and/or TX)  VALUE or ...
  ... array data
KY (and/or TY)  VALUE or ...
  ... array data ...
KZ (and/or TZ) VALUE or ...
  ... array data ...
POROS (and/or PV) VALUE or ...
  ... array data ...
SEP         !  surface separator data if compositional problem
  ... data ...
INITIAL
  ... data specifying initial p, woc, goc, etc.
ENDINIT     !  end of Initial Data

C           *** MODIFICATION DATA START HERE ***

MODIFY PV  !  optional, to modify pore volumes
  ... data
MODIFY TX  !  optional, to modify transmissibility Tx
  ... data
... other optional additional MODIFY data

C            *** RECURRENT DATA START HERE ***

C          WELL data must preceed other well data
WELL
  ... well locations and perf data
C  Remaining data here are order independent,
C   except for chronological order of TIME and data changes.
WELLTYPE
 ... defines wells as producers or injectors, and defines units
LIMITWELL        ! optional
  ... economic limit data for wells
```

```
BHP              ! for wells not using thp tables
  ...  minimum flowing bhp values for bhp wells
THP              ! for wells using thp tables
  ... minimum wellhead pressures for thp wells
THPTABLE         ! needed only if there are thp wells
  ... tubinghead pressure tables for thp wells
RATE
  ... well rate data
WELLPLAT         ! if any platforms are to be used
  ... assign wells to platforms
  ... other platform keywords/data
LIMITFIELD  ... 5 numbers !  optional field economic limit data
TIME t1          ! proceed to time = t1
TIME t1 dtime    ! optional, in place of previous line,
C           proceed to time t1, with printout every dtime days

WELL  ! optional, to add new wells or change perfs of old ones
  ... data
RATE      !   optional,   to   change   well   rates   (default   rates   are
      !  shutin, with no recirculation)
  ... data
TIME t2          ! proceed to time = t2
...  WELL, RATE, etc. data changes ...
TIME t3
... etc. ...  data changes  ... time entries
...
TIME tlast
END
```

Table 2 Example Datafile

```
(Voluminous and Repetitive Portions Omitted)
C                 *** START INITIAL DATA ***
TITLE

  test.dat
  IMPES
  3-COMPONENT COMPOSITIONAL
  44x26x12 GRID    8030-DAY HISTORY RUN
  EXAMPLE OF INPUT FORMATS  - NONEXECUTABLE FILE
  *** TIMESTEP CONTROL ENTRIES:  DTMIN 30

ENDTITLE

GRID  44 26  12

MISC  1.047 3E-6  62.07 .35 10E-6 7120  !   Bwi cw denw visw cr pref

PVTEOS
    266.          ! reservoir temperature, deg F

    CPT     PC      TC     AC   MW  ZCRIT PCHOR  OMEGA OMEGB OMEGAS OMEGBS
     C1   673.1   344.22 .008  16.04 .355  77.  .43933 .0777  .510  .082
   C2-6   636.3   670.95 .1636 47.38 .335 159.6 .492   .081   .488  .085
    C7+   268.43 1328.2  .578  228   .27  687   .436   .072   .435  .075

BIN
  .00237 .13908   ! b12 b13
         .00301   !     b23
BINS
  .00237 .05096
         .00301

INITIAL
   DEPTH   ! follow with any number of lines to define composition vs depth
     10400. 5553.7 .5898 .1926 .2176  ! depth psat {zi}
   PINIT  7120
   ZINIT 10400
   HWC   10435
C     initial pressure = 7120 psia at depth = 10400 ft,
C      hydrocarbon-water contact at depth = 10435 ft

SEP 1
  1014.7 150.0   ! first stage p psia, T deg F
   264.7  80.0
    14.7  60.0

SWT 1
C    Sw        krw                krow    Pcwo   Pcwoimb (optional)
   0.15000    0.000              1.000     0.0    0.0
   0.18186    0.001              0.958     0.0    0.0
   0.21258    0.037              0.917     0.0    0.0
   0.24220    0.069              0.876     0.0    0.0
   0.27024    0.105              0.850     0.0    0.0
   0.31140    0.155              0.757     0.0    0.0
   0.34580    0.210              0.657     0.0    0.0
   0.38364    0.250              0.557     0.0    0.0
   0.42842    0.305              0.318     0.0    0.0
   0.48000    0.390              0.000     0.0    0.0
   1.00000    0.700              0.000     0.0    0.0
```

```
SGT 1
C    Sg          krg     krog              Pcgo
   0.00000     0.000   1.000              0.0
   0.02000     0.050   0.950              0.0
   0.05000     0.100   0.780              0.0
   0.10000     0.200   0.600              0.0
   0.15000     0.300   0.400              0.0
   0.20000     0.400   0.285              0.0
   0.25000     0.500   0.220              0.0
   0.30000     0.550   0.180              0.0
   0.40000     0.600   0.075              0.0
   0.45000     0.630   0.050              0.0
   0.50000     0.650   0.020              0.0
   0.57000     0.670   0.000              0.0
   0.85000     1.000   0.000              0.0

  ***********  TOTAL OF 12 SATURATION TABLES  ************

COMPACTABLE     ! compaction tables  cr*10**6
   9000. 0.
   1 10  8
       .25 .27 .29 .31 .33 .35 .37 .39  ! --> porosity
C stress psi
    0.      2.63 2.80 2.60 2.17 2.20 2.23 3.03 3.70
 2500.      2.63 2.85 2.73 2.39 2.50 2.61 4.47 7.24
 3000.      2.64 2.86 2.75 2.63 3.06 4.80 7.85 13.3
 3500.      2.64 2.88 2.84 3.12 4.60 8.31 13.4 21.4
 4500.      2.65 2.97 3.48 5.11 8.29 13.1 33.3 60.8
 5000.      2.65 3.06 4.11 10.4 17.2 28.3 48.3 79.0
 5500.      3.98 4.85 6.87 12.5 23.0 40.8 65.1 91.4
 6500.      3.98 4.83 7.49 15.8 32.2 58.7 92.1 124.
 7500.      3.99 5.15 9.14 20.6 42.3 74.1 108. 130.
 8000.      3.99 5.28 10.7 27.0 54.7 84.5 103. 110.
2 10  8
       .25 .27 .29 .31 .33 .35 .37 .39
    0.      1.79 2.89 4.13 4.69 4.76 4.80 5.24 5.93
 2500.      1.79 2.89 4.14 4.69 4.77 4.80 5.24 6.34
 3000.      1.79 2.90 4.16 4.78 6.01 12.0 31.0 66.3
 3500.      1.80 2.90 4.16 4.94 7.31 18.2 46.0 86.1
 4500.      1.80 2.91 4.37 8.74 24.9 57.7 141. 171.
 5000.      1.80 2.91 5.69 24.9 64.5 117. 150. 160.
 5500.      2.70 4.62 10.8 31.4 71.2 111. 130. 134.
 6500.      2.76 7.64 24.7 57.1 88.8 104. 106. 106.
 7500.      3.80 13.6 36.6 65.9 83.7 89.4 89.7 89.7
 8000.      5.45 18.6 43.5 65.8 75.6 77.4 77.4 77.4

COMPACTYPE ZVAR
 4*1 8*2

DELX  XVAR
   4921.300    3280.800    2132.500     820.200     820.200     574.200
    574.200     519.500     519.500     519.500     519.500     519.500
    519.500     656.000     656.000     656.000     656.000     574.000
    574.000     656.000     656.000     656.000     656.000     738.000
    738.000     820.000     820.000     738.000     738.000     902.500
    902.500     820.200     820.200     820.200     820.200     820.000
    820.000    1805.000    1968.000    2461.000    2734.000    2734.000
   2734.000    4921.000
DELY YVAR
   9582.000    3714.000    1750.000    1368.000     686.000     686.000
    686.000     686.000     820.000     820.000     656.000     656.000
    656.000     656.000     656.000     656.000     820.000     820.000
    686.000     686.000     686.000     686.000    1368.000    2130.000
```

```
   3822.000    4711.000
THICKNESS  VALUE
     35.5620     32.9130     30.5150     28.8100     27.6780     26.6000
     25.6280     24.7990     24.2550     23.7450     23.2530     22.7790

         ************** Nx * Ny * Nz values, x first then y then z *****

POROS VALUE
      0.1693      0.1828      0.1725      0.1669      0.1642      0.1620
      0.1600      0.1587      0.1575      0.1564      0.1552      0.1541
          ************
SWC  VALUE
      0.4000      0.4000      0.4000      0.4000      0.4000      0.4000
      0.4000      0.4000      0.4000      0.4000      0.4000      0.4000
          ************
       **** arrays TX,TY,TZ,ROCKTYPE all entered with VALUE ****
DEPTH  LAYER
 10500.0000  10500.0000  10489.0000  10486.0000  10484.0000  10482.0000
 10480.0000  10479.0000  10477.0000  10475.0000  10473.0000  10472.0000
    ** a total of Nx * Ny values  ********

ENDINIT     !  END OF INITIAL DATA

C            ***  START RECURRENT DATA  ***

C            NOTE: PI IS RB-CP/DAY-PSI
C            WELL NUMBERS USED HERE.  CAN USE 8-CHARACTER WELLNAMES.
WELL
             I     J    K                    PI   ! header line
       101
             24    5    5                   3.195
             24    5    7                   .2000
             24    5    8                   .2000
             24    5    9                   2.000
       217
             38    6    2                   1.070
             38    6    4                   .3060
     **** etc. 150 wells ****
WELLTYPE
 101  -   323    STBOIL    !  stboil/day producer
 3027 -   3147   MCFINJ    !  mcf/day gas injector
 401  -   508    STBWATINJ !  stbwater/day injector

BHP
 101  -  323     500.
 3027 -  3147    9200.
 401  -  508    12500.

