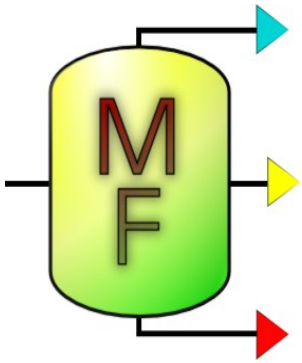
Multiflash Command Reference

KBC Advanced Technologies plc.

Version 6.1
June 2016



KBC Advanced Technologies plc.
42-50 Hersham Road
Walton on Thames
Surrey
KT12 1RZ
UK
T +44 (0) 1932 242 424
F +44 (0) 1932 224 214

e-mail: software@kbcat.com

This manual and the information contained within is the copyright of Infochem / KBC Advanced Technologies plc.

KBC Advanced Technologies plc.
42-50 Hersham Road
Walton on Thames
Surrey
KT12 1RZ
UK
T +44 (0) 1932 242 424
F +44 (0) 1932 224 214

Disclaimer

While every effort has been made to ensure that the information contained in this document is correct and that the software and data to which it relates are free from errors, no guarantee is given or implied as to their correctness or accuracy. Neither Infochem / KBC Advanced Technologies plc nor any of its employees, contractors or agents shall be liable for direct, indirect or consequential losses, damages, costs, expenses, claims or fee of any kind resulting from any deficiency, defect or error in this document, the software or the data.

Contents

Introduction 1

Changes from previous versions 2

Using Multiflash 5

- Entering commands 5
- Specifying a problem 5
- Units 10
- Changing the problem 10
- Error messages and diagnostics 10
- Output 11
- Model configuration files 12
- Example Calculation 12

Databanks and components 15

- Commands for setting component databanks 15
- Loading components from a databank 16
- Data entry for a normal component 17
- Petroleum fractions 29
- PVT analysis 31
- Black oil analysis 36
- Inhibitor calculator 37
- Salinity 38

Model definition 40

- Introduction 40
- Equation of state models 40
- Activity models 51
- Other thermodynamic models for fluids 64
- Thermodynamic models for solids 66
- Viscosity models 71
- Thermal conductivity models 74
- Surface tension models 76

Binary interaction parameters 79

- BIPDATA 79
- BIPSET 80
- Troubleshooting - BIPs 82

Phase descriptors and stream types 83

- The PD command 83
- The streamtype command 84
- The KEYS component command 85

PVT Experiment Data....89
Storing PVT Experiment data....89
Conditions....91
Commands for setting conditions....91
Matching wax data....93
Matching asphaltene deposition point....94
PVT table files....95
Pipesim PVT files....95
OLGA table generator....96
Prosper table generator....96
Other commands....98
The TITLE command....98
Setting Enthalpy and Entropy bases....98
Commands for changing units....99
The SET command....101
The REMOVE command....102
The INCLUDE command....103
Alphabetical list of selected commands....104

Introduction

Multiflash is an advanced software package for performing complex equilibrium calculations quickly and reliably. The main utility is a multiple phase equilibrium algorithm which is interfaced to Infochem / KBC's package of thermodynamic models and a number of physical property data banks.

This document describes the Multiflash command language which can be used to configure and drive the software. Multiflash itself can be used and accessed in many different ways (see below) but the command language is common to all implementations. An overview of the Multiflash software structure is given in the diagram on the following page.

The command-line version of Multiflash is largely machine independent. It can be run by typing in all the necessary instructions at the command line or by reading in files of commands. The software is also available as an interactive Microsoft Windows program (see the *User's Guide for Multiflash for Windows*) and as an add-in for use with the Microsoft Excel spreadsheet program (see the *User's Guide for Multiflash Excel Interface*).

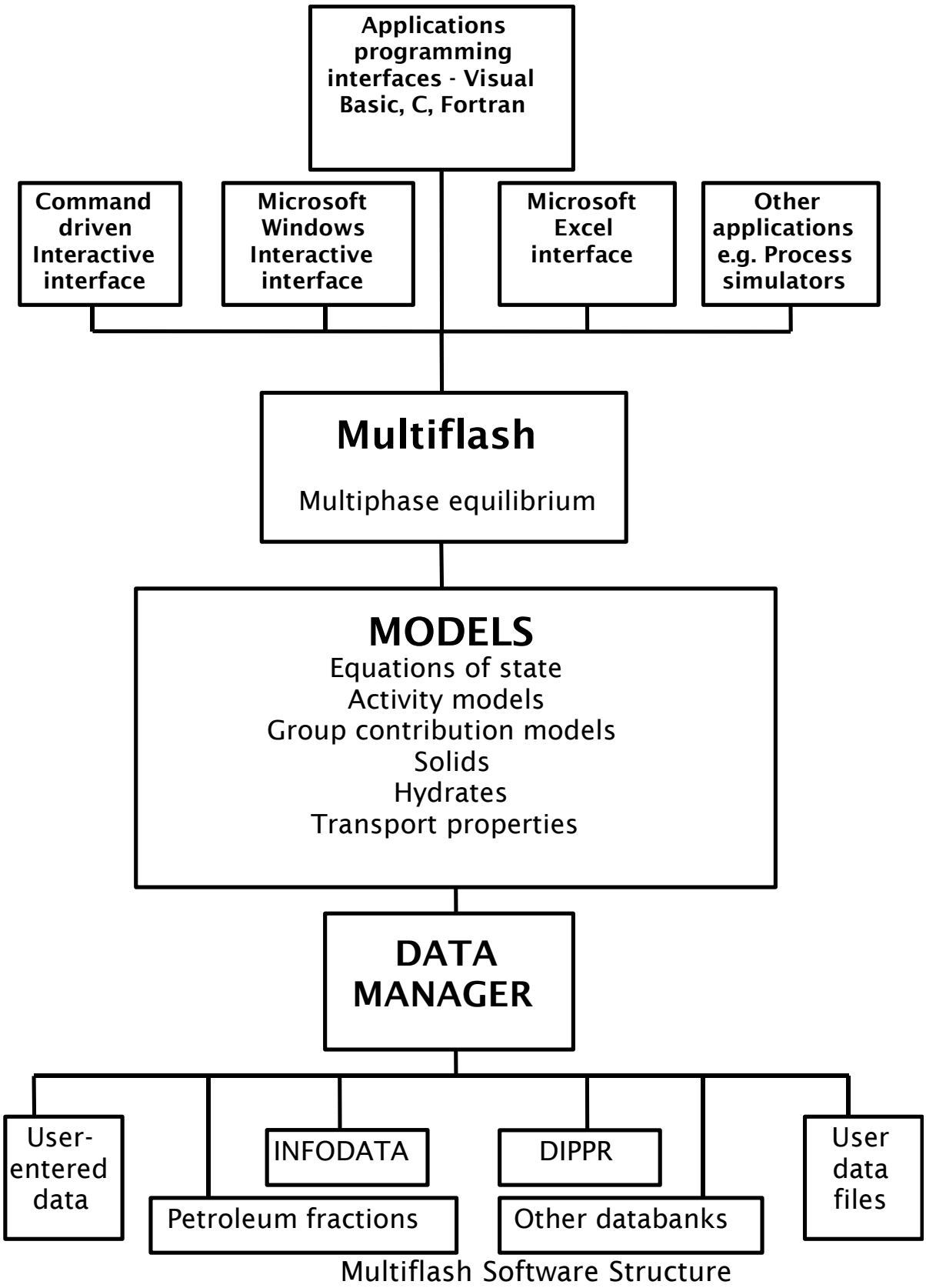
For application developers Multiflash can be used as a set of procedures callable from Microsoft Visual Basic (see the *Programmer's Guide for the Multiflash Visual Basic Interface*), C and C++ (see the *Programmer's Guide for the Multiflash C Language Interface*) and Fortran. The low-level application interface is descried in the *Multiflash Programmer's Guide*.

Changes from previous versions

Since Multiflash 6.0, the command processor, which is used to read MFL files, has been redesigned for efficiency and for flexibility. This had to take into account the new Graphical User Interface (GUI), in which there is no output window to send command outputs to.

As such these are the fundamental changes:

- All the SHOW/LIST/WRITE commands have been disabled.
- HELP command as been disabled as there is no output for messages. The Error messages help file will provide all the necessary information.
- All MATCH commands have been disabled:
  - The new matching facilities available in the PVT Lab tests tool in Multiflash do not make use of commands, hence there are no commands available to drive the model tuning.
- Error messages do not appear in the main window. If any error occurs in the “send commands” tool in the GUI will show an error dialog box.
- Calculation commands have been disabled
  - Calculations are solely driven by the GUI, hence commands like PTF, PFRACF, FRACPHENV, PHENV, FIXEDPHASE, etc, have been disabled.
  - Wax appearance temperature and wax precipitation curve are disabled.
- Delump/blend commands has been disabled. All delumping/blending operations are done without using the command processor.
- PVT tables command have been disabled are only needed to store PVT table information.
- SET commands are disabled. Any calculation modes or property output are set in the GUI.



Multiflash Software Structure

Using Multiflash

This section gives an overview of how the command language can be used to set up and run a problem.

Entering commands

In the Multiflash GUI, the command dialog box all commands may be entered to perform tasks not supported by the GUI. The other interfaces have different ways of accepting commands, *e.g.* in the Excel interface commands can be entered on a spreadsheet. An alphabetical list of commands is given on page 104.

Most commands need to be followed by extra information to complete their action, *e.g.*

```
PUREDATA INFODATA;
```

The `PUREDATA` command is followed by the `INFODATA` keyword to instruct Multiflash to use the infodata data bank to load component properties. Commands may be entered in upper- or lower-case and may be abbreviated to the shortest unique character string.

In all versions commands may be read in from existing input files, the MFL files. Input files can be read in using the `INCLUDE` command, *e.g.*

```
INCLUDE c4c5.mfl;
```

Include files can have any name that is acceptable on your computer system. The sample files supplied with Multiflash use the file extensions `.mfl`.

Specifying a problem

A phase or chemical equilibrium problem is specified using the sequence of operations described below.

Source of pure component data

The data source (databank) for normal components is set using the command `PUREDATA`, followed by the databank name. *E.g.*

```
PUREDATA INFODATA;
```

sets the databank to be the standard Infochem fluids databank. More than one databank may be used to set up a problem but only one can be active at any time. For more details see Commands for setting component databanks on page 13.

Components in a mixture

Normal components are added to the mixture using the command COMPONENTS followed by a list of component names. The maximum number of components is currently 200. If the component name includes punctuation or spaces then it should be put in double quotation marks. Each name must be a valid component name for the databank in force. *E.g.*

```
components methane butane water "carbon dioxide ";
```

An alternative form of the components command allows a component to be defined in a particular place in the sequence of components. For example, to define component 4 without, necessarily, defining components 1 to 3

```
components 5 methanol;
```

When defining components from more than one databank, components are added to the end of the existing list of components unless you specify otherwise, *e.g.*

```
PUREDATA DIPPR;
COMPONENTS methanol hydrogen "carbon monoxide";
PUREDATA Infodata;
COMPONENTS oxygen;
```

makes oxygen the fourth component.

For more details see Commands for setting component databanks on 13. Information on how to define components that are petroleum fractions is given in the section on page 28.

Source of binary interaction parameter data

Binary interaction parameters (BIPs) are required by most of the mixture models in Multiflash. BIP data may be taken from a databank or entered directly on the command line. If BIP data are not defined the models will use their internal default values. The default BIPs may or may not lead to reasonable predictions of mixture properties. The behaviour depends on the model and mixture.

BIP databanks are defined by the command BIPDATA followed by the bank name. *E.g.*

```
BIPDATA infobips oilandgas;
```

The oilandgas bank provides BIP values for components typical of oil and gas mixtures for cubic eos. And the infobips databank provides the interaction parameters for VLE and refrigerants.

The command BIPSET is used to define BIPs directly. This is a somewhat complex and, potentially, error-prone process. It is recommended that the necessary commands are entered in a problem setup file which can then be included, checked and revised as necessary.

For more details see Binary interaction parameters on page 79.

Mixture models

A complete description of the available models and how they are defined is are given in the section starting on p.39. It is necessary to set up models that Multiflash can use to calculate (at least) fugacity coefficients in phase equilibrium calculations. If other properties such

as enthalpy or volume are of interest then the appropriate models must be made available.

A model is specified using the command MODEL, followed by a user-defined model identifier, a Multiflash model keyword and any further model-dependent information. For example, the following command sets up the PR eos with the name MPR

```
MODEL MPR PR PRBIP;
```

The model identifier can be any unique name assigned by the user. It is used subsequently to refer to the model, *e.g.* when defining a phase descriptor (see below).

The MODEL command also allows a set of BIPs defined with the BIPSET command to be associated with a model. For example, the following command defines the Wilson A activity equation as a model called MWILSONMPR using the PR eos (MPR) as the vapour phase model and taking BIPs from a bipset called Wilson1:

```
MODEL MWILSONMPR wilson a MPR Wilson1;
```

Note that the MODEL command is also used to define transport property mixture models. The following command sets up a viscosity model called MLBCMPR (which uses a previously-defined MPR).

```
MODEL MLBCMPR LBC LFIT MPR;
```

Phase descriptors

The idea of a phase descriptor (PD) is central to the operation of Multiflash. The phase descriptor contains all the information required to identify a phase and to retrieve its thermodynamic and transport properties. A PD must be specified for each possible phase that Multiflash is to consider. It is possible that only a subset of the list of possible phases will actually be present at equilibrium. The maximum number of PDs that may be defined is currently 20 and the maximum number of phases that may be present at equilibrium is 7.

Multiflash is designed to allow the use of different thermodynamic models for different phases and for different properties of a given phase. The solution methods used do not make any assumptions about thermodynamic consistency between fugacity coefficients, volumetric properties and thermal properties.

The phase descriptor for each phase is defined using the PD (or phasedescriptor) command (see p. 83). The command sets up a user-defined identifier for the phase; the phase type (*e.g.* liquid), plus a list of model identifiers for fugacities, volume, enthalpy/ entropy, viscosity, thermal conductivity and surface tension. The PD identifier can be any unique name assigned by the user, it is simply used as a label, *e.g.* in the output, and does not have any other significance.

The list of models may be terminated at any point by the end marker ; and any models not specified will be taken as undefined. A * in place of a model identifier denotes ‘use the last named model’ or ‘undefined’ if the last model is unsuitable for that property.

If the volume model is not defined then it is set to the fugacity model, if the enthalpy/entropy model is not defined then it is set to the volume model. If you need to calculate viscosity, thermal conductivity or surface tension then a model *must* be defined for each property required.

For example,

```
PD hc_liquid liquid MPR;
```

defines a PD called hc_liquid. The phase type is liquid and all the thermodynamic properties (fugacities, volume, enthalpy and entropy) will be calculated with the model MPR. No transport property models are defined.

In the following example the thermodynamic properties will be calculated using the model MPR (as no other models are defined for volume or enthalpy/entropy - note the use of *) and, in addition, the viscosity will be calculated using the viscosity model MLBCMPR. The thermal conductivity and surface tension remain undefined.

```
PD hc_liquid liquid MPR * * MLBCMPR;
```

The example below adds a thermal conductivity model MCLSMPR and a surface tension model MMCSMPR

```
PD hc_liquid liquid MPR * * MLBCMPR MCLSMPR MMCSMPR;
```

Key components

For phase equilibrium calculations involving more than one liquid phase the user may wish to define a set of key components which are associated with a phase descriptor. This is done by using the command KEYS, followed by the PD name, followed by the name or the CARN number of the key component,s *e.g.*

```
KEYS liquid2 water methanol;
```

Or

```
KEYS liquid2 007732-18-5 000067-56-1;
```

where liquid2 is a phase descriptor that has previously been defined. For more details see p. 85.

Similarly the keyword "not" can also be used to indicate that a component should not be present (or present in the smallest amount) in a phase. For example, to identify the non-aqueous liquid phase in a hydrocarbon/water/methanol problem the following command could be used:

```
KEYS liquid1 not water not methanol;
```

Or

```
KEYS liquid1 not 007732-18-5 not 000067-56-1;
```

For many calculations it is not *necessary* to define key components. Multiflash will allocate the phase descriptors for multiple liquid phases in the order in which the phases are found. However this means that the same phase may appear in different columns of the output as conditions change, *e.g.* Liquid2 may become Liquid1.

For fixedphase calculations, when the fixed phase is one of the liquid phases, key components must be defined because Multiflash has no other way of distinguishing between multiple liquid phases.

In versions previous to Multiflash 4.4 only one key component could be defined per phase descriptor, this was done by using the command KEY. Although, from MF 4.4 onwards the use of KEY (as opposed to KEYS) is no longer recommended Multiflash still recognises it for compatibility reasons.

From version 4.4 onwards a new keyword can be used to define aqueous phases: AQUEOUS:

```
KEYS water AQUEOUS;
```

This keyword will automatically add the following list o components as aqueous components to the list of key components:

- water
- methanol
- ethanol
- MEG
- DEG
- TEG
- Glycol
- Propylene glycol
- Propylene glycol monomethyl ether
- dipropylene glycol
- dipropylene glycol monomethyl ether

Composition

The compositions (or amounts of each component) are defined by the AMOUNTS command. The default units for amounts are mole numbers but they may be changed using the units and/or inputunits and outputunits commands (see p.98). The AMOUNTS command is followed by a list of values for *all* components, *e.g.*

```
AMOUNTS .3 .3 .4 0.;
```

An alternative form of the command allows the amount of an individual component to be changed or entered. The components for which the amount is to be changed can be identified by name or serial number. Because the serial number is an integer, the amount must contain a decimal point, *e.g.*

```
AMOUNTS methanol 2.0;
```

or (if methanol is the third component)

```
AMOUNTS 3 2.0;
```

For more details see Amounts of components on p. 91.

Calculation conditions

Calculation conditions are defined with the commands TEMPERATURE, PRESSURE, ENTHALPY, ENTROPY, INTERNALENERGY and VOLUME (or DENSITY), followed by the numerical value. The volume and density commands are equivalent - the value is interpreted as a volume or a density depending on the input units set for volume/density. The default units for all quantities are SI, *i.e.* temperature in K, pressure in Pa, energy in J/mol and volume in mol/ m^3.

For example, the following command sets the temperature and enthalpy

```
TEMPERATURE 350 ENTHALPY -28.575;
```

All input conditions must be defined before they are used. Once defined they will remain in force until changed. For more details see Commands for setting conditions on p. 91.

Doing a calculation

All the calculation commands are listed in the section starting on p. 91. In general commands ending with the suffix FLASH are phase equilibrium. A typical example would be PTFLASH for an isothermal flash. Before using a calculation command the amounts and other input quantities must be specified.

Units

The default units for input and output are SI. They may be changed by the commands INPUTUNITS and OUTPUTUNITS respectively. Alternatively the UNITS command sets both input units and output units to the same unit. The commands are followed by the keyword for the property for which the units are to be changed, *e.g.* TEMPERATURE, followed by the unit setting. Where the unit relates to more than one property *e.g.* enthalpy and internal energy, then the units for both will be changed if any one of them is altered. For more details see Commands for changing units on page 99.

The following example sets the input units for temperature to degrees Celsius, the output (display) units for pressure to mm of mercury, and both input and output units for density to kg/m^3:

```
INPUTUNITS TEMPERATURE C;
OUTPUTUNITS PRESSURE mmHg;
UNITS density kg/m3;
```

Changing the problem

Once a problem has been set up any of the specifications may be changed by use of the commands already described. All specifications remain fixed until changed or reset.

The command REMOVE is used to remove (or undefined) related groups of specifications which may be all, bipsets , models, components, pds, streamtypes . Remove all removes all BIP data, models, components, phase descriptors and stream types and resets the units back to the values they had when Multiflash was started, *i.e.* any units set in the mfconfig.dat file are kept. For more details see p. 102.

Error messages and diagnostics

When errors occur an error number will appear followed by a single line description of the error, *e.g.*

```
*** ERROR     210 ***
Loading component from data bank has failed - Diagnostic = component number
DIAGNOSTIC    1
```

For further information HELP followed by the error number will usually provide more information. *e.g.*

```
help 210;

*** ERROR     210 ***
Module: Multiflash command processor
Subprogram: MFCOMP
Error: 210
Multiflash has tried to load the data for a component from the data
files/data bank into the pure-component data system. The load procedure
```

```
has failed.
Diagnostics: (1) Component number
```

The ‘module’ is the section of Multiflash in which the error occurred. The ‘subprogram’ name is intended for use by Infochem technical support. The final line of the above example shows a ‘diagnostics’ message which identifies further information that is printed out after the error message. In this case it shows that component number 1 caused the problem.

The diagnostic level setting (not related to the diagnostics mentioned above) provides a way of getting more information about the progress of a calculation. The diagnostic information is primarily for use by Infochem in tracking down errors and is not intended to be meaningful in general use. The normal level is 0, higher levels, up to 5, give more information, lower levels, down to -2, give less. These are set using the command `SET diagnostics` followed by an integer value for the diagnostic level, *e.g.*

```
SET diagnostics 3;
```

Diagnostics are switched off by `SET nodiagnostics;`

Output

The command-line version of Multiflash produces output on the screen and in the log file (see below). Other interfaces provide output in the most appropriate form, *e.g.* the Visual Basic programming interface can return output in a character string or via function arguments.

Output is produced in response to a calculation command, now only in the **VBA** interface.

Multiflash provides several levels of physical property output, specified by `SET physprops` followed by a code, which is made up of the numbers 0, 1 or 2, optionally followed by the letters A,D,E,F & T (or VCS) depending on what outputs are required.. The zero setting produces minimum output of phases and compositions, 1 adds thermal/volumetric properties and 2 adds derivatives such as Cp and Cv and also the speed of sound. The A adds the activity coefficients of the components in each phase. The D adds the diffusivity. The E code adds thermal properties calculated relative to elements in their standard states (at 298K and 1atm) which are useful for chemical reaction studies. The F adds the fugacity coefficients. The T code adds all transport properties. For transport properties selection use instead V for viscosities, C for thermal conductivity and S for surface tension. In all cases output will only be produced if the relevant models have been defined. For example, to list all thermodynamic and transport properties

```
SET physprops 2VCS;
```

Model configuration files

Problem definition can be viewed as falling into two broad categories, those parts which usually remain fixed and those which can vary frequently. Examples of the former would be the sources of data, the models used and the phase descriptors. On the other hand specifications which may change are the components, the compositions and the calculation conditions.

In order to minimise effort for the user Infochem has set up a series of model configuration files (suffix `.mfl`), which define the more static elements of the problem. Each model configuration file corresponds to a model or group of models with a particular set of model variants. For example `rks.mfl` defines the RKS model. The `.mfl` files are supplied as part of the software distribution.

Example Calculation

The following example sets up the models and phase descriptors for investigating hydrate formation.

```
remove all;
units temperature K pressure Pa enthalpy J/mol
entropy J/mol/K volume m3/mol amounts mol viscosity
Pas thcond W/m/K;
puredata infodata;
bipset RKSABIP3 3 constant eos none ;
bipdata INFOBIPS OILANDGAS4 ;
model MRKSANRTL RKSA PSAT LDEN NRTL RKSABIP3;
model MHYD1MRKSANRTL HYDRATE I MRKSANRTL;
model MHYD2MRKSANRTL HYDRATE II MRKSANRTL;
model MICEMRKSANRTL FREEZEOUT 007732-18-5#WATER#
MRKSANRTL;
pd GAS gas MRKSANRTL;
pd LIQUID1 liquid MRKSANRTL;
pd HYDRATE1 hydrate MHYD1MRKSANRTL;
pd HYDRATE2 hydrate MHYD2MRKSANRTL;
pd ICE condensed MICEMRKSANRTL;
pd WATER liquid MRKSANRTL;
keys WATER AQUEOUS;
keys LIQUID1 *;
```

The components and their amounts are now specified. A series of isothermal flashes is performed at varying temperatures taking a fixed pressure "slice" across the phase diagram. The onset of hydrate formation at a given pressure is predicted as well as the temperature at which a fixed amount of hydrate will form. Calculation of the ice point at the given pressure is shown.

```
components methane butane water;
amounts .49 .49 .02;
pressure 1e5;
# Change T at fixed pressure to cross phase diagram
temperature 250;ptflash;
temp 200; ptf;
temp 220; ptf;
temp 290; ptf;
temp 300; ptf;
# use the fixedphase flash to look for the
# temperature at which hydrate first forms
fixedphase hydrate2 0.0;pfracf;
# repeat the calculation to look for the formation
# of 1% (molar) hydrate
fixedphase hydrate2 0.01;pfracf;
# use the same fixedphase option to look for the
# ice line at 1 bar
fixedphase ice 0.0;pfracf;
```

Databanks and components

Commands for setting component databanks

PUREDATA

The PUREDATA command sets the default databank for pure component properties for components that may be present.

The command has the following format

```
PUREDATA  databank_name;
```

databank_name may be one of the following:

databank_name	Meaning
INFODATA	Infochem fluids databank
DIPPR	DIPPR data compilation of pure compound properties, from AIChE. Requires separate licence. See separate documentation.
ERASE	erases (removes) currently defined databank

Other databanks, typically containing users' own data, in one of the standard formats may be specified as follows:

```
PUREDATA  databank_type  file_name1 file_name2;
```

The databank_type, associated file_name1 and file_name2 are defined in the following table.

Databank_type	required files	Notes
OWNINFODATA	1. databank index 2. databank (.bin) file	files must correspond to the Infodata structure

Loading components from a databank

COMPONENTS

The COMPONENTS command loads normal components that may be present in mixed fluid and solid phases. Each component name is searched for in the currently defined fluid components databank (see PUREDATA command).

The command has the following format:

```
COMPONENTS entry_mode n name_1  name_2  name_3  ….;
```

entry_mode may be either insert, overwrite or amend. If omitted the default mode is overwrite.

n is an integer serial number that defines the position of the first component (name_1) in the component list. If n is omitted it is taken as the last component currently defined. The current version of Multiflash allows up to 200 components to be defined.

Name_1 name_2 etc. must be valid component names for the databank currently defined with the puredata command. If a name has embedded spaces or commas then it should be enclosed in inverted commas, *e.g.* "carbon dioxide" .

In insert mode each new component causes the component to be inserted in the existing list of components at position n (or at the end if n is omitted) without deleting any of them. This is done by incrementing the numbers of existing components as necessary.