INJGAS
 3027 - 3147            ! wells 3027-3147 inclusive
   .8601  .1343  .0056  ! injected gas composition

RATE
 315     6990.1

DTMIN   30             ! force 1 month time steps (use DTMIN only for history
                       !  runs and only when it works ok)
TIME  30.4

RATE
 315    10623.2

TIME  60.8
```

```
RATE
 138     3104.7
 315    10202.9

TIME   91.4

RATE
 118     5217.8
 138     5626.3
 248      259.0
 315     6780.2

TIME  121.8

RATE
 118     5364.0
 138     2608.2
 315     3975.6
TIME   152.2
   ***************  etc.

TIME   8030.000
END             ! END OF DATAFILE
```

Table 3 Description of Test Problems

Problems spe1,2,3,5,7,9 and 10 are the SPE Comparative Project Problems of the same number, described in References 37-43. Equivalent data files for other simulators should be provided by or available from the other vendors.

All black oil datafiles illustrate PVTBO. All compositional datafiles illustrate PVTEOS. All datafiles have corresponding outfiles with extensions .out. For example, there are outfiles spe1.out, spe2.out, test4a.out, etc.

Example SensorPlot datafiles for X-Y plots from Plot2Excel are denoted by extension ".sp" for spe1,spe2,spe3,spe5,spe7,spe9,spe10,test3a, test7, and test16. The SensorMap datafiles spe9.sm and spe10.sm illustrate data for contour maps from Map2Excel.

The cpu time for each problem, printed at the end of the outfile, is for a 2.8 GHz PC. The primary purpose of these datafiles is illustration of the Sensor keyword input, and certain Sensor features. However, several of the problems are of particular interest in comparing Sensor to other models in respect to robustness/stability and efficiency. These latter problems include test2, test3, test4, spe9, and spe10. You can run those problems on the model of your choice and compare results to the Sensor results.

datafile	grid	blocks/active	pvt	comments
spe1.dat	10 10 3	300/300	3-phase black oil	Gas injection
spe2.dat	10 1 15	250/250	3-phase black oil	Radial coning
spe3.dat	9 9 4	324/324	9-cpt condensate	Cycling, blowdown
spe5a.dat	7 7 3	147/147	6-cpt oil	Wag, Scenario 1
spe5b.dat				Wag, Scenario 2
spe5c.dat				Wag, Scenario 3
spe7_1a.dat	9 5 6	270/270	3-phase black oil	Horizontal well, Case 1a
spe7_4a.dat				Case 4a
spe9.dat	24 25 15	9000/9000	3-phase black oil	Depletion, constant dip slab
spe10_case1.dat	100 1 20	2000/2000	3-phase black oil	Gas injection in an xz cross-section with geostatistical kx,kz
spe10_case2.dat	10 20 10	2000/2000	2-phase black oil	Water-oil problem. Water injection in a 2000-block upscaled grid.
blackoil.dat	5 5 5	125/125	3-phase black oil	Automatic BLACKOIL conversion from EOS
convert.dat	24 25 15	9000/9000	3-phase black oil	Automatic well conversion on shutin
fcm1.dat	10 1 1	10/10	8-cpt	Pseudoized to 2 cpt, first-contact miscible (FCM) option
fcm2.dat	7 7 43	2107/2060	8-cpt	Pseudoized to 2 cpt, first-contact miscible (FCM) option
init1.dat	1 1 50	50/50	6-cpt	INITIAL equilibrium test.
misc.dat	various		16-cpt	contains misc1.dat, misc2.dat, misc3.dat, comparing coarse and fine grid MCM runs in 1D, 2D, and 3D.

opt_pcon.dat	50 10 10 5000/5000	6-cpt	Pressure control (PCON) and platform optimization (OPTPLAT).
paul1.dat	10 10 3 300/300	coal, black oil	Coal bed methane case 1 from SPE 20733
paul2.dat	10 10 2 200/200	coal, black oil	case 2
paul3.dat	10 10 2 200/200	coal, black oil	case 3
salt1.dat	24 15 15 5400/5400	3-phase black oil	Salinity option, 900 day depletion
salt2a.dat	10 1 4 40/20	3-phase black oil	Test of non-neighbors with SALT option.
salt2b.dat	10 1 4 40/20	3-phase black oil	salt2a and salt2b should give same results, 2a with no non-neighbors, 2b with non-neighbors
septable.dat	7 1 3 21/21	6-cpt	Spe5-based test of separator liquid recovery factor table
spe5_pcon.dat	7 7 3 147/147	6-cpt	spe5-based test of field pressure control (PCON)
spe20178.dat	various	3-cpt	Stalkup and Dean cases examining effect of refinement on FCM cases (using OIL option)
spe79692.dat	10 10 10 1000/1000	7-cpt	FCM case from SPE 79692
test1.dat	23 16 26 9568/8832	multiple pvt	This datafile illustrates input data for multiple reservoirs (stacked), multiple pvt type (black oil and compositional), regions and superregions, reservoir-dependent tracers, kvtable, and kvtable with eos reversion using tracers.
test2.dat	300 1 1 300/300	24-cpt oil	1D near-miscible gas injection
test3a.dat	1 41 136 5576/5576	3-phase black oil	Pronounced water and gas coning in horizontal well depletion of a thin viscous oil column in a highly permeable formation, with CO2 injection and produced water reinjection.
test3b.dat	1 41 136 5576/5576	6-cpt	6-component compositional version of test3a.dat.
test4a.dat*	50 10 10 5000/5000	3-phase black oil	Depletion of critical-point black oil Initial composition varies with depth.
test4b.dat*	50 10 10 5000/5000	6-cpt	6-component compositional version of test4a.dat.
test5.dat	7 7 10 490/490	3-phase black oil	Illustrates tubinghead pressure table entry and use.
test6.dat	10 10 3 300/300	3-phase black oil	This file illustrates a 5-fold difference in cpu time for pattern flood simulation, depending upon the type of grid used. See comments in Title of test6.dat and plots at the end of test6.out.
test7.dat	24 25 15 9000/9000	3-phase black oil	This problem illustrates use of tracers to determine what fraction of the oil produced from a given

					(equity) block came from each of the several blocks.
test8.dat	20	1	13	260/260	6-cpt oil	Single core, forced-imbibition blowdown lab experiment.
test9.dat	1	1	1	1/1	5-cpt oil	1-block run illustrating compaction with water weakening.
test10.dat	5	1	1	5/5	8-cpt oil	5-block run illustrating Baker's saturated-weighted linear interpolation for krw and kro.
test11.dat	10	1	5	50/50	Foamy oil	Foamy oil problem.
test11a.dat	10	10	3	300/300	Foamy oil	Foamy oil problem. Constant dip slab.
test12.dat	11	11	3	363/363	3-phase black oil	Oil reservoir 9-spot pattern with gas injection.
test13.dat*	11	11	1	121/121	2-phase black oil	Water-oil, 5-spot pattern waterflood with nine-point.
test14.dat	23	16	26	9568/8832	2-phase black oil	Multiple reservoir, water-oil, implicit, depletion
test15.dat	11	11	1	121/121	3-phase black oil	Multiple reservoir (areally embedded) multiple pvt, with nine-point
test16.dat	10	1	5	50/50	3-phase black oil	Single-well radial depletion of a lean gas condensate. Illustrates re-entry of black oil pvt table in Recurrent Data at times of change in Surface Process conditions. See SensorPlot datafile test16.sp to generate x-y plots with Plot2Excel.
turb.dat	1	1	4	4/4	2-phase black oil	Gas/water well test example
turba.dat	1	1	4	4/4	2-phase black oil	illustrating determination of layer D values for turbulence option (BETA) used in turba.dat
wag1.dat	10	10	3	300/300	3-phase black oil	Spe1-based demonstrations of WAG
wag2.dat	10	10	3	300/300	3-phase black oil	injectors

* File test4.dat contains datasets test4a.dat, test4b.dat
File test13.dat contains datasets test13a,b,c, including a 15x8x1 parallel grid.

Table 4 Example of Printout of Gridblock Property Ranges

```
 AVERAGE AND MINIMUM/MAXIMUM VALUES AND GRID LOCATIONS
-----------------------------------------------------
PROPERTY      MINIMUM    I  J  K       MAXIMUM    I  J  K    AVERAGE
----------  ----------  --------    -----------  --------  ----------
PV          .5077E+05   6 22  1      .1625E+10  44  1 12   .6780E+07   RB
THICKNESS   .4037E+01   5 23  1      .7839E+03  21 10 12   .1230E+03   FT
DEPTH       .9486E+04  22 13  1      .1181E+05  17  1 12   .1051E+05   FT
TX          .1000E-04   2 25  1      .2652E+03  23 10 12   .2823E+01   RB-CP/D-PSI
TY          .1000E-04   1 26  2      .1914E+03  21 11 12   .8713E+00   RB-CP/D-PSI
TZ          .9000E-04  15 26 10      .9717E+03  16 15  7   .3235E+01   RB-CP/D-PSI

POROSITY    .4990E-01  10 26 10      .4196E+00  22 16  5   .2133E+00   FRACTION
```

Table 5 Platform Table Printout for Example 4 of Section 3.4

```
                                        PLATFORM TABLE
                                 ------------------------------------------------------------
                                          TIME STEP FROM       .0  TO      2.0 DAYS
                                 RATES:  STB/D MCF/D    CUMULATIVES:  MSTB MMCF    GOR:  SCF/STB
                                 ------------------------------------------------------------

              PRODUCTION RATES          CUMULATIVE PRODUCTION                    INJN. RATES      CUM. INJECTION
             ------------------------   ------------------------    %            --------------   --------------    ONTIME
PLATFORM     WATER       OIL      GAS   WATER      OIL      GAS   WCUT    GOR    WATER      GAS   WATER      GAS PROD WINJ GINJ
-------------------------------------------------------------------------------------------------------------------------------
       1         1     11462    50000       0       23      100     .0  4362        0     2000       0        4 1.00 1.00 1.00
       2         1     17388    80000       0       35      160     .0  4600        0    81900       0      164 1.00 1.00 1.00
       3         0      7687    40000       0       15       80     .0  5203        0        0       0        0 1.00 1.00 1.00

  TOTALS         2     36537   170000       0       73      340     .0  4652        0    83900       0      168

PLATFORM RATE STATUS CODE:
               SI      shut in
               Q       target rate
               PC      pressure constraint
               RC      constrained by entered maximum well rates
               QW      constrained by water rate limit
               QG      constrained by gas rate limit
               IC      production rate is reduced due to gas injection capacity
 PLATFORM    PRODN    GINJ    WINJ
 ------------------------------
        1       Q       Q      SI
        2       Q       Q      SI
        3       Q      SI      SI   COSMETIC


                                  PLATFORM GAS RATES (MCF/D)  TIME =      2.0 DAYS
                                 ----------------------------------------------

                        GAS PRODUCTION RATES                          GAS INJECTION RATES
                       ---------------------------------------       ---------------------------------------------------
     PLATFORM    GROSS PROD       FUEL       SALE    NET PROD       PRODUCED    OUTSIDE    TRANSFER    UNACCNT       TOTAL
     ---------------------------------------------------------------------------------------------------------------------
        1           50000.      1000.     10000.     39000.            0.       2000.         0.      19500.       2000.
        2           80000.      1600.     20000.     58400.          58400.      4000.     19500.         0.      81900.
        3           40000.                                                                                           0.

   TOTALS          170000.      2600.     30000.     97400.          58400.      6000.     19500.     19500.      83900.

      BALANCE =   NET PROD +   OUTSIDE - TOTAL INJECTION - UNACCOUNTED (NON-COSMETIC PLATFORMS)
              =     97400. +     6000. -    83900.       -    19500.
              =      0. MCF/D (      .0 MMCF CUMULATIVE =   .0000 FRACTION OF CUM GROSS GAS PRODUCED)


                                PLATFORM GAS CUMULATIVES (MMCF)  TIME =      2.0 DAYS
                                -------------------------------------------------------

                          GAS PRODUCTION                              GAS INJECTION
                       ---------------------------------------       ---------------------------------------------------
     PLATFORM    GROSS PROD       FUEL       SALE    NET PROD       PRODUCED    OUTSIDE    TRANSFER    UNACCNT       TOTAL
     ---------------------------------------------------------------------------------------------------------------------
        1             100.         2.        20.        78.             0.         4.         0.        39.          4.
        2             160.         3.        40.       117.           117.         8.        39.         0.        164.
        3              80.         0.         0.         0.             0.         0.         0.         0.          0.