In overwrite mode, any existing component of the same component number will be replaced by the newly defined component.

In amend mode, the component is left unaltered except its name and/or physical properties will be overwritten by any new value entered by the user (see below).

For example, the following commands load three components from the DIPPR databank and the fourth component from the Infodata databank.

```
PUREDATA DIPPR;
COMPONENTS methanol hydrogen "carbon monoxide";
PUREDATA Infodata;
COMPONENTS oxygen;
```

IONS

This command is like the COMPONENTS command except that it defines an ion as a component in the fluid mixture, for example:

```
PUREDATA Infodata;
IONS Ca++ Cl-;
```

SEGMENTS

This command is used to define a segment from which a polymer is constituted. It is used for the PC-SAFT model. A segment is not a true component, so it is not normally read from a databank. See below for an example.

Data entry for a normal component

Defining a new component

The usual (and recommended) way of defining pure components is to use the PUREDATA command to define a databank and then to use the COMPONENTS command to select components from that databank. Any number of databanks may be used by repeating this sequence for each databank. The PETROFRACS command (p.29) is provided for defining petroleum fraction pseudocomponents. In addition, the user may define components by entering data directly into Multiflash. This technique can be used to define all the necessary data for a compound or to overwrite selective data items for a component already defined from a databank.

Direct data entry can be quite complex and requires a detailed understanding of the way the data are used by the mixture models and of the correlating equations used to represent pure component properties. It is not recommended that you attempt to define components in this way unless there is no alternative. However, if you *do* wish to proceed, the best method is to prepare the data in the form of an problem setup file which can then be corrected as necessary.

Data items are entered using the COMPONENTS command as follows.

```
COMPONENTS  entry_mode  n  component_name DATA
   property_name1 property_value1
   property_name2 property_value2…. ;;
```

To create a new component entry_mode should be either overwrite or amend (see above).

N is an integer serial number that defines the position of the component in the component list. If n is omitted it is taken as the last component currently defined.

Component_name. Is the user-defined name for the new component. It may be a string up to 72 characters in length. If it contains embedded spaces or commas then it should be enclosed in inverted commas.

The DATA keyword marks the start of data entry for the compound. The end of data for a component is denoted by a semi-colon. A second semi-colon is needed to terminate the COMPONENTS command.

The Property_name must be one of the keywords listed in the sections Pure component constant properties (p.17) and Pure component temperature-dependent properties (p.20). The Property_value is one or more numbers that define a simple constant property such as the critical temperature or the coefficients in a correlation for a temperature-dependent property such as the vapour pressure.

Example

The following is a description of a typical data entry sequence. The command PUREDATA erase cancels any current data source and allows a new, non-databank, component to be defined. All components loaded prior to this remain defined. The new component is then defined using the COMPONENTS command.

For any constant property the data entry is relatively straightforward, *e.g.* Pcrit 40.5e5. The situation is a little more complex for temperature dependent properties. It is necessary to define the property name, equation number and correlation coefficients. The

information given in Pure component temperature-dependent properties (p.20) defines the relevant property names, equation numbers and details of the coefficients required.

For some quantities, *e.g.* for parachor or dipole moment, there are fixed units; for other quantities units are assumed to be as the current input units setting. There is no unit conversion when entering coefficients of temperature dependent pure component properties; these are all in standard SI units.

For example, to add a component:

```
PUREDATA erase;
COMPONENTS 3 naphthalene data
mw 128.175
tboil 491.14
tmelt 353.15
pcrit .4051e07
tcrit 748.4
acentricfactor .303
cpideal 1 4.0 52.0 160 1.80665 -6.0491 17.8647
-15.4058 0 10000
lden 4 616.74 0.25473 748.35 0.27355 333.15 748.35
psat 3 -7.19879 .75005 -2.23858 -3.78919 353.15
748.4; ;
```

Note that two end markers are needed. The first defines the end of the `data` command, the second the end of the `components` command.

Amending data for an existing component

The procedure is similar to setting up a new component except that the some data for the component must already have been entered. This is done either by loading the component from a databank or by defining some data as described above.

Data items are entered or overwritten using the `COMPONENTS` command as follows.

```
COMPONENTS entry_mode n  component_name DATA
   property_name1 property_value1
   property_name2 property_value2…. ;;
```

The command parameters are as described for the case of creating a new component.

Example

To change the critical temperature for a component as it is loaded from a databank;

```
PUREDATA  Infodata;
COMPONENTS hydrogen data tcrit 30;;
```

The following example is equivalent to the first one. Several components are loaded and then the properties of component 1 are changed:

```
PUREDATA  Infodata;
COMPONENTS hydrogen, ethane, methane;
COMPONENTS amend 1 data tcrit 30;;
```

Pure component constant properties

The property units are given in parentheses where applicable. Units for other properties may be set by the `UNITS` or `INPUTUNITS` commands.

Keyword	Property description
CARNUMBER	Chemical abstracts registry number. The format is: 6 digits -2 digits-1 digit. *E.g.* 007440-37-1
FORMULA	Chemical formula (up to 20 character)
FAMILYCODE	Chemical family code (up to 20 character)
UNIFAC	UNIFAC subgroups and abundance
MOLECULARWEIGHT *OR* MW	relative molar mass (molecular weight in g/mol)
TCRIT	critical temperature
PCRIT	critical pressure
VCRIT	critical volume
ACENTRICFACTOR	acentric factor defined as $\omega = -1 - \log_{10} p_r$ at $T_r = 0.7$, where $T_r = T / T_c$ and $p_r = p_{sat} / p_c$
TBOIL	normal boiling point
HFORMATION	standard enthalpy change on formation in the perfect gas state at 298.15K
SSTANDARD	standard entropy in the perfect gas state at 298.15K and 1bar
TMELT	normal melting point
HMELT	enthalpy change on fusion at the melting point
SMELT	entropy change on fusion at the melting point
CPMELT	solid/liquid difference in Cp at the melting point
VMELT	solid/liquid difference in molar volume at the melting point
RUNIQUAC	UNIQUAC r parameter
QUNIQUAC	UNIQUAC q parameter
THLWATER	characteristic temperature for Henry's Law correlation for component in water (K)
VHLWATER	characteristic volume for Henry's Law correlation for solubility of a component in water (m3/mol)
DIPOLEMOMENT	dipole moment (debye)
POLARIZABILITY	Polarizability of component in $Å^3$
QUADRUPOLEMOMENT	Quadrupole moment in Debye Å
PARACHOR	parachor ($(dyne/cm)^{¼}$ cm^3/mol)
RADGYR	radius of gyration (m)
HOCASS	Hayden O'Connell self association parameter
GFORMATION	Gibbs energy of formation in the perfect gas state at 298.15K and 1 bar
TTRIPLE	triple point temperature
PTRIPLE	triple point pressure
HCOMBUSTION	enthalpy of combustion
V25	molar volume at 298.15K for use with solubility parameters
SOLUPAR	liquid phase solubility parameter
SOLIDSOLUPAR	solid solution phase solubility parameter
ZCRIT	critical compressibility factor
REFRACTINDEX	refractive index
TFLASH	flash temperature
TAUTO	autoignition temperature
FLAMLOWER	lower flammability limit (volume % in air)

Keyword	Property description
FLAMUPPER	upper flammability limit (volume % in air)
SPGRAVITY *OR* SG	specific gravity at 60 ºF
EXPANSIVITY	Thermal expansivity of liquid at 1 atm and 60 F (K^{-1})
OMASCALE	Scaling factor to give conventional value of Ωa for cubic EOS
OMBSCALE	Scaling factor to give conventional value of Ωb for cubic EOS
OMARKS	Scaling factor to give conventional value of Ωa specific to RKS model.
OMBRKS	Scaling factor to give conventional value of Ωb specific to RKS model.
OMAPR	Scaling factor to give conventional value of Ωa specific to PR model.
OMAPR	Scaling factor to give conventional value of Ωb specific to PR model.
CNUMBER	Carbon number for petroleum fractions
REFVISCOSITY	reference viscosity for liquid at the boiling point
REFVISST	reference viscosity for liquid at the boiling point for SUPERTRAPP viscosity model
REFVISPD	reference viscosity for liquid at the boiling point for Pedersen viscosity model
REFVISTW	reference viscosity for liquid at the boiling point for Two model
REFVISLB	reference viscosity for liquid at the boiling point for LBC model
LJEVISC	Lennard-Jones σ parameter (m)
LJBVISC	Lennard-Jones ε/k parameter for viscosity (K)
VCLBC	Critical volume for the LBC model (m3)
EOSC	Corresponding states reference equation code
TYPE	The type of components.
HDATUM	Options for calculating enthalpy 1 for compound 2 for elements 3 for standard enthalpy (298.15K)
SDATUM	Options for calculating entropy 1 for compound 2 for elements 3 for standard entropy (298.15K, 1 bar)
COMPREFNO	The component reference number
HYDOC	Hydrate cavity occupation codes
HYD1	Hydrate parameter 1
HYD2	Hydrate parameter 2
HYD3	Hydrate parameter 3
ASSBETA	CPA β parameter
ASSEPSILON	CPA ε_{AB} parameter (J/mol)
ASSGAMMA	CPA γ_{AB} parameter
ASSDELTA	CPA δ_{AB} parameter (K^{-1})
ASSFF	Number of association sites (CPA)
ASSAC	CPA a_c parameter (Jm^3/mol^2)
ASSBC	CPA b parameter (m^2/mol)

Keyword	Property description
ASSKAPPA	CPA κ parameter
SAFTKAPPA	PC-SAFT κ parameter
SAFTEPSILON	PC-SAFT ε_{AB} parameter
SAFTGAMMA	PC-SAFT γ_{AB} parameter
SAFTFF	Number of association sites (PC-SAFT)
SAFTEK	PC-SAFT ε / k parameter
SAFTSIGMA	PC-SAFT σ parameter
SAFTLAMBDA	PC-SAFT λ parameter
SAFTM	PC-SAFT m parameter
MCRKS1	First Mathias Copeman parameter with RKSA model
MCRKS2	Second Mathias Copeman parameter with RKSA model
MCRKS3	Third Mathias Copeman parameter with RKSA model
MCPR1	First Mathias Copeman parameter with PRA model
MCPR2	Second Mathias Copeman parameter with PRA model
MCPR3	Third Mathias Copeman parameter with PRA model
VSRKS1	The first volume shift parameter (m3/mol) with eos RKSA
VSRKS2	The second volume shift parameter (m3/mol K) with eos RKSA
VSRKS3	The third volume shift parameter (m3 K/mol)
VSPR1	The first volume shift parameter (m3/mol) with eos PRA
VSPR2	The second volume shift parameter (m3/mol K) with eos PRA
VSPR3	The third volume shift parameter (m3 K/mol) with eos PRA
CPIDLIQ	Set to 1 to use the pure liquid enthalpy as reference for excess energy models. Set to 2 to also use pure liquid enthalpy as reference for gas phase.
MWEST	Mw parameter used to tune the SuperTRAPP viscosity model
MWEPD	Mw parameter used to tune both variants of Pedersen viscosity model
TBOTW	Boiling point parameter used to tune the Two viscosity model

Pure component temperature-dependent properties

The form of each correlation is defined below. Property units are given in parentheses. They are fixed and are not affected by the UNITS or INPUTUNITS command.

The format require for each property correlation is

```
property_name equation_number coefficients Tmin Tmax;
```

property_name is one of the keywords from the following table.

equation_number identifies the correlation.

coefficients are the numerical values of the correlation coefficients. The number of coefficients required depends on the equation number.

Tmin and Tmax are the minimum and maximum temperature limits for the correlation in K.

Keyword	Equation number	Additional information
CPIDEAL		ideal gas Cp correlation (J/mol K)
	0	data unknown; 0 coefficients
	1	Harmens method, 9 coefficients a3, a4, a5, a6, a7, a8, a9, Tmin, Tmax $C_p / R = a_3 + (a_4 - a_3)y^2(1 + (y-1)F(y))$ where $F(y) = a_6 + a_7 y + a_8 y^2 + a_9 y^3$, $y = \frac{T}{T + a_5}$
	2	DIPPR equation 107, 7 coefficients a3, a4, a5, a6, a7, Tmin, Tmax $C_p = a_3 + a_4\left(\frac{a_5 / T}{\sinh(a_5 / T)}\right)^2 + a_6\left(\frac{a_7 / T}{\cosh(a_7 / T)}\right)^2$
	5	DIPPR equation 100, 7 coefficients a3, a4, a5, a6, a7, Tmin, Tmax $C_p = a_3 + a_4 T + a_5 T^2 + a_6 T^3 + a_7 T^4$

CPLIQUID — liquid Cp correlation (J/mol K)

0 — data unknown, 0 coefficients

1 — DIPPR equation 114, 9 coefficients
a1, a2, a3, a4, a5, a6, a7, Tmin, Tmax

$$C_p = a_1 / \tau + a_2 + a_3\tau + a_4\tau^2 + a_5\tau^3 + a_6\tau^4 + a_7\tau^5$$

where $\tau = 1 - T / T_c$

5 — DIPPR equation 100, 7 coefficients
a1, a2, a3, a4, a5, Tmin, Tmax

$$C_p = a_1 + a_2 T + a_3 T^2 + a_4 T^3 + a_5 T^4$$

CPSOLID — solid Cp correlation (J/mol K)

0 — data unknown, 0 coefficients

5 — DIPPR equation 100, 7 coefficients
a1, a2, a3, a4, a5, Tmin, Tmax

$$C_p = a_1 + a_2 T + a_3 T^2 + a_4 T^3 + a_5 T^4$$

PSAT — saturated vapour pressure (Pa)

0 — data unknown, 0 coefficients

1 — Wagner (form 1) 5 coefficients
a1, a2, a3, Tmin, Tmax

$$\ln p = \ln p_c + \frac{a_1\tau + a_2\tau^2 + a_3\tau^3}{T_r}$$

where $T_r = T / T_c$, $\tau = 1 - T_r$

2 — Antoine equation, 9 coefficients
a1, a2, a3, a4, a5, a6, a7, Tmin, Tmax

$$\ln p = a_1 + \frac{a_2}{T + a_3} + a_4 T^{a_6} + a_5 \ln T + \frac{a_7}{T^2}$$

Note that if the third term is unused a4 should be set to 0 and a6 must be set to a positive value such as 1.0.

3 — Wagner (form 2), 6 coefficients
a1, a2, a3, a4, Tmin, Tmax

$$\ln p = \ln p_c + \frac{a_1\tau + a_2\tau^{\frac{3}{2}} + a_3\tau^3 + a_4\tau^6}{T_r}$$

where $T_r = T / T_c$, $\tau = 1 - T_r$

4 — Wagner (form 3), 6 coefficients
a1, a2, a3, a4, Tmin, Tmax

$$\ln p = \ln p_c + \frac{a_1\tau + a_2\tau^{\frac{3}{2}} + a_3\tau^{\frac{5}{2}} + a_4\tau^5}{T_r}$$

where $T_r = T / T_c$, $\tau = 1 - T_r$

5 — DIPPR equation 100, 7 coefficients
a1, a2, a3, a4, a5, Tmin, Tmax

$$p = a_1 + a_2 T + a_3 T^2 + a_4 T^3 + a_5 T^4$$

HVAP — enthalpy of vaporisation correlation (J/mol)

0 — data unknown, 0 coefficients

1 — Watson/DIPPR 106 equation, 7 coefficients
a1, a2, a3, a4, a5, Tmin, Tmax

$$\Delta H = a_1 \tau^Y$$

where $Y = a_2 + a_3 T_r + a_4 T_r^2 + a_5 T_r^3$, $T_r = T / T_c$,
$T_r = T / T_c$, $\tau = 1 - T_r$

2 — Wagner type equation for enthalpy of vaporisation, 9 coefficients
a1, a2, a3, a4, a5, a6, a7, Tmin, Tmax

$$\Delta H / RT_c = a_1 \tau^{1/3} + a_2 \tau^{2/3} + a_3 \tau + a_4 \tau^{4/3} + a_5 \tau^{5/3} + a_6 \tau^2 + a_7 \tau^6$$

where $\tau = 1 - T / T_c$

5 — DIPPR equation 100, 7 coefficients
a1, a2, a3, a4, a5, Tmin, Tmax

$$\Delta H = a_1 + a_2 T + a_3 T^2 + a_4 T^3 + a_5 T^4$$

LDENS — saturated liquid density correlation (mol/m3)

0 — data unknown, 0 coefficients

1 — Infochem equation, 5 coefficients
a1, a2, a3, Tmin, Tmax

$$\rho = a_1 + a_2 \tau^{a_3}$$

where $\tau = 1 - T / T_c$

2 — Hankinson and Thompson equation (modified), 6 coefficients
a1, a2, a3, a4, Tmin, Tmax

$$\rho = 1 / V_c + a_1 \tau^{\frac{1}{3}} + a_2 \tau^{\frac{2}{3}} + a_3 \tau + a_4 \tau^{\frac{4}{3}}$$

where $\tau = 1 - T / T_c$

Note that in the above equation ρ is the *mass* density and Vc is the critical volume *per unit mass*

3 — Rackett equation (modified), 5 coefficients
a1, a2, a3, Tmin, Tmax

$$1 / \rho = a_1 \left(a_2 + a_3 \tau\right)^Y$$

where $Y = 1 + \tau^{\frac{2}{7}}$, $\tau = 1 - T / T_c$

4 — DIPPR equation 105, 6 coefficients
a1, a2, a3, a4, Tmin, Tmax

$$1 / \rho = a_2^{\,Y} / a_1$$

where $Y = 1 + \left(1 - T / a_3\right)^{a_4}$

5 — DIPPR equation 100, 7 coefficients
a1, a2, a3, a4, a5, Tmin, Tmax

$$\rho = a_1 + a_2 T + a_3 T^2 + a_4 T^3 + a_5 T^4$$

SDENS — solid density correlation (mol/m3)

0 — data unknown, 0 coefficients

5 — DIPPR equation 100, 7 coefficients
a1, a2, a3, a4, a5, Tmin, Tmax

$$\rho = a_1 + a_2 T + a_3 T^2 + a_4 T^3 + a_5 T^4$$

LTHCOND — liquid thermal conductivity correlation (W/m K)

0 — data unknown, 0 coefficients

1 — Jamieson equation, 6 coefficients
a1, a2, a3, a4, Tmin, Tmax

$$\lambda = a_1\left(1 + a_2\tau^{\frac{1}{3}} + a_3\tau^{\frac{2}{3}} + a_4\tau\right)$$

where $\tau = 1 - T / T_c$

2 — DIPPR equation 101, 7 coefficients
a1, a2, a3, a4, a5, Tmin, Tmax

$$\ln\eta = a_1 + a_2 / T + a_3 \ln T + a_4 T^{a_5}$$

5 — DIPPR equation 100, 7 coefficients
a1, a2, a3, a4, a5, Tmin, Tmax

$$\lambda = a_1 + a_2 T + a_3 T^2 + a_4 T^3 + a_5 T^4$$

VTHCOND — vapour thermal conductivity correlation (W/m K)

0 — data unknown, 0 coefficients

1 — DIPPR equation 102, 6 coefficients
a1, a2, a3, a4, Tmin, Tmax

$$\lambda = \frac{a_1 T^{a_2}}{1 + a_3 / T + a_4 / T^2}$$

2 — Reduced correlation, 6 coefficients
a1, a2, a3, a4, Tmin, Tmax

$$\lambda = \frac{\sqrt{T_r}}{a_1 + a_2 T_r + a_3 T_r^2 + a_4 T_r^3}$$

where $T_r = T / T_c$

3 — Monatomic ideal gas, 2 coefficients, Tmin, Tmax

$$\lambda = 3750\eta_0 R / M$$

5 — DIPPR equation 100, 7 coefficients
a1, a2, a3, a4, a5, Tmin, Tmax

$$\lambda = a_1 + a_2 T + a_3 T^2 + a_4 T^3 + a_5 T^4$$

LVISC — liquid viscosity correlation (Pa s)

0 — data unknown, 0 coefficients

1 — Reid, Prausnitz and Poling equation 1,
4 coefficients a1, a2, Tmin, Tmax

$$\eta = a_1 T^{a_2}$$

2 — DIPPR equation 101, 7 coefficients
a1, a2, a3, a4, a5, Tmin, Tmax

$$\ln\eta = a_1 + a_2 / T + a_3 \ln T + a_4 T^{a_5}$$

3 — Reduced correlation, 7 coefficients
a1, a2, a3, a4, a5, Tmin, Tmax

$$\ln\left(\eta / a_5\right) = a_1 X^{\frac{1}{3}} + a_2 X^{\frac{4}{3}}$$

where $X = \frac{a_3 - a_4}{T - a_4} - 1$

4 — Reid, Prausnitz and Poling equation 2/3, 6
coefficients a1, a2, a3, a4, Tmin, Tmax

$$\ln\eta = a_1 + a_2 / T + a_3 T + a_4 T^2$$

5 DIPPR equation 100, 7 coefficients
a1, a2, a3, a4, a5, Tmin, Tmax

$$\eta = a_1 + a_2 T + a_3 T^2 + a_4 T^3 + a_5 T^4$$

VVISC vapour viscosity correlation (Pa s)

0 data unknown, 0 coefficients

1 DIPPR equation 102, 6 coefficients
a1, a2, a3, a4, Tmin, Tmax

$$\eta = \frac{a_1 T^{a_2}}{1 + a_3 / T + a_4 / T^2}$$

2 Reichenberg equation, 5 coefficients
a1, a2, a3, Tmin, Tmax

$$\eta = \frac{a_1 T_r}{\left(1 + a_2 T_r^{a_3}\left(T_r - 1\right)\right)^{\frac{1}{6}}}$$

where $T_r = T / T_c$

3 Chapman-Enskog equation, 5 coefficients
a1, a2, a3, Tmin, Tmax

$$\eta = \frac{26.69 \times 10^{-7}\left(MT\right)^{\frac{1}{2}}}{a_1^2 \Omega^{2,2}\left(T^*, a_3\right)}$$

where

$$\Omega^{2,2} = AT^{*-B} + C\exp\left(-DT^*\right) + E\exp\left(-FT^*\right) + 0.2a_3^2 / T^*$$

$T^* = T / a_2$, $A = 1.16145$, $B = 0.14874$, $C = 0.52487$,
$D = 0.77320$, $E = 2.16178$, $F = 2.43787$

5 DIPPR equation 100, 7 coefficients
a1, a2, a3, a4, a5, Tmin, Tmax

$$\eta = a_1 + a_2 T + a_3 T^2 + a_4 T^3 + a_5 T^4$$

STENSION surface tension correlation (N/m)

0 data unknown, 0 coefficients

1 DIPPR equation 106, 7 coefficients
a1, a2, a3, a4, a5 Tmin, Tmax

$$\sigma = a_1\left(1 - T_r\right)^Y$$

where $Y = a_2 + a_3 T_r + a_4 T_r^2 + a_5 T_r^3$, $T_r = T / T_c$

2 Extended Sprow and Prausnitz equation, 5
coefficients a1, a2, a3, Tmin, Tmax

$$\sigma = a_1 \tau^{a_2}\left(1 + a_3 \tau\right)$$

where $\tau = 1 - T / T_c$

5 DIPPR equation 100, 7 coefficients
a1, a2, a3, a4, a5, Tmin, Tmax

$$\sigma = a_1 + a_2 T + a_3 T^2 + a_4 T^3 + a_5 T^4$$

VIRIALCOEFF second virial coefficient correlation (m3/mol)

0 data unknown, 0 coefficients

1 DIPPR equation 104, 7 coefficients
a1, a2, a3, a4, a5 Tmin, Tmax

$$B = a_1 + a_2 \tau + a_3 \tau^3 + a_4 \tau^8 + a_5 \tau^9$$

where $\tau = T_c / T$

5 DIPPR equation 100, 7 coefficients
a1, a2, a3, a4, a5, Tmin, Tmax

$$B = a_1 + a_2 T + a_3 T^2 + a_4 T^3 + a_5 T^4$$

DIELECTRIC		relative permittivity/dielectric constant correlation
	0	data unknown, 0 coefficients
	1	Maryott-Smith equation 1, 6 coefficients a1, a2, a3, a4, Tmin, Tmax $\varepsilon = a_1 + a_2 T + a_3 T^2 + a_4 T^3$
	2	Maryott-Smith equation 2, 4 coefficients a1, a2, Tmin, Tmax $\varepsilon = \exp(a_1 + a_2 T)$
	3	Infochem equation, 6 coefficients a1, a2, a3, a4, Tmin, Tmax $\varepsilon = 1 + a_1 \exp(-a_2 T - a_3 T^2 - a_4 T^3)$

Pure component Unifac subgroup structure

The specification of Unifac subgroup structure is more complex as it requires a sequence of subgroup identifiers followed by molecular abundances. The abundances are normally integers, but Multiflash can accept non-integer values. Multiflash allows up to 10 different subgroups to be specified in any one component. Multiflash uses a set of generic subgroups that are used to define subgroups structures for all variants of Unifac; details are given in the section of the Unifac model. If the user refers to a subgroup by a name that is not recognised by Multiflash, it is taken to be a new user-defined subgroup. Multiflash allows up to 10 user-defined subgroups to be defined at any one time.

Example

```
COMPONENTS “1,3-butanediol” data unifac CH3 1 CH2 2
CH 1 OH 1 s-OH 1;;
```

Defining polymers using segments

It is useful to define polymers for the PC-SAFT model, especially copolymers, from polymer segments. So initially a number of segments are defined, for example for monomeric groups of the polyethylene and polypropylene polymers:

```
SEGMENTS SegPE data saftsigma 4.0217e-10 saftek 252.0
mw 38.023; SegPP data saftsigma 4.1000e-10 saftek
217.0 mw 43.384;
```

Each segment must be given values for its SAFTSIGMA and SAFTEK parameters as well as its molecular weight. A copolymer of polyethylene and polypropylene can now be defined using the SAFTBOND property and the molecular weight.

```
COMPONENTS PEP47 data saftbond SegPE 0.125 SegPP
0.875 0 mw 236000 tcrit 10000 pcrit 10000
acentricfactor 2.5;
```

The SAFT bond fractions are the lower left triangle of a matrix defining the bond fractions in the polymer chain between the named segments. The critical temperature, critical pressure and acentric factor are only used to provide starting estimates for phase equilibrium calculations, and have no impact on the converged results given by the PC-SAFT model. Note that in the fluid composition, the

amounts of the segments must all be set to zero, as the segments are not real components of the mixture.