   TOTALS             340.         5.        60.       195.           117.        12.        39.        39.        168.
```

PLATFORM GAS COMPOSITIONS TIME = 2.0 DAYS

	GAS PRODUCTION				GAS INJECTION			
PLATFORM	GROSS PROD	FUEL	SALE	NET PROD	PRODUCED	OUTSIDE	TRANSFER	TOTAL
1								
C1N2	.78179	.78179	.78179	.78179	.78000	.85000	.00000	.85000
C2-3	.17949	.17949	.17949	.17949	.17000	.13000	.00000	.13000
C4F2	.03819	.03819	.03819	.03819	.04000	.02000	.00000	.02000
F3F4	.00044	.00044	.00044	.00044	.01000	.00000	.00000	.00000
F5	.00009	.00009	.00009	.00009	.00000	.00000	.00000	.00000
F6	.00000	.00000	.00000	.00000	.00000	.00000	.00000	.00000
F7	.00000	.00000	.00000	.00000	.00000	.00000	.00000	.00000
2								
C1N2	.78417	.78417	.78417	.78417	.78417	.85000	.78000	.78639
C2-3	.17728	.17728	.17728	.17728	.17728	.13000	.17000	.17324
C4F2	.03801	.03801	.03801	.03801	.03801	.02000	.04000	.03761
F3F4	.00044	.00044	.00044	.00044	.00044	.00000	.01000	.00269
F5	.00009	.00009	.00009	.00009	.00009	.00000	.00000	.00007
F6	.00000	.00000	.00000	.00000	.00000	.00000	.00000	.00000
F7	.00000	.00000	.00000	.00000	.00000	.00000	.00000	.00000
3								
C1N2	.78664							.78664
C2-3	.17508							.17508
C4F2	.03775							.03775
F3F4	.00044							.00044
F5	.00010							.00010
F6	.00000							.00000
F7	.00000							.00000

Table 6 Example of Timestep Table

TIME (STEP): 500.0 (26) ITNS (CUM): 1 (45) % REC. OIL (GAS): 6.14 (14.84) PAVG (HC): 3034.1 (2907.0)
DXMAX(15,16, 2) .014 DSMAX(11,14, 2) .034 DPMAX(14,15, 1) -88.7 TSCUTS 0 VOLERR .000059 SSAT .00794
GRIDBLOCKS: TOTAL,WATER,HC,3PHASE 9000 0 9000 8852 HCPV GINJ = .000 HCPV WINJ = .003
SOLVER ITERATIONS (TOTAL / MAX / # MAX): 211 / 9 / 0 AVG ITNS/SOLVE = 4.69

	QWI	QGI	QW	QO	QG	GOR	WCUT	QGASLIFT	
RATES	1547.3	.0	2818.2	20923.2	118251.2	5651.7	.119	0	(STB/D,MSCF/D)
CUMS	614.1	.0	720.4	13358.8	44896.9			0	(MSTB,MMSCF)

MASS IN PLACE (Units are MSTB and MMSCF)

COMPONENT	INITIAL	CURRENT	PRODUCED	INJECTED	MB
OIL	217702.10	204343.29	13358.81	.00	1.000000
GAS	302603.88	257707.02	44896.85	.00	1.000000
WATR	210813.59	210707.32	720.37	614.10	1.000000

REGION TABLE

TIME STEP FROM 480.0 TO 500.0 DAYS

	CUM PRODUCTION			CUM INJECTION		PRODUCTION RATES			INJECTION RATES				%	%	
REGION	WATER	OIL	GAS	WATER	GAS	WATER	OIL	GAS	WATER	GAS	GOR	WCUT	OIL REC	GAS REC	HC PAVG
1	87	11339	39960	0	0	378	18435	107196	0	0	5815	.020	7.2	18.3	2872
2	632	2018	4936	614	0	2440	2488	11055	1547	0	4444	.495	3.3	5.8	2996

Table 7 Example of Platform Table

PLATFORM TABLE

TIME STEP FROM 321.7 TO 500.0 DAYS

PLATFORM	CUM PRODUCTION WATER	CUM PRODUCTION OIL	CUM PRODUCTION GAS	CUM INJECTION WATER	CUM INJECTION GAS	PRODUCTION RATES WATER	PRODUCTION RATES OIL	PRODUCTION RATES GAS	INJECTION RATES WATER	INJECTION RATES GAS	GOR	% WCUT	ONTIME PR	ONTIME WI	ONTIME GI
1	75	7574	31174	659	0	324	11833	69784	1648	0	5897	2.7	1.00	1.00	1.00
2	835	5993	19794	0	0	2341	8710	39177	0	0	4497	21.2	1.00	1.00	1.00
TOTALS	910	13567	50968	659	0	2666	20542	108961	1648	0	5304	11.5			

PLATFORM RATE STATUS CODE:

SI	shut in
Q	target rate
PC	pressure constraint
RC	constrained by entered maximum well rates
QW	constrained by water rate limit
QG	constrained by gas rate limit
IC	production rate is reduced due to gas injection capacity

PLATFORM	PRODN	GINJ	WINJ
1	PC	SI	PC
2	PC	SI	SI

PLATFORM GAS RATES (MCF/D) TIME = 500.0 DAYS

PLATFORM	GAS PRODUCTION RATES GROSS PROD	FUEL	SALE	NET PROD	GAS INJECTION RATES PRODUCED	OUTSIDE	TRANSFER	UNACCNT	TOTAL
1	69784.	0.	0.	69784.	0.	0.	0.	69784.	0.
2	39177.	0.	0.	39177.	0.	0.	0.	39177.	0.
TOTALS	108961.	0.	0.	108961.	0.	0.	0.	108961.	0.

PLATFORM GAS CUMULATIVES (MMCF) TIME = 500.0 DAYS

PLATFORM	GAS PRODUCTION GROSS PROD	FUEL	SALE	NET PROD	GAS INJECTION PRODUCED	OUTSIDE	TRANSFER	UNACCNT	TOTAL
1	31174.	0.	0.	31174.	0.	0.	0.	31174.	0.
2	19794.	0.	0.	19794.	0.	0.	0.	19794.	0.
TOTALS	50968.	0.	0.	50968.	0.	0.	0.	50968.	0.

PLATFORM GAS COMPOSITIONS TIME = 500.0 DAYS

PLATFORM	GAS PRODUCTION GROSS PROD	FUEL	SALE	NET PROD	GAS INJECTION PRODUCED	OUTSIDE	TRANSFER	TOTAL
1								
OIL	.00000	.00000	.00000	.00000	.00000	.00000	.00000	.00000
GAS	1.00000	1.00000	1.00000	1.00000	1.00000	.00000	.00000	.00000
2								
OIL	.00000	.00000	.00000	.00000	.00000	.00000	.00000	.00000
GAS	1.00000	1.00000	1.00000	1.00000	1.00000	.00000	.00000	.00000

Table 8 Example of Well Table

WELL TABLE

TIME STEP FROM 321.7 TO 500.0 DAYS

WELL	PLAT	TYPE	LOCATION I	J	K	PRESSURES PSIA THP	BHP	GRID	RATES (STB/D & MCF/D) WATER	OIL	GAS	% WCUT	GOR	CUMS (MSTB & MMCF) WATER	OIL	GAS	STAT	SG	SW
P2	1	PROD	5	1	2	0	1048	1460	0	61	1131	.0	18540	0	71	604	PC	.160	.174
			5	1	3		1050	1417	0	321	2795	.0	8684	0	313	1544		.128	.175
			5	1	4		1052	1483	0	61	410	.0	6676	0	56	215		.116	.176
TOTALS									0	444	4337	.0	9760	0	442	2364			
P3	1	PROD	8	2	2	0	1109	1414	0	72	1084	.0	14868	0	101	631	PC	.151	.187
			8	2	3		1111	1409	0	253	1659	.0	6545	0	298	1157		.117	.190
			8	2	4		1114	1689	0	10	57	.0	5563	0	9	31		.105	.193
TOTALS									0	337	2802	.0	8318	0	410	1820			
... ETC. ...																			
P14	1	PROD	8	13	2	0	1429	1472	0	1977	11844	.0	5990	0	776	4201	Q	.111	.237
			8	13	3		1431	1501	0	498	2283	.0	4582	0	198	806		.103	.190
			8	13	4		1434	1585	0	24	91	.0	3752	0	11	34		.095	.193
TOTALS									0	2500	14219	.0	5687	0	986	5043			
WI	1	WINJ	24	25	11	0	4543	3863	57	0	0			22	0	0	PC	.000	.881
			24	25	12		4551	4275	83	0	0			32	0	0		.000	.882
			24	25	13		4559	4376	174	0	0			69	0	0		.000	.882
			24	25	14		4574	4451	780	0	0			312	0	0		.000	.882
			24	25	15		4607	4451	551	0	0			221	0	0		.000	.882
TOTALS									1648	0	0			659	0	0			
TOTALS	PLATFORM 1 PRODUCTION								324	11833	69784	2.7	5897	75	7574	31174			
TOTALS	PLATFORM 1 INJECTION								1648	0	0			659	0	0			
P15	2	PROD	11	14	2	0	1170	1484	16	269	1234	5.6	4571	2	250	806	PC	.091	.360
			11	14	3		1173	1472	0	553	1790	.1	3236	0	410	1061		.087	.304
			11	14	4		1176	1794	0	10	40	.0	3892	0	6	19		.091	.244
TOTALS									16	834	3065	1.9	3676	3	668	1888			
P16	2	PROD	13	15	2	0	1211	1531	0	1	15	.1	9196	0	2	11	PC	.121	.305
			13	15	3		1213	1486	45	315	1124	12.5	3565	15	390	1024		.076	.372
			13	15	4		1216	1846	0	8	27	1.5	3276	0	6	15		.078	.347
TOTALS									45	325	1167	12.2	3587	15	400	1051			
.. ETC. ...																			
P26	2	PROD	17	25	2	0	1292	2061	49	40	143	55.4	3562	17	70	144	PC	.049	.421
			17	25	3		1297	2165	5	14	54	26.3	3765	1	16	37		.059	.389
			17	25	4		1303	2654	18	18	47	48.9	2518	5	27	43		.044	.411
TOTALS									73	74	245	49.9	3333	24	114	226			
TOTALS	PLATFORM 2 PRODUCTION								2341	8710	39177	21.2	4497	835	5993	19794			
TOTALS	PLATFORM 2 INJECTION								0	0	0			0	0	0			
TOTALS	FIELD PRODUCTION								2666	20542	108961	11.5	5304	910	13567	50968			
TOTALS	FIELD INJECTION								1648	0	0			659	0	0			

Table 9 Example of End-of-Run Summaries

WELL WI SUMMARY (INJ STBWAT/D PLATFORM 1 FIRST PERF: I,J,K= 24 25 11)

TIME		INJECTION RATES		CUM INJECTION		PRESSURE PSIA			SATN. %		
DAYS	STEP	WATER	GAS	WATER	GAS	PTH	PBH	GRID	SG	SW	STAT
10.00	1	834	0	8	0	0	4544	4258	.0	88.2	PC
25.00	2	811	0	20	0	0	4544	4227	.0	88.2	PC
47.50	3	880	0	40	0	0	4544	4199	.0	88.2	PC
81.25	4	950	0	72	0	0	4544	4173	.0	88.1	PC
131.88	5	1028	0	124	0	0	4544	4143	.0	88.1	PC
207.81	6	1143	0	211	0	0	4544	4095	.0	88.1	PC
321.72	7	1350	0	365	0	0	4544	4002	.0	88.1	PC
500.00	8	1648	0	659	0	0	4544	3863	.0	88.1	PC

WELL P2 SUMMARY (PROD STBOIL/D PLATFORM 1 FIRST PERF: I,J,K= 5 1 2)

TIME		PRODUCTION RATES			%		CUM PRODUCTION			PRESSURE PSIA			SATN. %		
DAYS	STEP	WATER	OIL	GAS	WCUT	GOR	WATER	OIL	GAS	PTH	PBH	GRID	SG	SW	STAT
10.00	1	0	1308	1749	.0	1337	0	13	17	0	3045.64	3455.94	2.4	17.3	Q
25.00	2	0	1308	1803	.0	1378	0	33	45	0	2847.37	3347.05	4.1	17.3	Q
47.50	3	0	1308	1976	.0	1510	0	62	89	0	2684.90	3207.28	4.4	17.3	Q
81.25	4	0	1308	2768	.0	2116	0	106	182	0	2447.47	3023.86	6.1	17.4	Q
131.88	5	0	1315	4082	.0	3102	0	173	389	0	2080.29	2756.11	8.3	17.4	Q
207.81	6	0	1200	5664	.0	4720	0	264	819	0	1598.32	2365.07	10.7	17.4	Q
321.72	7	0	866	6773	.0	7821	0	363	1591	0	1048.93	1763.63	14.0	17.4	PC
500.00	8	0	444	4337	.0	9760	0	442	2364	0	1048.93	1460.97	16.0	17.4	PC

... ETC. FOR OTHER WELLS ...

PLATFORM 1 SUMMARY

TIME		CUM PRODUCTION			CUM INJECTION		PRODUCTION RATES			INJECTION RATES			%	SALE GAS		STATUS		
DAYS	STEP	WATER	OIL	GAS	WATER	GAS	WATER	OIL	GAS	WATER	GAS	GOR	WCUT	RATE	CUM	PR	GI	WI
10	1	0	170	229	8	0	11	17000	22978	833	0	1351	.1	0	0	Q	SI	PC
25	2	0	425	573	20	0	13	17000	22937	811	0	1349	.1	0	0	Q	SI	PC
47	3	0	807	1101	40	0	14	17000	23436	880	0	1378	.1	0	0	Q	SI	PC
81	4	1	1381	2048	72	0	13	17000	28066	950	0	1650	.1	0	0	Q	SI	PC
131	5	1	2241	4034	124	0	14	17000	39238	1027	0	2308	.1	0	0	Q	SI	PC
207	6	3	3532	8405	211	0	18	17000	57555	1143	0	3385	.1	0	0	Q	SI	PC
321	7	17	5464	18733	365	0	122	16954	90668	1349	0	5347	.7	0	0	Q	SI	PC
500	8	75	7573	31174	658	0	324	11832	69784	1647	0	5897	2.7	0	0	PC	SI	PC

PLATFORM 2 SUMMARY

TIME		CUM PRODUCTION			CUM INJECTION		PRODUCTION RATES			INJECTION RATES			%	SALE GAS		STATUS		
DAYS	STEP	WATER	OIL	GAS	WATER	GAS	WATER	OIL	GAS	WATER	GAS	GOR	WCUT	RATE	CUM	PR	GI	WI
10	1	0	140	188	0	0	28	14000	18868	0	0	1347	.2	0	0	Q	SI	SI
25	2	0	350	473	0	0	36	14000	18967	0	0	1354	.3	0	0	Q	SI	SI
47	3	2	665	911	0	0	79	14000	19460	0	0	1390	.6	0	0	Q	SI	SI
81	4	9	1137	1695	0	0	211	14000	23254	0	0	1660	1.5	0	0	Q	SI	SI
131	5	37	1846	3273	0	0	546	14000	31166	0	0	2226	3.8	0	0	Q	SI	SI
207	6	139	2909	6479	0	0	1348	14001	42217	0	0	3015	8.8	0	0	Q	SI	SI
321	7	417	4440	12809	0	0	2435	13438	55572	0	0	4135	15.3	0	0	PC	SI	SI
500	8	834	5993	19794	0	0	2341	8709	39176	0	0	4497	21.2	0	0	PC	SI	SI

SUMMARY OF ALL PLATFORMS

TIME		CUM PRODUCTION			CUM INJECTION		PRODUCTION RATES			INJECTION RATES			%	SALE GAS	
DAYS	STEP	WATER	OIL	GAS	WATER	GAS	WATER	OIL	GAS	WATER	GAS	GOR	WCUT	RATE	CUM
10	1	0	310	418	8	0	39	31000	41846	833	0	1349	.1	0	0
25	2	1	775	1047	20	0	49	31000	41905	811	0	1351	.2	0	0
47	3	3	1472	2012	40	0	93	31000	42896	880	0	1383	.3	0	0
81	4	10	2518	3744	72	0	225	31000	51320	950	0	1655	.7	0	0
131	5	39	4088	7308	124	0	560	31000	70404	1027	0	2271	1.8	0	0
207	6	143	6442	14885	211	0	1367	31001	99772	1143	0	3218	4.2	0	0
321	7	434	9904	31542	365	0	2558	30393	146241	1349	0	4811	7.8	0	0
500	8	909	13566	50968	658	0	2665	20542	108961	1647	0	5304	11.5	0	0

REGION 1 SUMMARY

TIME		CUM PRODUCTION			CUM INJECTION		PRODUCTION RATES			INJECTION RATES				% OIL	% GAS	HC
DAYS	STEP	WATER	OIL	GAS	WATER	GAS	WATER	OIL	GAS	WATER	GAS	GOR	WCUT	REC	REC	PAVG
10	1	0	256	346	0	0	0	25626	34622	0	0	1351	.000	.2	.2	3682
25	2	0	639	865	0	0	0	25569	34590	0	0	1353	.000	.4	.4	3636
47	3	0	1218	1670	0	0	1	25711	35781	0	0	1392	.000	.8	.8	3592
81	4	0	2089	3151	0	0	1	25810	43897	0	0	1701	.000	1.3	1.4	3542
132	5	1	3402	6258	0	0	19	25932	61374	0	0	2367	.001	2.2	2.9	3464
208	6	13	5413	12986	0	0	158	26492	88593	0	0	3344	.006	3.4	6.0	3326
322	7	39	8449	28035	0	0	227	26649	132113	0	0	4957	.008	5.4	12.8	3073
500	8	133	11760	46149	0	0	529	18569	101603	0	0	5471	.028	7.5	21.2	2792

REGION 2 SUMMARY

TIME		CUM PRODUCTION			CUM INJECTION		PRODUCTION RATES			INJECTION RATES				% OIL	% GAS	HC
DAYS	STEP	WATER	OIL	GAS	WATER	GAS	WATER	OIL	GAS	WATER	GAS	GOR	WCUT	REC	REC	PAVG
10	1	0	53	72	8	0	38	5373	7223	833	0	1344	.007	.1	.1	3841
25	2	1	135	181	20	0	48	5430	7314	810	0	1347	.009	.2	.2	3776
47	3	3	254	342	40	0	92	5288	7115	880	0	1345	.017	.4	.4	3720
81	4	10	429	592	72	0	224	5189	7423	950	0	1430	.041	.7	.7	3663
132	5	38	685	1049	124	0	540	5067	9029	1027	0	1782	.096	1.1	1.2	3583
208	6	129	1028	1898	211	0	1208	4509	11178	1143	0	2479	.211	1.7	2.2	3446
322	7	395	1454	3507	364	0	2330	3743	14127	1349	0	3774	.384	2.4	4.2	3185
500	8	776	1806	4819	658	0	2136	1972	7357	1647	0	3729	.520	3.0	5.7	2909

FIELD SUMMARY

TIME		CUM PRODUCTION			CUM INJECTION		PRODUCTION RATES			INJECTION RATES				% OIL	% GAS	HC
DAYS	STEP	WATER	OIL	GAS	WATER	GAS	WATER	OIL	GAS	WATER	GAS	GOR	WCUT	REC	REC	PAVG
10	1	0	310	418	8	0	40	31000	41847	834	0	1350	.001	.1	.1	3727
25	2	1	775	1047	20	0	50	31000	41905	811	0	1352	.002	.4	.3	3675
47	3	3	1472	2012	40	0	94	31000	42897	880	0	1384	.003	.7	.7	3628
81	4	10	2518	3744	72	0	226	31000	51321	950	0	1655	.007	1.2	1.2	3576
132	5	39	4088	7308	124	0	560	31000	70404	1028	0	2271	.018	1.9	2.4	3497
208	6	143	6442	14884	211	0	1367	31001	99772	1143	0	3218	.042	3.0	4.9	3360
322	7	434	9904	31542	364	0	2558	30394	146241	1350	0	4812	.078	4.5	10.4	3104
500	8	909	13566	50968	658	0	2666	20542	108961	1648	0	5304	.115	6.2	16.8	2825

TIMESTEP SUMMARY

*** NOTE: TIMESTEP CUTS ARE CUMULATIVE CUTS

TIME	STEP	ITERATIONS STEP/CUM/CUT	DXMAX(I,J,K)	DSMAX(I,J,K)	DPMAX(I,J,K)	VOLUME MATBAL ERROR	VOLUME MATBAL WATER	% RECOVERY OIL	% RECOVERY GAS	PAVG	QOIL	GOR
10.0	1	3 3 0	.009 15 22 2	.052 19 21 2	-494.8 11 21 2	.0000	1.0000	.1	.1	3810	31000	1350
25.0	2	3 6 0	.011 11 21 2	.044 19 21 2	-190.7 11 21 2	.0000	1.0000	.4	.3	3755	31000	1352
47.5	3	2 8 0	.011 11 21 3	.039 19 21 2	-187.9 11 21 3	.0000	1.0000	.7	.7	3704	31000	1384
81.2	4	3 11 0	.024 1 20 2	.093 1 20 2	-234.2 11 21 2	.0000	1.0000	1.2	1.2	3652	31000	1655
131.9	5	6 17 0	.039 1 10 2	.143 1 10 2	-330.0 13 15 3	.0000	1.0000	1.9	2.4	3579	31000	2271
207.8	6	4 21 0	.042 11 17 3	.129 11 11 2	-547.5 11 17 3	.0000	1.0000	3.0	4.9	3453	31001	3218
321.7	7	6 27 0	.097 15 16 2	.164 10 11 2	-810.9 15 16 2	.0000	1.0000	4.5	10.4	3220	30394	4812
500.0	8	6 33 0	.067 11 3 1	.162 10 13 2	-567.8 2 17 1	.0000	1.0000	6.2	16.8	2955	20542	5304

Table 10 Example of Map Printout