For a discussion of the copolymer method, see *Modeling of polymer phase equilibria using Perturbed-Chain SAFT* by Tumakaka, Gross and Sadowski in Fluid Phase Equilibria, **194-197**, 541, (2002). The definition of the bond fractions is given by Tumakaka, Gross and Sadowski, (although their examples of actual values of bond fractions are not realistic).

Minimum data requirements

The minimum data required to perform any phase equilibrium calculation with Multiflash using the basic equations of state is the critical temperature, critical pressure and acentric factor for each component. The data requirements are model dependent and models other than the equations of state require additional data. Isenthalpic and isentropic calculations also require coefficients for the ideal-gas Cp values for each component. To work in mass units rather than in molar units the molecular weight is required.

Chemical equilibrium calculations using the basic equations of state require the critical temperature, critical pressure, acentric factor, Cp values, enthalpy of formation, standard entropy and chemical formula for each component. Again, other models require additional data.

Petroleum fractions

Petroleum fractions or pseudo components are commonly used in modelling oil and gas processing operations. A petroleum fraction may be used to represent the aggregate properties of complex mixtures that are split into fractions of roughly constant boiling point by gas chromatography or standard tests such as ASTM distillation.

Characterisation methods

The CHARDATA command sets the characterisation methods that will be used when defining petroleum fractions (PETROFRAC and PVTANALYSIS commands). In this context the *characterisation method* refers to the set of correlations that are used to estimate the properties of a petroleum fraction based on the information supplied, *e.g.* molecular weight and specific gravity. The correlations are used to estimate all the properties that are normally required for equations of state and basic transport property models. These properties include: molecular weight, boiling point, critical temperature, critical pressure, critical volume, parachor, dipole moment, solubility parameter, enthalpy of formation, standard entropy, perfect gas Cp, vapour pressure, enthalpy of vaporisation, saturated liquid density, saturated liquid viscosity, saturated liquid Cp and saturated surface tension. Any property values supplied in the PETROFRAC command are used in preference to estimated values.

The CHARDATA command has the following format:

```
CHARDATA method_id TB_variant ;
```

The possible values for command parameters are defined in the following table

parameter	Value	comments
method_id	INFOCHAR	The default set of correlations recommended by Infochem. TB/MW/SG are related by the Soereide correlation (see below) and critical properties are estimated by the Lee-Kesler correlations: Kesler, M.G., and Lee, B.I., *Improve predictions of enthalpy of fractions*, Hydrocarbon Proc., **55**(3), 153 (1976)
TB_variant	TBSOEREIDE	The default variant for infochar. Specifies the Soereide correlation form TB in terms of MW and SG. See: Brule, M.R. and Whitson, C.H., *SPE Phase behaviour monograph*, (1996). Only valid for infochar.
	TBAPI	Specifies the API Technical Data Book Procedure 2B2.2 procedure for MW in terms of TB. and SG. Only valid for infochar.

Defining petroleum fractions

To define a petroleum fraction the user must supply some basic information using the PETROFRACS command. It has the following format:

```
PETROFRACS entry_mode component_no component_name
petroleum_type DATA property_name1 value1
property_name2 value2  …  ;;
```

entry_mode may be either insert, overwrite or amend. If omitted the default mode is insert (see description of COMPONENTS command).
component_no is an integer that determines where the petroleum fraction is placed in the list of components used by Multiflash. It may be omitted in which case it is taken to be the last component in the current list.
component_name is a user-defined name for the petroleum fraction.
Petroleum_type is a keyword for marking whether the petroleum fraction is normal fraction or asphaltene or resin or normal paraffin or iso-paraffin. This keyword is optional and may be omitted. If it is omitted, the default setting is normal fraction. It becomes important when asphaltene association model is used. For the Coutinho wax model, the normal paraffins are required in the calculations.
The DATA keyword sets the information on the physical properties of the petroleum fraction. It must be followed by a series of property_name keywords and the corresponding property values. The most useful property keywords are listed below.

Keyword	Detailed information
MOLECULARWEIGHT *OR* MW	molecular weight (g/mol)
SPGRAVITY *OR* SG	SG60/60, i.e. specific gravity at 60°F relative to water at 60°F
TBOIL	Normal boiling point
TCRIT	Critical temperature
PCRIT	Critical pressure
ACENTRICFACTOR	Pitzer acentric factor
VISCOSITY	Viscosity at true boiling point

It is not necessary to supply all of the properties listed. The more data provided the better, but the minimum input sets are a carbon number or any two of molecular weight, specific gravity and boiling point or all three of: critical temperature, critical pressure and acentric factor. The specific gravity, molecular weight and boiling point are usually available for heavy fractions. The alternative of defining critical properties and the acentric factor allows direct transfer of pseudocomponents from a process simulator into Multiflash. Although it is not usual, *any* of the constant or temperature-dependent properties stored in the INFODATA databank could be specified in the data list.

The units of the boiling point, critical temperature and critical pressure are set by the UNITS or INPUTUNITS commands.

The command sequence must be terminated by two end markers. The first ends the DATA keyword and the second ends the PETROFRACS command.

Example

The following example defines a petroleum fraction called C7PLUS as Multiflash component number 3.

```
PETROFRACS 3 C7PLUS
   data  mw 329  sg .881  tboil 648;;
```

PVT analysis

Experimental PVT analysis data can be directly used to specify an input stream for Multiflash using the PVTANALYSIS command. The PVT analysis typically consists of compositions for identifiable discrete components and for single carbon number fractions (SCN) or petroleum cuts. The compositions may be for a complete reservoir fluid or may be separated into gas compositions and liquid compositions. In the latter case it is necessary to supply a separator gas-oil ratio (GOR) so that the streams may be recombined.

The available PVT characterisation method in Multiflash is called Infoanal2. Any the information in a MFL file created by the original method will be loaded properly, you should re-characterise the fluid with the revised method.

The command format for the fluid with n-Paraffin distribution is in the following order.

```
PVTANALYSIS
   Infoanal2
   mw        type_id      x_1
   sg        type_id      x_1 APISG
   components      name_x_1  name_x_2  name_x_3...
   C6, C7, ... & Plusfraction;
   totamounts      x_1  x_2  x_3 ...;
   cutlbound       x_1  x_2 x_3 ...;
   cutlbunits      cut_unit
   cutmw           x_1  x_2 x_3 ...;
   cutsg           x_1  x_2 x_3 ...;
   D86
```

```
stoamounts/nfractions  x_1  x_2  x_3 ...;
gasamounts      x_1  x_2  x_3 ...;
units           amountunit (fluid/liquid)
npunits         amountunit (n-paraffins)
gasunits        amountunit
nplus           n
npnplus         n
startsplit      n or name (e.g. C6 )
npstartsplit    n or name (e.g. N6 )
pluslbound      x_1 x_2 x_3 ...;
nppluslbound    x_1 x_2 x_3 ...;
sara            wpsat wparo wpres wpasp
estimatesara
gorvalue        x_1
gorunits        gor_unit
wax             x_1
estimatewax
watercut        x_1
includewater
total           x_1;
```

Infoanal2 is the keyword for specifying the revised PVT characterisation method. This keyword is optional.

mw - allowed values of type_id are: fluid, sto or fraction. This should be followed by the appropriate value x_1 of the molecular weight in g/mol.

Type_ID	Value of MW (x_1)
fluid	the value for the whole fluid
sto	the value of total stock tank oil
fraction	the value for the heaviest cut

The mw keyword is optional. If not entered the value is estimated from the specific gravity or from the hydrocarbon distribution.

sg - allowed values of type_id are: fluid , sto or fraction. This should be followed by the appropriate value x_1 of the specific gravity (SG).

Type_ID	Value of SG (x_1)
fluid	the value for the whole fluid
sto	the value of total stock tank oil
fraction	the value for the heaviest cut

The sg keyword is optional. If not entered the value is estimated from the molecular weight or from the hydrocarbon distribution.

If the value of SG is to be specified in API degrees, the keyword APISG should be added.

components - followed by the discrete component names name_x_1, name_x_2, name_x_3, etc, and the range of the petrofractions ending with a ; terminator. The range of the petrofractions must be specified to be the starting single carbon number of the petrofractions (e.g. C6), followed by & and then the plusfraction (the last petrofraction). The discrete components which can be handled by PVT analysis are listed below. If you have some components which are not listed here, you can always add them to your final component list from a databank after PVT analysis is done. Please note that the keyword components must be used before any of the amounts keywords given below.

1	Nitrogen	11	n-pentane
2	Hydrogen sulphide	12	Methylcyclopentane
3	carbon dioxide	13	Benzene
4	Methane	14	Cyclohexane
5	Ethane	15	Methylcyclohexane
6	Propane	16	Toluene
7	Isobutane	17	Ethylbenzene
8	n-butane	18	m-xylene
9	Isopentane	19	o-xylene
10	Neopentane	20	p-xylene

Amounts/totamounts - followed by the amounts x_1, x_2, x_3 etc. of the discrete components and the amounts of all the cuts in the total fluid or the separator liquid stream, ending with the amount of the plusfraction and a ; terminator.

cutlbound – followed by the amounts x_1, x_2, x_3 etc. of the user-defined lower carbon-number boundaries for the petroleum cuts. This keyword is only available with Infochem revised characterisation method. This keyword is optional.

cutlbunit – units used to define the carbon-number boundaries after keyword cutlbound. The boundaries can be defined as a true boiling point equivalent to the carbon number, hence the permitted units are either keyword cnumber or a temperature unit. This keyword is only available with Infochem revised characterisation method. This keyword is only required if keyword cutlbound is used.

cutmw – followed by the amounts x_1, x_2, x_3 etc. of the user-defined molecular weights for the petroleum cuts. This keyword is only available with Infochem revised characterisation method. This keyword is optional.

cutsg – followed by the amounts x_1, x_2, x_3 etc. of the user-defined specific gravities for the petroleum cuts. This keyword is only available with Infochem revised characterisation method. This keyword is optional.

D86 – indicates that the characterisation has been measured by the ASTM D86 method. This keyword is optional.

stoamounts - followed by the amounts x_1, x_2, x_3 etc. of the mass or mole percent of all the n-paraffin in STO and a ; terminator. Please note that all the amounts for discrete components have to go to the total fluid/liquid stream.

nfractions - followed by the fractions x_1, x_2, x_3 etc. of all the n-paraffin in each cut and a ; terminator. Please note that all the amounts for discrete components have to go to the total fluid/liquid stream.

Only one of the keywords namounts and nfractions is allowed to be used to specify the information of the n-paraffins.

gasamounts - followed by the amounts x_1, x_2, x_3 etc. of the discrete components and the amounts of all the cuts in the gas stream ending with a ; terminator. The number of amount values must not exceed the number of the disrcete and the liquid cuts specified. This keyword is optional.

If a discrete component is present without knowing its value, its value should be entered as a star * to denote undefined in these cases amounts, totamounts, namounts, nfractions and gasamounts. n-Paraffin distribution always starts from N6. If some of the n-Paraffins are not known, their values should be entered as a star * to denote undefined.

units - specifies the type of input units for the liquid discrete components, petroleum cuts and total amounts. This keyword can only be used after the keywords amounts/totamounts, namounts/nfractions and gasamounts. Allowed values for amountunit are: mol or mole (default), kmol, kg, g, lbmol, lb. This keyword is optional.

npunits - specifies the amount units in which the amount values are given by namounts or nfractions. Allowed values for amountunit are: mole or g. If this keyword is missing, it takes the default value from units. This keyword is optional.

gasunits - specifies the amount units in which the amount values are given by gasamounts. Allowed values for amountunit are: mol or mole (default), kmol, kg, g, lbmol, lb. This keyword is optional.

nplus - the integer number of pseudocomponents that the plusfraction will be split into. The default value is 1. The nplus value generated by the PVTANALYSIS command will be greater than the input value if asphaltenes and resins are present and specified by SARA. This keyword is optional.

npnplus - the integer number of normal paraffin pseudocomponents that will be created. This value is required for characterising the normal paraffin distribution which is associated with the coutinho wax model; the default value is 15 if the keyword is omitted.

startsplit - the cut sequence number or name (e.g. C20 etc.) at which the split is to start which can only be in the range 1 to (number of cuts + 1); the default is the latter value if this keyword is omitted. If the number of normal and non-normal cuts are different, then the smaller value will be used to calculate the default value of startsplit. This keyword is optional.

npstartsplit - the normal paraffin cut sequence number or name (e.g. N20 etc.) at which the normal paraffin split is to start; the default is the value used for N6 if this keyword is omitted.

pluslbound – followed by the amounts x_1, x_2, x_3 etc. of the user-defined lower carbon-number boundaries for the pseudocomponents. This keyword is optional.

nppluslbound – followed by the amounts x_1, x_2, x_3 etc. of the user-defined lower carbon-number boundaries for the normal paraffin pseudocomponents. This keyword is optional.

sara - indicates the presence of a saturates/aromatics/resins/ asphaltenes (SARA) analysis and must be followed by four values: the wt% of saturates, the wt% of aromatics, the wt% of resins and the wt% of asphaltenes. If the values are unknown, the symbol * should be specified. This keyword is optional.

estimatesara – indicates that the SARA analysis will be estimated by Multiflash. This keyword is optional.

gorvalue - specifies the gas-oil ratio x_1 at standard conditions (1 bar and 15ºC) as a pure volume ratio. The value must be in the units set by the gorunits keyword. The value is used to recombine the gas stream amounts with the liquid amounts in the correct proportion to obtain the composition of the recombined fluid. This keyword is optional.

gorunits - specifies the units in which the gorvalue is given.. Allowed values for gor_unit are: m3/m3 (default), or scf/stb . The unit conversion factor from scf/stb to m3/m3 is 0.1801175. This keyword is optional.