```
   P            TIME =    30.4   DATE:  0  0    0   MAP  WINDOW  1
   ------------------------------------------------------------

   K =  1
   ------

  J  I =   1          2          3          4          5          6          7          8          9         10
 26
 25                                                                                                     8021.4
 24                                                                                8046.9     8026.6     7987.5
 23                                                                      8053.1     8039.5     8024.4     7992.6     7947.6
 22                                                 8057.0     8034.4     8013.6     7994.9     7974.1     7943.8     7899.0
 21                               8040.1     8013.6     7986.1     7957.9     7930.6     7908.9     7895.6     7852.9
 20    8052.6     8033.6     7998.8     7962.6     7929.7     7897.2     7865.3     7843.9     7826.0     7804.0
 19    8028.7     7992.7     7945.9     7902.4     7865.5     7834.2     7807.1     7784.7     7762.6     7737.2
 18    7993.7     7943.9     7889.8     7843.0     7809.4     7776.2     7738.0     7713.5     7693.5     7670.7
 17    7955.2     7896.8     7838.6     7795.3     7736.9     7694.6     7644.1     7638.2     7642.9     7624.7
 16    7924.3     7858.4     7801.2     7735.3     7638.0     7605.7     7611.1     7610.7     7600.4     7591.6
 15               7824.3     7759.0     7688.7     7593.2     7574.0     7580.1     7584.5     7582.7     7575.4
 14                          7712.8     7624.4     7579.0     7512.0*    7562.2     7576.7     7574.3     7567.5
 13                          7702.4     7633.7     7598.8     7568.9     7579.9     7585.1     7581.5     7574.0
 12                                     7648.2     7626.4     7606.4     7599.5     7597.9     7593.5     7585.0
 11                                     7662.1     7657.0     7649.6     7656.7     7657.6     7655.1     7651.2
 10                                                7689.8     7690.1     7699.9     7699.6     7697.3     7694.6
  9                                                7708.9     7710.7     7707.6*    7705.4     7701.8     7697.7
  8                                                           7731.2     7725.4     7719.9     7712.5     7706.0
  7                                                           7745.1     7741.1     7738.1     7729.8     7718.7
  6                                                                                 7749.8     7745.0     7734.2
  5                                                                                            7754.8     7749.7
  4                                                                                                       7758.9
  3
  2
  1