Wax – specifies the wax content (UOP) value which is used by Multiflash to estimate the normal paraffin distribution in association with the Coutinho wax model. This keyword is optional.

estimatewax – indicates that the wax content will be estimated by Multiflash. This keyword is optional.

watercut - specifies the water cut x_1 as the volume fraction of the total liquid. It is used to calculate how much water to add to the hydrocarbon fluid. This keyword is optional.

includewater – indicates that water is to be included as one of the components present in the fluid mixture. This keyword is optional. If the watercut keyword is used, water must be present whether or not includewater is used.

total - specifies the total amount x_1 of hydrocarbon fluid the user wants to have present in current input units. This keyword is optional.
Note that the minimal specification requires only the keyword components and amounts and one or more values. The keywords units, gasunits, startsplit, nplus, gorunits, gorvalue, watercut and total may only be used after the keywords amounts, namounts and gasamounts.

Example

The following example creates a complete input stream from the experimental PVT analysis data which includes well defined components (N_2, H_2S, CO_2, methane ethane, propane, isobutane, butane, isopentane, neopentane, n-pentane), a group of cuts and the heavy end cut with a molecular weight of 515 and a specific gravity of 0.935. The cuts and heavy end will be split into 15 pseudocomponents starting at C6. A SARA analysis is provided so resin and asphaltene components will also be generated.

```
PVTanalysis infoanal2
mw fraction 515.0
sg fraction 0.935
components N2 H2S CO2 methane ethane propane
isobutane butane isopentane neopentane n-pentane C6 &
C30;
amounts 0.02 0.03 0.5 8.37 3.97 3.73 0.68 2.5 1.11 * 1.80
3.0 3.45 4.02 4.19 4.32 3.83 3.45 3.37 3.18 2.95 2.65 2.72
2.44 2.02 2.22 2.04 1.85 1.69 1.56 1.46 1.43 1.32 1.29 1.24
15.60;
units g
nplus          15
startsplit     C6
sara 64 23 12.2 0.8;
```

Black oil analysis

Experimental black oil analysis data can be directly used to specify an input stream for Multiflash using the BLACKOIL command. The minimum input for the Blackoil analysis consists of Specific gravity of stock tank oil and residual GOR (Rs). The oil properties such as Gas Gravity, Watson characterisation K factor and the Gas analysis are optional in the input but they are quite useful information to make the calculated bubblepoint or dewpoint more accurate.

The command format is in the following order.

```
BLACKOIL
   gasanalysis x_1, x_2, x_3, x_4, x_5, x_6, x_7,
   x_8 ;
   gorvalue        x_1
   gorunits        gor_unit
   gasgravity     x_1
   sg             x_1
   nplus           n
   npnplus         n
   startsplit      n or name (e.g. C6 )
   npstartsplit    n or name (e.g. N6 )
   pluslbound      x_1 x_2 x_3 ...;
   nppluslbound    x_1 x_2 x_3 ...;
   sara            wpsat wparo wpres wpasp
   estimatesara
   wax            x_1
   estimatewax
   watercut        x_1
   total           x_1;
```

gasanalysis – is followed by eight values for the fixed eight discrete components. The eight discrete components are Nitrogen, H2S, CO2, methane, ethane, propane, I-butane, n-butane. The information about the gas analysis is optional but it is useful to make the calculated bubblepoint or dewpoint more accurate.

gorvalue - specifies the gas-oil ratio x_1 at standard conditions (1 bar and 15ºC) as a pure volume ratio. The value must be in the units set by the gorunits keyword. This keyword is required in the input in order to make the blackoil analysis method work correctly.

gorunits - specifies the units in which the gorvalue is given.. Allowed values for gor_unit are: m3/m3 (default), or scf/stb . The unit conversion factor from scf/stb to m3/m3 is 0.1801175. This keyword is required in the blackoil input.

sg – specifies the specific gravity of the stock tank oil at standard condition. This keyword is required in the input of the blackoil analysis.

nplus - the integer number of pseudocomponents that the plusfraction will be split into. The default value is 1. The nplus value generated by the BLACKOIL command will be greater than the input

value if asphaltenes and resins are present and specified by SARA. This keyword is optional.

`npnplus` - the integer number of normal paraffin pseudocomponents that will be created. This value is required for characterising the normal paraffin distribution which is associated with the coutinho wax model; the default value is 15 if the keyword is omitted. This keyword only works with the Blackoil analysis method or the Infochem resvied characterisation method (`Infoanal2`).

`startsplit` - the cut sequence number or name (e.g. C20 etc.) at which the split is to start which can only be in the range 1 to (number of cuts + 1); the default is the latter value if this keyword is omitted. If the number of normal and non-normal cuts are different, then the smaller value will be used to calculate the default value of startsplit. This keyword is optional.

`npstartsplit` - the normal paraffin cut sequence number or name (e.g. N20 etc.) at which the normal paraffin split is to start; the default is the value used for N6 if this keyword is omitted.

`pluslbound` – followed by the amounts `x_1, x_2, x_3` etc. of the user-defined lower carbon-number boundaries for the pseudocomponents. This keyword is optional.

`nppluslbound` – followed by the amounts `x_1, x_2, x_3` etc. of the user-defined lower carbon-number boundaries for the normal paraffin pseudocomponents. This keyword is optional.

`sara` - indicates the presence of a saturates/aromatics/resins/ asphaltenes (SARA) analysis and must be followed by four values: the wt% of saturates, the wt% of aromatics, the wt% of resins and the wt% of asphaltenes. This keyword is optional.

EstimateSARA – indicates that the SARA will be estimated by Multiflash. This keyword is optional.

`Wax` – specifies the wax content (UOP) value which is used by Multiflash to estimate the normal paraffin distribution in association with the coutinho wax model. This keyword is optional.

Estimatewax – indicates that the wax content will be estimated by Multiflash. This keyword is optional.

`watercut` - specifies the water cut `x_1` as the volume fraction of the total liquid. It is used to calculate how much water to add to the hydrocarbon fluid. This keyword is optional.

`total` - specifies the total amount `x_1` of hydrocarbon fluid the user wants to have present in current input units. This keyword is optional.

Inhibitor calculator

The inhibitor calculator can be used to add the content of inhibitors (methanol, MEG, DEG, TEG and ethanol), the salt content of aqueous streams to mixtures. This converts the measured compositions of the inhibitors, salts in aqueous streams into an equivalent amount to that of water in the mixture for hydrate inhibition calculations. The command for adding Inhibitors is INHIBITOR. To use the INHIBITOR, it is necessary to define water and inhibitors into the stream from any databank first and then input amount of water in correct unit. The command format for adding the four inhibitors at the same time is:

```
INHIBITOR BASIS inhibitor_name1 inhibitor_value1
inhibitor_name2 inhibitor_value2 ...;
```

BASIS is used to specify fractions of inhibitors in either mass or mole or volume fraction. The keywords are MASSFRACTION, MOLEFRACTION and VOLUMEFRACTION respectively.

The inhibitor_name means the name of the inhibitors which includes methanol, MEG, DEG, TEG and ethanol and inhibitor_value is the fractions of the inhibitors. Note that the total of the fractions entered must be less than 1.

Salinity

The ion ratio and salt content of aqueous streams can be entered using the SALINITY command which converts the measured salt compositions into an equivalent amount of a salt pseudocomponent or ion ratios for freezing point depression or hydrate inhibition calculations. The properties have been tuned for use only with the RKSA model with Infochem mixing rules (see p. 43).

To use the SALINITY command it is first necessary to include water and saltcomponent (the salt pseudocomponent) or ions such as Na+ and Cl- in the components list. Water may be loaded from any databank. Saltcomponent or Na+, K+, Ca++, Cl- and Br- ions are defined in the INFODATA databank. There are three ways of entering the salt compositions or ions which are described in the following sections. In each case the amount of saltcomponent and ions equivalent to the salt composition is estimated and added to the input amounts.

Ion analysis

The concentrations of the following ions may be entered: Na^{+}, Ca^{++}, Mg^{++}, K^{+}, Sr^{++}, Ba^{++}, Fe^{++}, Cl^{-}, $SO4^{--}$, $HCO3^{-}$ and Br^{-}.

The command format is:

```
SALINITY model_selector ionconcentrations
   Na x  Ca x  Mg x  K x  Sr x  Ba x  Fe x
   Cl x  SO4 x  HCO3 x Br x
   spgravity sg;
```

The keyword model_selector is given as follows. If the model_selector is omitted, the default setting is saltcomp.

model_selector	Comments
saltcomp	For calculating the amount of a salt pseudocomponent proportional to the amount of water in the component list.
electrolyte	For calculating ion ratios of salt compositions proportional to the amount of water in the component list.
elecNaCl	For calculating ion ratios of salt compositions proportional to the amount of water in the component list. For this option to components introduced are restricted to Na^{+} and Cl^{-} ions which are added on a sodium chloride equivalent basis.

The ionconcentrations keyword sets the input mode for entering the salinity as ion concentrations in mg/l. There is no choice of units.

The concentration of each ion follows the ion name (in any order). Only those ions with non-zero concentrations need to be listed.

If the specific gravity of the solution is known it may be entered using the `spgravity` keyword. If omitted it is estimated.

Total dissolved solids

The command format is:

```
SALINITY model_selector tds x spgravity sg;
```

The `model_selector` selections are the same as described above. If `model_selector` is omitted, the default setting is `saltcomp`.

The `tds` keyword sets the input mode for entering the salinity as total dissolved solids in mg/l. There is no choice of units.

If the specific gravity of the solution is known it may be entered using the `spgravity` keyword. If omitted it is estimated.

Salt analysis

The concentrations of the following salts may be entered: NaCl, NaBr, $CaCl_2$, $MgCl_2$, KCl , $SrCl_2$, $BaCl_2$, $FeCl_2$, Na_2SO_4 and $NaHCO_3$

The command format is:

```
SALINITY model_selector analysis_type
   NaCl x  NaBr x CaCl2 x  MgCl2 x  KCl x  SrCl2 x
   BaCl2 x  FeCl2 x  Na2SO4 x NaHCO3 x ;
```

The `model_selector` selections are the same as described above. If the `model_selector` is omitted, the default setting is `saltcomp`.

The `analysis_type` keyword sets the input mode for entering the salinity as a salt analysis. There are three options for the units: `saltmassfractions`, `saltmolalities` and `saltmolefractions`.

The concentration of each salt in the units selected follows the salt name (in any order). Only those salts with non-zero concentrations need to be listed.

Example

The following example enters the amount of saltcomponent equivalent to a 13% by mass solution of NaCl.

```
PUREDATA Infodata;
units amounts g;
COMPONENTS water saltcomponent;
Amounts water 1.0;
SALINITY saltcomp saltmassfractions NaCl 0.13;
```

Model definition

Introduction

The thermodynamic and transport property models in Multiflash and their applicability are described in more detail in the *Models and Physical Properties Manual.*

The `MODEL` command is used to define a model. It has the format:

```
MODEL  model_id MF_model_name  [Model_options]... ;
```

`model_id` is a *user-defined* name that will be used to refer to the particular combination of the property model and options specified.

`MF_model_name` is the Multiflash name for the basic model. The list of recognised models is given below together with the applicable options.

`Model_options` are additional keywords that describe model variants, references to other, previously-defined, models or references to the source of binary interaction parameters.

For example,

```
MODEL MPR PR PRBIP;
```

defines the identifier MPR for the Peng-Robinson equation of state with the set of BIPs called PRBIP.

Equation of state models

Ideal gas equation of state

$$p = \frac{NRT}{V}$$

part of model definition	Multiflash keyword	Comments
Model name	`IDG`	

The following example defines the ideal gas eos model and gives it the identifier `MIDG`

```
model MIDG idg;
```

Benedict-Webb-Rubin-(Starling) equation of state

$$p = \frac{RT}{V}\left(N + \frac{B}{V} + \frac{C}{V^2} + \frac{D}{V^5} + \frac{C'}{V^2}\left(1 + \frac{\gamma^2}{V^2}\right)\exp\left(-\frac{\gamma^2}{V^2}\right)\right)$$

Reference: Starling, *Fluid thermodynamic properties for light petroleum systems*, Gulf Publishing Co., Houston (1973).

part of model definition	Multiflash keyword	Comments
Model name	`BWRS`	

model variant	`* or BWRS`
	`SH`
	
	`BWR`
		BWRS equation
		BWRS equation using parameters from Starling and Han correlations
		BWR equation with parameters from Orye and others

BIP data	`Bip_set`	optional, `Bip_set` is the name for a set of BIPs defined by the BIPSET command.

The following example defines the BWRS eos model and BIP taken from a (previously defined) bipset called BWRSBIP and gives it the identifier `MBWRS`

```
model MBWRS bwrs bwrs BWRSBIP;
```

Hayden-O'Connell gas phase model

This treats each component in the gas phase as forming a monomer-dimer equilibrium. For most components that deviate only slightly from ideal behaviour, the model reduces to the volume-explicit virial equation:

$$V = \frac{RT}{p} + B$$

The second virial coefficient B is estimated for each component from a generalised correlation (J.G. Hayden and J.P. O'Connell, Ind. Eng. Chem.. Proc. Des. Dev, **14**, 209 (1975)). This correlation accounts for non-polar, polar and chemical association effects. The pure component properties required by the model are: critical temperature, critical pressure, radius of gyration, dipole moment and an empirical association parameter. Values for these quantities are stored in the Infodata databank.

A second virial coefficient model such as HOC can account for gas phase non-idealities up to pressures of about 5 to 10 bar. The implementation of the HOC model in Multiflash allows the vapour phase association of substances such as acetic acid to be represented.

part of model definition	Multiflash keyword	Comments
Model name	`HOCONNELL`	

The following example defines the Hayden-O'Connell model and gives it the identifier `MHOC`

```
model MHOC hoconnell;
```

Lee-Kesler-(Plöcker) equation of state

This is a 3-parameter corresponding states model based on interpolating the reduced properties of a mixture between those of two reference substances, one spherical and the other non-spherical. The equation for each property is of the form

$$z^{mix} = z^{(0)} + \frac{\omega}{\omega^{(1)}}\left[z^{(1)} - z^{(0)}\right]$$

The method is rather slow and complex compared with cubic eos but can yield accurate predictions of density and enthalpy for non-polar mixtures. It is not particularly recommended for phase equilibrium calculations. The model definition is

part of model definition	Multiflash keyword	Comments
Model name	`LKP`	
model variant	`* or LKP` `LK`	Lee-Kesler Plöcker mixing rule original Lee-Kesler mixing rule
BIP data	`Bip_set`	optional, `Bip_set` is the name for a set of BIPs defined by the BIPSET command. The model can accept one temperature-independent BIP (per pair of components)

The following example defines the LKP eos model and gives it the identifier `MLKP`

```
model MLKP lkp;
```

Peng-Robinson equation of state

$$p = \frac{NRT}{V-b} + \frac{a}{V^2 + 2bV - b^2}$$

part of model definition	Multiflash keyword	Comments
Model name	`PR`	
BIP data	`Bip_set`	optional, `Bip_set` is the name for a set of BIPs defined by the BIPSET command

The following example defines the PR eos model and gives it the identifier `MPR`

```
model MPR pr;
```

Advanced Peng-Robinson equation of state

The enhanced Infochem version of the PR eos offers a wide range of options for various aspects of the eos

part of model definition	Multiflash keyword	Comments
Model name	PRA	
method for calculating eos *a* parameter	* or PR PR78 PSAT PSAT78 MC	standard PR method PR78 method fit parameter to reproduce saturated vapour pressure using correlation from databank as above but for PR78 method use Mathias-Copeman parameters
method for calculating eos *b* parameter	* or PR LDEN	standard PR method fit parameter to reproduce saturated liquid density at 298K or Tr=0.7 (Peneloux method)
Mixing rules	* or VDW GEX HV NRTL HVPSM	standard VDW 1-fluid excess Gibbs energy MHV2 Huron-Vidal Infochem modified NRTL Huron-Vidal-Pedersen (HVP)
Excess Gibbs energy model	Gex_model	required for GEX and HV, must be the model identifier for an activity coefficient model
BIP data	Bip_set	optional, Bip_set is the identifier for a set of BIPs defined by the BIPSET command

The following example first defines a liquid phase activity model MNRTLVLE using the NRTL equation. It then defines an equation of state model MPRA based on PR with the *a* parameter fitted to the vapour pressure and the *b* parameter fitted to the liquid density. The mixing rule is MHV2 which uses the excess Gibbs energy model defined for MNRTLVLE, *i.e.* NRTL.

```
model MNRTLVLE nrtl vle;
model MPRA pra psat lden gex MNRTLVLE;
```

The following example defines the PR78 eos model and gives it the identifier MPR78

```
model MPR78 pra pr78 lden vdw PR78BIP;
```

For the PRA78 model,

```
model MPRA78 pra psat78 lden vdw PR78BIP;
```

For the PR-HVP model.

```
model MPRHVP pra pr lden hvpsm PRHVPBIP3;
```

Redlich-Kwong-(Soave) equation of state

$$p = \frac{NRT}{V-b} + \frac{a}{V(V+b)}$$

part of model definition	Multiflash keyword	Comments
Model name	RKS	
method for calculating eos *a* parameter	* or RKS API RK	Soave modification of RK API modification of RKS original RK
BIP data	Bip_set	optional, Bip_set is the name for a set of BIPs defined by the BIPSET command

The following example defines the RKS eos model with the API modification and BIP taken from a (previously defined) bipset called RKSBIP and gives it the identifier MRKS

```
model MRKS rks api RKSBIP;
```

Advanced RKS equation of state

The enhanced Infochem version of the RKS eos offers a wide range of options for various aspects of the eos

part of model definition	Multiflash keyword	Comments
Model name	RKSA	
method for calculating eos *a* parameter	* or RKS API RK PSAT MC	Soave modification of RK API modification of RKS original RK eos fit parameters to reproduce saturated vapour pressure using correlation from databank use Mathias-Copeman parameters
method for calculating eos *b* parameter	* or RKS LDEN	standard RKS method fit parameter to reproduce saturated liquid density at 298K or Tr=0.7 (Peneloux method)
Mixing rules	* or VDW GEX HV NRTL HVPSM	standard VDW 1-fluid excess Gibbs energy MHV2 Huron-Vidal Infochem modified NRTL Huron-Vidal-Pedersen (HVP)
Excess Gibbs energy model	Gex_model	required for GEX and HV, must be the model identifier for an activity coefficient model
BIP data	Bip_set	optional, Bip_set is the identifier for a set of BIPs defined by the BIPSET command

The following example sets up the version of the RKS eos recommended for modelling the fluid phases in hydrate calculations.

```
model MRKSANRTL rksa psat lden nrtl;
```

PSRK equation of state

This model consists of the RKSA equation of state with vapour pressures fitted using Mathias-Copeman parameters (if available), the Peneloux volume correction and the PSRK type mixing rules. The excess Gibbs energy is provided by the PSRK variant of the Unifac method. This is the same as the normal VLE Unifac model except that the group table has been extended to include a large number of common light gases.

The PSRK model is an extension of the Unifac method. It is intended to predict the phase behaviour of a wide range of polar mixtures using the solution of groups concept as embodied in Unifac. The main benefit of PSRK is that it is able to handle mixtures containing gases much better than Unifac and unlike a normal equation of state it can handle polar liquids. This is because (a) it uses an equation of state with an excess Gibbs energy mixing rules thereby avoiding problems of how to handle supercritical components in an activity coefficient equation; (b) the Unifac group parameter table has been extended in PSRK to include 32 common light gases.

part of model definition	Multiflash keyword	Comments
Model name	UNIFAC	
Model variant	PSRK	the PSRK type mixing rules, extension of the Unifac method
Model name	RKSA	
method for calculating eos *a* parameter	PSAT	fit parameters to reproduce saturated vapour pressure using correlation from databank
method for calculating eos *b* parameter	LDEN	fit parameter to reproduce saturated liquid density at 298K or Tr=0.7 (Peneloux method)
Mixing rules	PSRK	the PSRK type mixing rules, extension of the Unifac method.
Excess Gibbs energy model	Gex_model	must be the PSRK Unifac model identifier

The following example sets up the version of the PSRK eos recommended for modelling the fluid phases.

```
model MUNIFACPSRK UNIFAC PSRK;

model MRKSAPSRK RKSA MC LDEN PSRK MUNIFACPSRK;
```

Consult the section on Unifac for details of how to use more options for the definition of the Unifac groups and subgroups used, etc.

PSRK-NRTL equation of state

This model is the same as the PSRK model except that the excess Gibbs energy is provided by the NRTL equation. So unlike the PSRK equation, the PSRK-NRTL variant requires that BIPs are provided for the NRTL equation in order to give accurate results. The model definition is set out in the following table.

part of model definition	Multiflash keyword	Comments
Model name	NRTL	
type of phase behaviour	* or VLE	Vapour-liquid equilibrium default for alpha parameter
vapour phase model	*	Leave undefined as not required for role as Gex model
BIP data	Bip_set	Optional, Bip_set is the name for a set of BIPs defined by the BIPSET command; requires 3 BIPs A_{ij}, A_{ji} and α_{ij}
Model name	RKSA	
method for calculating eos *a* parameter	PSAT	fit parameters to reproduce saturated vapour pressure using correlation from databank
method for calculating eos *b* parameter	LDEN	fit parameter to reproduce saturated liquid density at 298K or Tr=0.7 (Peneloux method)
Mixing rules	PSRK	the PSRK type mixing rules, extension of the Unifac method.
Excess Gibbs energy model	Gex_model	must be the PSRK Unifac model identifier

Example

```
model MNRTL nrtl vle * NRTLBIP3;

model MRKSAPSRK RKSA MC LDEN PSRK MNRTL;
```

Cubic plus association (CPA) model

The CPA model consists of the Redlich-Kwong-Soave equation plus an additional term based on Wertheim's theory that represents the effect of chemical association.

The CPA model also uses the Peneloux density correction to match the liquid density calculated from the equation of state to that stored in the chosen physical property data system at a reduced temperature of 0.7 or at 298.15K, whichever temperature is the lower.

The CPA model may be used for hydrate calculations with methanol, MEG and salt inhibition, as these are the only cases for which parameters are currently provided. This will be extended in future versions.

part of model definition	Multiflash keyword	Comments
Model name	ASSOC	
method for calculating eos *a* parameter	PSAT	fit parameters to reproduce saturated vapour pressure using correlation from databank.
Bipdata	bipset	BIP values for association model – physical term.
Bipdata	bipset	BIP values for association model – association term.

The following example sets up the CPA association model. BIP values are taken from the bipset ASSOCBIP and ASSOCBIP-2.

```
bipset ASSOCBIP 1;
bipset ASSOCBIP-2 1;
model MASSOC ASSOC PSAT ASSOCBIP ASSOCBIP-2;
```

Zudkevitch Joffee model

The ZJ eos is based on the original version of Redlick-Kwong (RK) eos.

For the original version of RK, the “a” and “b” parameters are expressed explicitly in terms of the critical temperature and pressure

For the ZJ eos, the a_i and b_i parameters are defined by simultaneously solving the equations of fugacity coefficients along the saturation line and the RKS equation for both vapour and saturated liquid phase, where the required saturation vapour pressure and saturated liquid density are obtained from the correlation of the pure components.

part of model definition	Multiflash keyword	Comments
Model name	`ZJ`	
Variant	`ZJ`	The ai and bi parameters are fitted to match vapour and saturated liquid density
Bipdata	`bipset`	BIP values for ZJ physical term.

PC-SAFT model

The PC-SAFT equation is a development of the SAFT model that has been shown to give good results for a wide range of polar and non-polar substances including polymers. Polymers are one of the most important areas of application of PC-SAFT. The model appears to be one of the most accurate and realistic equations of state currently available for modelling polymer systems.

PC-SAFT stands for the Perturbed Chain Statistical Associating Fluid Theory and it incorporates current ideas of how to model accurately the detailed thermodynamics of fluids within the framework an equation of state. The mathematical structure is very complex and cannot be conveniently described in a manual. Users are referred to Appendix A of the reference given in the *Models and Physical Properties Manual.*

It is a development of the SAFT model that has been shown to give good results for a wide range of polar and non-polar substances including polymers. The model appears to be one of the most accurate and realistic equations of state currently available for modelling polymer systems.

Two version of PC-SAFT models are available in the latest version of Multiflash.

part of model definition	Multiflash keyword	Comments
Model name	SAFT	
Model variant	PC	The model variant identifier for the original PC-SAFT model.
Model variant	ORIGINAL	The second model variant for PC-SAFT model
Bipdata	bipset	BIP values for PC-SAFT model – attraction term
Bipdata	bipset	BIP values for PC-SAFT model – association term

The following example sets up the original PC-SAFT model. BIP values are taken from the bipsets SAFTBIP and SAFTBIP-2.

```
bipset SAFTBIP 1 constant eos none ;
bipset SAFTBIP-2 1 constant association J/mol ;
model MPCSAFT SAFT PC ORIGINAL SAFTBIP SAFTBIP-2;
```

The simplified version of PC-SAFT is a simplified version of PC-SAFT by the Danish Technical University (which is located at Lyngby in Denmark).

part of model definition	Multiflash keyword	Comments
Model name	SAFT	
Model variant	LYNGBY	The model variant identifier for the simplified version of PC-SAFT model.
Model variant	ORIGINAL	The second model variant for the simplified version of PC-SAFT model
Bipdata	bipset	BIP values for PC-SAFT model – attraction term
Bipdata	bipset	BIP values for PC-SAFT model – association term

The following example sets up the simplified version of PC-SAFT model.

```
bipset SAFTBIP 1 constant eos none ;
bipset SAFTBIP-2 1 constant association J/mol ;
model MPCSAFTLB SAFT LYNGBY ORIGINAL SAFTBIP SAFTBIP-
2;
```

Multi-reference fluid corresponding states (CSMA) model

The CSMA model is based on a collection of very accurate equations of state for a number of reference fluids. It will provide accurate values of properties for any of the reference fluids (see below for a list) and it uses a 1-fluid corresponding states approach to estimate mixture properties. It is formulated so that mixture properties will reduce to the (accurate) pure component values as the mixture composition approaches each of the pure component limits. In Multiflash 4.0, the GERG-2004 mixing rule is also available.