J  I =  11         12         13         14         15         16         17         18         19         20
 26                           8011.3     7995.1     7985.4     7984.5     7994.6     8007.3     8022.2     8037.4
 25    7976.4     7930.2     7893.3     7876.3     7868.1     7870.9     7886.0     7904.3     7920.9     7934.0
 24    7932.8     7878.9     7840.4     7823.2     7820.3     7827.6     7846.2     7868.1     7887.5     7903.5
 23    7878.7     7819.6     7777.6     7766.6     7766.2     7773.5     7795.2     7824.5     7850.0     7868.8
 22    7845.0     7749.7     7709.8     7693.8     7689.4     7699.7     7735.5     7773.2     7805.4     7831.2
 21    7788.0     7683.0     7628.4     7597.8     7602.6     7617.2     7671.8     7709.6     7750.2     7779.6
 20    7719.7     7609.1     7529.0     7483.1     7516.8     7546.4     7583.5     7632.3     7685.5     7723.7
 19    7660.0     7540.9     7517.1     7359.0*    7491.0     7528.5     7563.7     7608.2     7656.0     7702.7
 18    7624.0     7542.2     7516.6     7491.2     7505.9     7525.0     7551.3     7583.9     7625.1     7674.7
 17    7579.9     7541.9     7514.0     7512.1     7518.2     7529.0     7545.6     7564.0     7596.1     7643.7
 16    7572.5     7536.5     7521.2     7522.2     7524.8     7530.6     7543.4     7553.1     7571.7*    7614.6
 15    7559.4     7541.2     7530.0     7527.7     7527.6     7531.7     7539.6     7547.2     7557.5     7586.6
 14    7552.5     7540.3     7535.1     7532.1     7529.4     7530.0     7532.3     7538.2     7524.5     7536.4
 13    7555.6     7529.8#    7538.2     7538.1     7533.9     7528.9     7524.0     7502.2     7439.5     7435.8
 12    7572.5     7553.2     7549.3     7546.7     7540.9     7530.5     7508.5     7474.3*    7280.1     7270.3
 11    7646.5     7632.7     7625.0     7619.6     7607.5     7572.5     7507.5     7312.2     6732.6*    7051.1
 10    7690.8     7689.3     7685.3     7678.0     7666.1     7624.9     7568.5     7466.2     7269.5     7390.7
  9    7695.0     7694.5     7693.5     7689.1     7679.6     7652.5     7608.0     7561.3     7504.0     7573.9
  8    7701.8     7701.4     7701.2     7699.6     7694.7     7681.6     7656.8*    7635.2     7624.5     7688.6
  7    7711.1     7709.3     7709.2*    7707.9     7705.6     7701.9     7694.9     7686.1     7685.5     7731.9
  6    7724.5     7720.2     7718.2     7716.5     7715.0     7714.5     7714.5     7712.4     7716.2     7765.5
  5    7742.5     7734.8     7729.4     7726.8     7726.2     7727.0     7729.0     7729.1     7734.4     7783.3
  4    7755.1     7750.3     7745.6     7742.9     7741.3     7740.4     7740.3     7742.3     7750.6     7799.0
  3               7764.5     7762.2     7760.4     7757.3     7755.0     7754.4     7758.0     7782.9     7816.9
  2                          7773.0     7770.1     7768.9     7771.8     7793.1     7820.0     7850.4
  1                                                           7833.9     7859.4     7878.2     7902.0
```

J	I = 21	22	23	24	25	26
26						
25						
24	7916.2					
23	7885.9	7900.3				
22	7857.0	7873.3				
21	7811.2	7831.9	7858.5			
20	7755.5	7781.8	7820.2	7851.4		
19	7740.6	7765.0	7781.9	7821.6		
18	7722.4	7749.9	7769.9	7790.1		
17	7694.7	7730.6	7758.4	7785.2		
16	7665.6	7706.0	7744.1	7804.4	7838.9	
15	7632.6	7679.7	7745.1	7805.7	7850.1	7911.9
14	7586.6	7666.9	7747.6	7808.7	7859.9	7922.7
13	7530.2	7663.8	7752.3	7816.6	7869.0	7941.4
12	7487.7	7649.9	7762.1	7836.9	7895.1	7980.7
11	7460.9	7644.2	7777.1	7863.0	7926.7	8010.3
10	7553.4	7699.0	7807.4	7886.8	7952.0	8028.9
9	7667.5	7770.1	7846.0	7913.5	7975.1	8040.8
8	7752.5	7818.6	7882.2	7942.9	7998.1	8051.0
7	7789.4	7841.6	7901.8	7968.2	8016.7	8058.8
6	7815.7	7861.9	7911.0	7967.7	8026.5	8064.7
5	7828.7	7873.1	7916.9	7974.7	8036.3	8068.0
4	7838.3	7885.3	7924.4	7983.7	8043.2	8069.3
3	7857.6	7901.8	7940.1	7998.9	8052.0	8069.3
2	7888.7	7920.2	7965.0	8024.6	8063.0	8069.4
1	7926.2	7964.3	8017.7	8061.3	8072.3	8072.0

K = 2

J	I = 1	2	3	4	5	6	7	8	9	10
26										
25										8068.2
24								8088.0	8072.0	8037.7
23						8088.4	8078.7	8069.0	8043.1	8004.5
22				8088.7	8068.6	8051.6	8039.2	8025.4	8002.3	7964.6
21			8072.3	8047.6	8024.0	8001.9	7982.7	7969.2	7963.2	7926.5
20	8083.1	8065.6	8033.2	8000.8	7973.5	7948.9	7926.5	7914.0	7903.1	7884.0
19	8061.2	8027.4	7984.4	7946.2	7916.3	7893.9	7877.3	7863.7	7847.3	7820.7
18	8029.7	7982.7	7933.5	7893.0	7866.9	7842.3	7815.5	7798.2	7783.5	7758.7
17	7994.9	7940.3	7887.9	7851.3	7799.6	7761.0	7725.3	7719.0	7730.1	7720.0
16	7966.9	7906.5	7855.5	7795.8	7705.2	7664.3	7673.4	7676.8	7661.3	7646.7
15		7877.3	7817.1	7749.3	7644.2	7622.1	7632.3	7635.1	7621.8	7609.4
14			7774.7	7681.7	7625.7	7543.0*	7606.7	7612.3	7588.3	7580.2
13			7768.7	7692.4	7652.0	7616.9	7631.0	7631.9	7616.6	7607.0
12				7708.1	7683.6	7661.9	7658.3	7655.9	7649.1	7641.1
11				7721.8	7713.0	7703.6	7706.3	7704.5	7700.0	7695.8
10					7738.4	7735.3	7737.7	7735.2	7730.8	7726.6
9					7753.4	7751.6	7745.8*	7742.5	7738.2	7733.8
8						7768.5	7762.6	7757.3	7750.5	7745.1
7						7780.1	7776.3	7773.9	7767.2	7759.0
6								7784.3	7780.6	7772.9
5									7788.9	7786.1
4										7794.0
3										
2										
1										

This is default map printout:

K = 5

J I =	1	2	3	4	5	6	7	8	9	10
1										
2										
3										
4										
5										
6										
7										
8										
9										
10					7857.0	7852.0	7847.7	7844.9	7838.4	7838.4
11				7868.6	7858.1	7853.1	7849.0	7845.7	7842.3	7841.9
12				7874.5	7861.3	7855.0	7851.1	7847.9	7845.7	7845.3
13			7911.1	7885.1	7868.6	7858.5	7853.7	7850.9	7849.2	7847.5
14			7925.5	7895.0	7882.2	7867.1	7857.8	7854.0	7851.8	7849.0
15		7976.1	7944.0	7910.2	7893.8	7884.7	7870.6	7861.5	7856.6	7853.1
16		7993.4	7961.3	7937.8	7907.8	7896.7	7889.2	7878.2	7869.8	7864.6
17			7983.1	7963.8	7939.9	7917.7	7906.5	7895.8	7885.6	7875.2
18				7985.2	7969.1	7958.0	7940.7	7926.7	7909.4	7897.1
19						7982.3	7973.2	7961.3	7948.5	7941.5
20								8001.7	7986.8	7981.9
21										8016.5
22										
23										
24										
25										
26										

This is map printout with "MAPSLINES 0" entered (suppress blank lines):

K = 5

J I =	1	2	3	4	5	6	7	8	9	10
10					7857.0	7852.0	7847.7	7844.9	7838.4	7838.4
11				7868.6	7858.1	7853.1	7849.0	7845.7	7842.3	7841.9
12				7874.5	7861.3	7855.0	7851.1	7847.9	7845.7	7845.3
13			7911.1	7885.1	7868.6	7858.5	7853.7	7850.9	7849.2	7847.5
14			7925.5	7895.0	7882.2	7867.1	7857.8	7854.0	7851.8	7849.0
15		7976.1	7944.0	7910.2	7893.8	7884.7	7870.6	7861.5	7856.6	7853.1
16		7993.4	7961.3	7937.8	7907.8	7896.7	7889.2	7878.2	7869.8	7864.6
17			7983.1	7963.8	7939.9	7917.7	7906.5	7895.8	7885.6	7875.2
18				7985.2	7969.1	7958.0	7940.7	7926.7	7909.4	7897.1
19						7982.3	7973.2	7961.3	7948.5	7941.5
20								8001.7	7986.8	7981.9
21										8016.5

This is map printout with "MAPSFULL 1" entered:

K = 5

J I =	1	2	3	4	5	6	7	8	9	10
1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
10	0.0	0.0	0.0	0.0	7857.0	7852.0	7847.7	7844.9	7838.4	7838.4
11	0.0	0.0	0.0	7868.6	7858.1	7853.1	7849.0	7845.7	7842.3	7841.9
12	0.0	0.0	0.0	7874.5	7861.3	7855.0	7851.1	7847.9	7845.7	7845.3
13	0.0	0.0	7911.1	7885.1	7868.6	7858.5	7853.7	7850.9	7849.2	7847.5
14	0.0	0.0	7925.5	7895.0	7882.2	7867.1	7857.8	7854.0	7851.8	7849.0
15	0.0	7976.1	7944.0	7910.2	7893.8	7884.7	7870.6	7861.5	7856.6	7853.1
16	0.0	7993.4	7961.3	7937.8	7907.8	7896.7	7889.2	7878.2	7869.8	7864.6
17	0.0	0.0	7983.1	7963.8	7939.9	7917.7	7906.5	7895.8	7885.6	7875.2
18	0.0	0.0	0.0	7985.2	7969.1	7958.0	7940.7	7926.7	7909.4	7897.1
19	0.0	0.0	0.0	0.0	0.0	7982.3	7973.2	7961.3	7948.5	7941.5
20	0.0	0.0	0.0	0.0	0.0	0.0	0.0	8001.7	7986.8	7981.9
21	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	8016.5
22	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
23	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
24	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
25	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
26	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

Table 11 Example of Sum of Perf Rates Unequal to Well Rate

WELL TABLE
TIME STEP FROM 7104.0 TO 7117.5 DAYS

			LOCATION			PRESSURES PSIA			RATES (STB/D & MCF/D)					CUMS (MSTB & MMCF)						
WELL	PLAT	TYPE	I	J	K	THP	BHP	GRID	WATER	OIL	GAS	% WCUT	GOR	WATER	OIL	GAS	STAT	SG	SW
1002	0	PROD	13	7	1	0	500	1723	0	19	727	0.7	36654	3	1406	11405	PC	0.683	0.090
			13	7	2		500	2073	0	19	302	0.4	15687	3	2009	7429		0.479	0.089
			13	7	3		506	2872	0	13	104	1.2	7471	0	168	486		0.589	0.181
			13	7	4		508	3607	0	79	185	0.5	2322	2	453	1044		0.815	0.185
TOTALS									1	118	1348	0.7	11458	10	3956	20517			

Table 12 Welltype Unit and Integer

welltype	integer equivalent
SWAG	-5
RBWATINJ	-4
STBWATINJ	-3
RBGASINJ	-2
MCFINJ	-1
STBOIL	1
MCF	2
MCFWET	3
STBLIQ	4
RBTOT	5
RBOIL	6
RBGAS	7
RBWAT	8
RBLIQ	9
STBWAT	10

Wetgas rate (MCFWET) is mols hydrocarbon/day, expressed as Mcf/day. SWAG rate is qg mcf/d gas + qw stb/d water, additionally defined by FSWAG=qg/(qg+qw).

Table 13 Printout Frequency Integer n

n	action
0	print every time step.
1	print only at times (dates) entered in datafile.
n > 1	print every n th time step and at times (dates) entered in datafile.
-n (n > 1)	print every n th time step and at times (dates) where the word PRINT appears on the TIME (DATE) dataline.
-1	do not print.

Table 14 List of Mapnames

Porosity and all saturations are fractions.

Map request names for invariant gridblock data:

```
CF           formation(rock) pore-volume compressibility, 1/psi
DELX         x-direction length, ft
DELY         y-direction length, ft
DEPTH        depth to gridblock center, ft
H            gross thickness, ft
HNET         net thickness, ft
INITREG      Initialization Region #
POROS        porosity
PVTTYPE      pvt type, integer
KX           kx, md
KY           ky, md
KZ           kz, md
ROCKTYPE     Saturation Table #
COMPACTYPE   compaction table #
SGR          maximum residual gas saturation Sgr, entered as data
CLAND        The constant C of Eqns (1a) or (2a) of Appendix 5
RESERVOIR    reservoir #
REGION       output region
SWC          connate water saturation
SORW         residual oil to water
SORG         residual oil to gas
SGC          critical gas saturation
SWCG         connate water saturation in a gas-water system
KRWRO        water relative perm at residual oil saturation to water
KROCW        oil relative perm to water at connate water saturation
KRGRO        gas relative perm at residual oil saturation to gas
SWINIT       initial water saturation entered by SWINIT array
SRWSRG       Sorw-Sorg
JLEV         reference Leverett J-function
LZ           vertical fracture spacing, ft
PV           initial pore volume, rb
TX           x-direction transmissibility, rb-cp/day-psi
TY           y-               "               "
TZ           z-               "               "
TXY1         diagonal transmissibility,       "
TXY2                  "                       "
PVF          multiplicative factor on PV
TXF          multiplicative factor on TX
TYF                    "             TY
TZF                    "             TZ
TEX          matrix-fracture exchange transmissibility, rb-cp/day-psi
TEXD         matrix-fracture diffusive transmissibility, 1/ft
THVE         ve thickness, ft
```

Map request names for time dependent gridblock data

```
P            pressure, psia (gas phase)
PSAT         saturation pressure
```

```
SAT          results in writing Sw and So for 3-phase saturation display
             with Map2Excel.
SW           water saturation
SO           oil saturation
SG           gas saturation
SGRC         current residual (trapped) gas saturation, Sgrc
SGMAX        historical maximum gas saturation
TENS         gas-oil interfacial tension, dynes/cm
X            oil phase mol fractions
Y            gas phase mol fractions
PVC          current pore volume, rb
TRACER       tracer fractions
HCPV         hydrocarbon pore volume, Mrb (1000'S rb)
PVSOF        pv * (So - Sorw), Mrb
VISO         oil phase viscosity, cp
VISG         gas phase viscosity
SWF          (Sw - Swc)/(1 - Swc)
SWSWC        Sw - Swc
SOSORW       So - Sorw
SOSORG       So - Sorg
PCWO         Pcwo, psi
PCGO         Pcgo, psi
POIL         oil phase pressure, psia
PWAT         water phase pressure, psia
GW           water gradient (density), psi/ft
GO           oil gradient (density), psi/ft
GG           gas gradient (density), psi/ft
KRW          krw, fraction
KRO          kro, fraction
KRG          krg, fraction
POROSC       current porosity, fraction
COMPACT      compaction for grid block, ft
COMPACTSUM   total compaction at i,j for all blocks k=1,Nz, ft
              (stored and printed in array location i,j,k1, where
               k1 is the first k value such that the location i,j,k1
               is an active block)
CSALT        salt concentration

Note: POROSC, COMPACT, and COMPACTSUM are computed only for blocks
      assigned to compaction tables.  The value of COMPACT is
      del(h) = h0*(f0-f) where h0 is initial net thickness and f0
      and f are initial and current block porosities, respectively.
```

Table 15 Black Oil pvt Example 5 Data

This is from test4.dat. The data were generated by Curtis Whitson and colleagues at Pera a/s from an equation of state for a critical-pointfluid. Their units are:

p bars
Bo,Bg m3/sm3
Rs sm3/sm3
rs sm3/sm3
deno kg/m3
deng kg/m3
viscosity cp

The entry

```
UNITS  c1 c2 c3 c4 c5 c6 c7 c8 c9
```

below converts entered units to standard units as follows:

```
p psia      = c1 * p(entered) + c2
Rs scf/stb  = c3 * Rs(entered)
Bo rb/stb   = c4 * Bo(entered)
rs stb/mmcf = c5 * rs(entered)
Bg rb/scf   = c6 * Bg(entered)
viscosity   = c7 * viscosity(entered)
coil 1/psi  = c8 * coil(entered)
cvoil 1/psi = c8 * cvoil(entered)
deno        = c9 * deno(entered)
deng        = c9 * deng(entered)
```

```
 PVTBO
UNITS 14.503768 0. 5.61458 1. 178107.62 .17810762 1. 1. .0624279
  DENSITY  827.40201  .97646  0. 0.
  PRESSURES 17 23  ! nsat ntot
 100 110 125 150 175 200 225 250 275 300 325 350 375 400 425 450
    475 494 525 550 600 700 800

C  SATURATED TABLE
       PSAT      SRS         RS
    100.000  0.00018614   44.767
    110.000  0.00018698   50.241
    125.000  0.00019103   58.68
    150.000  0.00020364   73.376
    175.000  0.00022216   88.941
    200.000  0.00024589  105.512
    225.000  0.00027480  123.275
    250.000  0.00030937  142.458
```

```
     275.000  0.00035049  163.329
     300.000  0.00039955  186.215
     325.000  0.00045847  211.513
     350.000  0.00052993  239.755
     375.000  0.00061774  271.767
     400.000  0.00072773  309.079
     425.000  0.00087070  355.138
     450.000  0.00107514  420.27
     475.000  0.00173976  574.77

C    The Undersaturated Table is entered here as two tables,
C     an oil table with header PSAT  P  BO  VISO, and a gas table,
C     with header PSAT  P  BG  VISG.  It could be entered as a
C     single table with header PSAT P BO VISO BG VISG.

C    UNDERSATURATED TABLE
     PSAT         P          BO         VISO
      100      100.000    1.25944    0.58454
               110.000    1.25500    0.60364
               125.000    1.24869    0.63238
                .....
               450.000    1.16640    1.26881
               475.000    1.16247    1.31769
               494.000    1.15961    1.35474
               525.000    1.15517    1.41500
               550.000    1.15177    1.46339
               600.000    1.14544    1.55959
               700.000    1.13431    1.74922
               800.000    1.12482    1.93464

     110       110.000    1.27731    0.55127
               125.000    1.27048    0.57836
               150.000    1.26001    0.62382
                ...
               450.000    1.18258    1.18421
               475.000    1.17841    1.23113
               494.000    1.17539    1.26673
               525.000    1.17070    1.32467
               550.000    1.16712    1.37126
               600.000    1.16044    1.46398
               700.000    1.14871    1.64722
               800.000    1.13872    1.82692

       125     125.000    1.30430    0.50716
               150.000    1.29259    0.54881
```

	175.000	1.28200	0.59086
	...		
	450.000	1.20745	1.06907
	475.000	1.20292	1.11311
	494.000	1.19964	1.14656
	525.000	1.19456	1.20110
	550.000	1.19068	1.24502
	600.000	1.18346	1.33259
	700.000	1.17080	1.50628
	800.000	1.16005	1.67736
150	150.000	1.35005	0.44560
	175.000	1.33743	0.48197
	200.000	1.32602	0.51879
	...		
	450.000	1.25068	0.90355
	475.000	1.24552	0.94300
	494.000	1.24178	0.97304
	525.000	1.23599	1.02211
	550.000	1.23159	1.06172
	600.000	1.22339	1.14096
	700.000	1.20909	1.29903
	800.000	1.19699	1.45583
175	175.000	1.39725	0.39506
	200.000	1.38375	0.42689
	225.000	1.37153	0.45917
	...		
	450.000	1.29656	0.76501
	475.000	1.29068	0.80017
	494.000	1.28644	0.82699
	525.000	1.27988	0.87091
	550.000	1.27489	0.90644
	600.000	1.26564	0.97773
	700.000	1.24954	1.12077
	800.000	1.23597	1.26363
200	200.000	1.44652	0.35253
	225.000	1.43215	0.38042
	250.000	1.41913	0.40875
	...		
	450.000	1.34567	0.64827
	475.000	1.33900	0.67944
	494.000	1.33419	0.70327
	525.000	1.32676	0.74236

	550.000	1.32113	0.77406
	600.000	1.31070	0.83786
	700.000	1.29262	0.96658
	800.000	1.27744	1.09601
225	225.000	1.49859	0.31603
	250.000	1.48334	0.34046
	275.000	1.46951	0.36532
	...		
	450.000	1.39875	0.54947
	475.000	1.39118	0.57696
	494.000	1.38572	0.59801
	525.000	1.37732	0.63262
	550.000	1.37096	0.66075
	600.000	1.35921	0.71753
	700.000	1.33891	0.83272
	800.000	1.32195	0.94932
250	250.000	1.55426	0.28421
	275.000	1.53810	0.30561
	300.000	1.52341	0.32741
	...		
	450.000	1.45667	0.46567
	475.000	1.44806	0.48977
	494.000	1.44187	0.50827
	525.000	1.43236	0.53874
	550.000	1.42517	0.56356
	600.000	1.41192	0.61381
	700.000	1.38913	0.71630
	800.000	1.37016	0.82075
275	275.000	1.61443	0.25617
	300.000	1.59729	0.27490
	...		
	450.000	1.52043	0.39456
	475.000	1.51063	0.41558
	494.000	1.50359	0.43174
	525.000	1.49280	0.45841
	550.000	1.48466	0.48019
	600.000	1.46971	0.52440
	700.000	1.44408	0.61506
	800.000	1.42285	0.70805
300	300.000	1.68011	0.23126
	325.000	1.66192	0.24761

```
          ...
        450.000    1.59126    0.33428
        475.000    1.58007    0.35250
        494.000    1.57205    0.36654
        525.000    1.55978    0.38975
        550.000    1.55054    0.40874
        600.000    1.53361    0.44741
        700.000    1.50474    0.52711
        800.000    1.48092    0.60941

325     325.000    1.75253    0.20900
        350.000    1.73320    0.22325
          ...
        450.000    1.67066    0.28327
        475.000    1.65783    0.29898
        494.000    1.64865    0.31109
        525.000    1.63465    0.33117
        550.000    1.62413    0.34763
        600.000    1.60491    0.38124
        700.000    1.57227    0.45087
        800.000    1.54549    0.52324

350     350.000    1.83333    0.18895
        375.000    1.81272    0.20133
          ...
        450.000    1.76065    0.24013
        475.000    1.74586    0.25359
        494.000    1.73532    0.26398
        525.000    1.71926    0.28124
        550.000    1.70723    0.29541
        600.000    1.68531    0.32442
        700.000    1.64828    0.38485
        800.000    1.61805    0.44804

 375    375.000    1.92513    0.17066
        400.000    1.90305    0.18138
        425.000    1.88286    0.19234
        450.000    1.86429    0.20355
        475.000    1.84714    0.21499
        494.000    1.83494    0.22384
        525.000    1.81642    0.23856
        550.000    1.80258    0.25067
        600.000    1.77743    0.27552
        700.000    1.73518    0.32753
        800.000    1.70088    0.38225
```

400	400.000	2.03290	0.15355
	425.000	2.00906	0.16274
	450.000	1.98722	0.17216
	475.000	1.96712	0.18178
	494.000	1.95286	0.18923
	525.000	1.93127	0.20165
	550.000	1.91519	0.21188
	600.000	1.88608	0.23293
	700.000	1.83743	0.27718
	800.000	1.79818	0.32402
425	425.000	2.16813	0.13663
	450.000	2.14194	0.14439
	475.000	2.11792	0.15233
	494.000	2.10094	0.15849
	525.000	2.07531	0.16876
	550.000	2.05629	0.17723
	600.000	2.02198	0.19470
	700.000	1.96502	0.23160
	800.000	1.91936	0.27089
450	450.000	2.36578	0.11793
	475.000	2.33575	0.12419
	494.000	2.31461	0.12906
	525.000	2.28283	0.13717
	550.000	2.25934	0.14388
	600.000	2.21718	0.15773
	700.000	2.14774	0.18711
	800.000	2.09255	0.21858
475	475.000	2.86902	0.08885
	494.000	2.83684	0.09203
	525.000	2.78884	0.09731
	550.000	2.75363	0.10168
	600.000	2.69099	0.11070
	700.000	2.58933	0.12990
	800.000	2.50981	0.15060

PSAT	P	BG	VISG
100	100.000	0.014671	0.016678
	110	0.013295	0.017048
	125	0.011661	0.017658
	...		
	450	0.003974	0.035923

```
      475          0.003850   0.037212
      494          0.003765   0.038173
      525          0.003641   0.039716
      550          0.003552   0.040940
      600          0.003398   0.043348
      700          0.003158   0.048089
      800          0.002981   0.052854

110   110.000      0.013299   0.017046
      125          0.011666   0.017654
      150          0.009709   0.018801
             ...
      450          0.003975   0.035889
      475          0.003851   0.037176
      494          0.003766   0.038137
      525          0.003641   0.039677
      550          0.003552   0.040900
      600          0.003398   0.043304
      700          0.003158   0.048039
      800          0.002980   0.052795

125   125.         0.011672   0.017655
      150          0.009714   0.018803
          ...
      450        0.003978   0.035901
      475        0.003854   0.037188
      494        0.003768   0.038149
      525        0.003644   0.039690
      550        0.003555   0.040913
      600        0.003400   0.043319
      700        0.003161   0.048056
      800        0.002983   0.052816

150   150.000    0.009720   0.018831
      175        0.008363   0.020135
         ...
      450         0.003986   0.036055
      475        0.003862   0.037349
      494        0.003777   0.038315
      525        0.003653   0.039865
      550        0.003564   0.041095
      600        0.003409   0.043516
      700        0.003170   0.048287
      800        0.002992   0.053085
```

```
 175  175.000      0.008365   0.020218
      200          0.007381   0.021658
           ...
      450          0.003998   0.036351
      475          0.003874   0.037658
      494          0.003789   0.038634
      525          0.003666   0.040201
      550          0.003577   0.041444
      600          0.003423   0.043894
      700          0.003184   0.048728
      800          0.003007   0.053600

 200 200.000      0.007381   0.021818
     225          0.006644   0.023367
       ...
     450          0.004012   0.036775
     475          0.003889   0.038101
     494          0.003805   0.039092
     525          0.003682   0.040682
     550          0.003594   0.041945
     600          0.003440   0.044436
     700          0.003203   0.049363
     800          0.003026   0.054340

225  225.000      0.006644   0.023631
     250          0.006077   0.025258
        ...
     450           0.004030   0.037330
     475           0.003908   0.038680
     494           0.003824   0.039690
     525           0.003702   0.041311
     550           0.003614   0.042601
     600           0.003462   0.045147
     700           0.003226   0.050195
     800           0.003050   0.055312

250  250.000       0.006078   0.025654
     275           0.005634   0.027332
       ...
     450         0.004051   0.038030
     475         0.003930   0.039411
     494         0.003847   0.040444
     525         0.003726   0.042106
     550         0.003639   0.043428
     600         0.003488   0.046044
```

```
      700         0.003253   0.051248
      800         0.003078   0.056544

 275 275.000      0.005639   0.027897
     300          0.005284   0.029603
        ...
     450         0.004075   0.038901
     475         0.003956   0.040321
     494         0.003874   0.041384
     525         0.003754   0.043095
     550         0.003668   0.044459
     600         0.003518   0.047163
     700         0.003286   0.052566
     800         0.003113   0.058090

300  300.000     0.005294   0.030381
     325         0.005006   0.032102
      ...
     450           0.004104   0.039984
     475           0.003987   0.041452
     494           0.003906   0.042553
     525           0.003788   0.044327
     550           0.003703   0.045745
     600           0.003555   0.048561
     700           0.003326   0.054216
     800           0.003154   0.060029

325  325.000     0.005024   0.033155
     350         0.004789   0.034887
       ...
     450         0.004139   0.041339
     475         0.004024   0.042868
     494         0.003945   0.044016
     525         0.003829   0.045871
     550         0.003745   0.047357
     600         0.003600   0.050318
     700         0.003374   0.056298
     800         0.003204   0.062482

350  350.000      0.004816   0.036301
     375          0.004622   0.038054
     400          0.004455   0.039757
     425          0.004310   0.041421
     450          0.004183   0.043052
     475          0.004070   0.044660
```

	494	0.003993	0.045870
	525	0.003879	0.047830
	550	0.003797	0.049404
	600	0.003655	0.052555
	700	0.003433	0.058961
	800	0.003266	0.065631
375	375.000	0.004664	0.039966
	400	0.004502	0.041764
	425	0.004361	0.043524
	450	0.004238	0.045256
	475	0.004128	0.046968
	494	0.004053	0.048261
	525	0.003942	0.050362
	550	0.003863	0.052055
	600	0.003724	0.055461
	700	0.003506	0.062438
	800	0.003342	0.069757
400	400.000	0.004563	0.044409
	425	0.004428	0.046301
	450	0.004309	0.048170
	475	0.004203	0.050027
	494	0.004130	0.051434
	525	0.004023	0.053732
	550	0.003946	0.055592
	600	0.003811	0.059351
	700	0.003599	0.067121
	800	0.003438	0.075335
425	425.000	0.004519	0.050155
	450	0.004405	0.052228
	475	0.004304	0.054298
	494	0.004234	0.055875
	525	0.004132	0.058462
	550	0.004057	0.060569
	600	0.003927	0.064851
	700	0.003722	0.073783
	800	0.003565	0.083300
450	450.000	0.004550	0.058609
	475	0.004455	0.061041
	494	0.004389	0.062906
	525	0.004292	0.065984
	550	0.004222	0.068504

	600	0.004098	0.073663
	700	0.003901	0.084524
	800	0.003750	0.096180
475	475.000	0.004991	0.088850
	494.000	0.004935	0.092030
	525	0.004852	0.097300
	550	0.004790	0.101660
	600	0.004681	0.11068
	700	0.004504	0.12988
	800	0.004366	0.15058

11 Keyword Index

*11.1 AC - EQUALS KX**

11.2 EQUALS KY - MAXITN

F

G

H

I

J

K

L

M

11.3 MAXLZ/TH - QMINUS

N

O

P

Q

11.4 RADIAL - TRACERF

R

S

T

11.5 TRFCUT - ZVAR

U

V

W

X

Y

Z