The model definition can be considered in two distinct parts: the definition of pseudo-critical properties for a mixture (mixing rules), and the prescription for combining the properties of the reference substances to give the total mixture properties (combining rules).

The current model implementation includes reference equations of state for the following substances: ammonia, argon, n-butane, iso-

butane, 1-butene, cis-2-butene, trans-2-butene, cyclohexane, CO, CO_2, COS, ethane, ethanol, ethylene, fluorine, helium, heptane, hexane, iso-hexane, hydrogen, H_2S, krypton, methane, neon, nitrogen, NF3, octane, oxygen, n-pentane, iso-pentane, neo-pentane, propane, propylene, SF6, SO2, toluene, water (IAPSW 95), xenon, R11, R113, R114, R115, R116, R12, R123, R124, R125, R13, R134a, R14, R143a, R152a, R22, R227fa, R23, R245fa, R32 and RC318. The equations of state are taken from various sources and do not all have the same quality or range of applicability. Other hydrocarbons and petroleum fractions are included using a generalised equation of state.

part of model definition	Multiflash keyword	Comments
Model name	CSMA	
Mixing rules	VDW or GERG	standard VDW 1-fluid or the GERG-2008 mixing rule.
BIP data	bipset	BIP values for the mixing rules for the critical temperature
BIP data	bipset	BIP values for the mixing rules for the critical molar volume. Default value is set to 1.
BIP data	bipset	With the GERG-2008 mixing rule, the binary weighting factor for the departure function contribution in the mixture is required that is defined in the INFOBIPS databank.

The following example sets up the advanced version of CSM model with VDW mixing rule and the definition of the fluid phases.

```
model MCSMA CSMA VDW CSMABIP;
bipset CSMABIP 1 constant eos none ;
bipset CSMABIP-2 1 constant eosv none ;
bipdata INFOBIPS ;
model MCSMA CSMA VDW CSMABIP CSMABIP-2;
pd GAS gas MCSMA;
pd LIQUID1 liquid MCSMA;
```

With the GERG-2004 mixing rule, the example sets up the advanced version of CSM model and the definition of the fluid phases are:

```
bipset CSMABIP 1 quadratic gergt none ;
bipset CSMABIP-2 1 quadratic gergv none ;
bipset CSMABIP-3 1 constant gergf none ;
bipdata INFOBIPS ;
model MCSMA CSMA GERG CSMABIP CSMABIP-2 CSMABIP-3;
pd GAS gas MCSMA;
pd LIQUID1 liquid MCSMA;
```

In Multiflash 4.1, the CSMA model also includes the GERG-2008 natural gas model. This is an industry-standard high-accuracy model for mixtures of natural gas components: methane, nitrogen, C02, ethane, propane, n-butane, iso-butane, n-pentane, iso-pentane, hexane, heptane, octane, nonane, decane, argon, oxygen, hydrogen, hydrogen sulphide, CO, water, helium. The model includes appropriate BIPs for all components in the list.

The model is fully described in the publications:

- O. Kunz, R. Klimeck, W. Wagner, M. Jaeschke, The GERG-2004 wide-range equation of state for natural gases and other mixtures, GERG Technical Monograph 15 (2007)

- Kunz, O., Wagner, W. The GERG-2008 wide-range equation of state for natural gases and other mixtures: An expansion of GERG-2004. To be submitted to J. Chem. Eng. Data (2011)

The model is very accurate for pure substances that are included in the above list of reference substances. It is also applicable to near-ideal mixtures such as air but for the best results it is necessary to fit values of the binary interaction parameters to match experimental data. For the natural gas mixtures or non-ideal mixtures such as water + CO_2, the GERG-2008 model should be used.

part of model definition	Multiflash keyword	Comments
Model name	GERG	
Mixing rules	GERG	GERG-2004 model for natural gas mixtures.
BIP data	bipset	BIP values for the mixing rules for the critical temperature
BIP data	bipset	BIP values for the mixing rules for the critical molar volume. Default value is 1.
BIP data	bipset	The binary weighting factor for the departure function contribution in the mixture.

The following example sets up the GERG-2008 model with BIPs and the definition of the fluid phases.

```
bipset GERGBIP 1 quadratic gergt none ;
bipset GERGBIP-2 1 quadratic gergv none ;
bipset GERGBIP-3 1 constant gergf none ;
bipdata INFOBIPS ;
model MGERG GERG GERG GERGBIP GERGBIP-2 GERGBIP-3;
pd GAS gas MGERG;
pd LIQUID1 liquid MGERG;
```

Steam Tables (IAPWS-95)

The international standard formulation for the properties of water is made up from the following components. The thermodynamic properties are obtained from the IAPWS95 formulation. The viscosity, thermal conductivity and surface tension are based on the latest IAPWS release. The references can be found in the *Models and Physical Properties Manual*.

part of model definition	Multiflash keyword	Comments
Model name	IAPWS95	

The following example sets up the model.

```
model MSTEAM IAPWS95;
```

CO2 high-accuracy model

This is a model for pure carbon dioxide using the most accurate correlations currently available. The model is based on the high accuracy equation of state published by Span and Wagner (1996). The viscosity is described by the correlation of Fenghour et al (1998) and the thermal conductivity uses the correlation of Scalabrin et al. (2006).

The references can be found in the *Models and Physical Properties Manual*.

part of model definition	Multiflash keyword	Comments
Model name	HACO2	

The following example sets up the model.

```
model MCO2 HACO2;
pd GAS gas MCO2;
pd LIQUID1 liquid MCO2;
```

Water/Amonia high-accuracy model

The high accuracy corresponding state model for water-ammonia binary system is based on the work developed by Beiner Tillner-Roth and Daniel G . Friend. The references can be found in the *Models and Physical Properties Manual*.

part of model definition	Multiflash keyword	Comments
Model name	WAMM	
Model variant	WAMM	Only variant available
Bipdata	Bipset	
Bipdata	Bipset	
Bipdata	Bipset	

The following example sets up the model.

```
bipset WAMMBIP 1 quadratic gergt none ;

bipset WAMMBIP-2 1 quadratic gergv none ;

bipset WAMMBIP-3 1 constant gergf none ;

bipdata INFOBIPS ;

model MWAMM WAMM WAMM WAMMBIP WAMMBIP-2 WAMMBIP-3;pd
GAS gas MWAMM;

pd LIQUID1 liquid MWAMM;

keys LIQUID1 *;

pd WATER liquid MWAMM;

keys WATER AQUEOUS;
```

Activity models

Ideal solution liquid activity method

$$\frac{G^E}{RT} = 0$$

part of model definition	Multiflash keyword	Comments
Model name	IDL	
vapour phase model	V_model	Model name for an equation of state model that is used to obtain the reference state vapour properties; not needed if the model is only used as the Gex_model in one of the advanced equations of state
Poynting correction	POYNTING or NOPOYNTING	Optionally selects if Poynting correction should be added to activity coefficients (default) or not.

The following example first defines the ideal gas eos model and then sets up the ideal liquid model with the identifier MIDLMIDG

```
model MIDG idg;
model MIDLMIDG idl MIDG POYNTING;
```

Or if the Poynting correction is not required

```
model MIDG idg;
model MIDLMIDG idl MIDG NOPOYNTING;
```

The default is to include the Poynting correction in which case the keyword can be omitted, which is done in subsequent examples of activity models.

```
model MIDG idg;
model MIDLMIDG idl MIDG;
```

Regular solution method

Regular solution theory can be used for vapour-liquid calculations for mixtures of non-polar or slightly polar components. The theory is applicable to systems which exhibit negligible entropies and volumes of mixing. However, it has been largely superseded by equations of state

Flory-Huggins theory is able to describe systems which include some long chain molecules. It has consequently applied to model polymer systems but it has been largely superseded by other models such as PC-SAFT.

part of model definition	Multiflash keyword	Comments
Model name	REGULAR	
Model Variant	REGULAR or FH	The model variants identifier for either the regular solution or Flory Huggins regular solution model.
vapour phase model	V_model	Model name for an equation of state model that is used to obtain the reference state vapour properties; not needed if the model is only used as the Gex_model in one of the advanced equations of state
Poynting correction	POYNTING or NOPOYNTING	Optionally selects if Poynting correction should be added to activity coefficients (default) or not.

The following example first defines the ideal gas eos model and then sets up the ideal liquid model with the identifier MREGULARMIDG

```
bipset REGULARBIP 1 constant eos none ;
model MIDG idg;
model MREGULARMIDG REGULAR REGULAR MIDG REGULARBIP;
```

For the Flory Huggins regular solution model:

```
bipset REGULARBIP 1 constant activity J/mol ;
model MIDG IDG;                                    model
MFHMIDG REGULAR FH MIDG REGULARBIP;
```

NRTL liquid activity method

This model may be used for vapour-liquid, liquid-liquid and vapour-liquid-liquid equilibrium calculations. The VLE option should be used for VLLE. It is necessary to supply BIP values to obtain accurate predictions.

$$G^E = \sum_i n_i \frac{\sum_j n_j A_{ji} G_{ji}}{\sum_j n_j G_{ji}}$$

$$G_{ij} = \exp\left(-\frac{\alpha_{ij} A_{ij}}{RT}\right)$$

part of model definition	Multiflash keyword	comments
Model name	NRTL	
type of phase behaviour	* or VLE LLE	Vapour-liquid equilibrium or liquid-liquid equilibrium; sets default for alpha parameter
vapour phase model	V_model	Model name for an equation of state model that is used to obtain the reference state vapour properties; not needed if the model is only used as the Gex_model in one of the advanced equations of state
BIP data	Bip_set	Optional, Bip_set is the name for a set of BIPs defined by the BIPSET command; requires 3 BIPs A_{ij}, A_{ji} and α_{ij}
Poynting correction	POYNTING or NOPOYNTING	Optionally selects if Poynting correction should be added to activity coefficients (default) or not.

The following example first defines the rk eos model and then sets up the vle version of the NRTL activity model with the identifier MNRTLMRKS. BIP values can be taken from the databank INFOBIPS.BIN for VLE or INFOLLBIPS.BIN for LLE by defining the BIP databank as

```
bipdata infobips;
```

Then set the BIP values using the command, for example

```
bipset NRTLBIP3 3 constant activity J/mol;
```

followed by model definitions.

```
model MRKS rks rk;
model MNRTLMRKS nrtl vle MRKS NRTLBIP3;
```

UNIQUAC liquid activity method

This model may be used for vapour-liquid, liquid-liquid and vapour-liquid-liquid equilibrium calculations. It is necessary to supply BIP values to obtain accurate predictions.

$$\frac{G^E}{RT} = \sum_i n_i \ln\left(\frac{r_i \sum_j n_j}{\sum_j r_j n_j}\right) + \frac{z}{2}\sum_i q_i n_i \ln\left(\frac{q_i \sum_j r_j n_j}{r_i \sum_j q_j n_j}\right) + \sum_i q_i n_i \ln\left(\frac{q_i \sum_j G_{ji} q_j n_j}{\sum_j q_j n_j}\right)$$

$$G_{ij} = \exp\left(-\frac{A_{ij}}{RT}\right), \qquad z = 10$$

part of model definition	Multiflash keyword	Comments
Model name	`UNIQUAC`	
vapour phase model	`V_model`	Model name for an equation of state model that is used to obtain the reference state vapour properties; not needed if the model is only used as the `Gex_model` in one of the advanced equations of state
BIP data	`Bip_set`	Optional, `Bip_set` is the name for a set of BIPs defined by the BIPSET command; requires 2 BIPs A_{ij} and A_{ji}
Poynting correction	`POYNTING` or `NOPOYNTING`	Optionally selects if Poynting correction should be added to activity coefficients (default) or not.

The following example first defines the RKS eos model and then sets up the UNIQUAC activity model with the identifier `MUNIQUACMRKS`. BIP values can be taken from the databank INFOBIPS.BIN for VLE or INFOLLBIPS.BIN for LLE by defining the BIP databank as

```
bipdata infobips;
```

Then set the BIP values using the command, for example

```
bipset UNIQUACBIP2 2 constant activity J/mol;
```

followed by model definitions.

```
model MRKS rks rks;
model MUNIQUACMRKS uniquac MRKS UNIQUACBIP2;
```

Wilson activity method: A variant

This model may be used for vapour-liquid equilibrium calculations but it is not capable of predicting liquid-liquid immiscibility. It is necessary to supply BIP values to obtain accurate predictions.

$$\frac{G^E}{RT} = \sum_i n_i \ln\left(\frac{\sum_j A_{ij} n_j}{\sum_j n_j}\right)$$

part of model definition	Multiflash keyword	Comments
Model name	WILSON	
variant	A	The variant determines the way in which BIPs are defined
vapour phase model	V_model	Model name for an equation of state model that is used to obtain the reference state vapour properties; not needed if the model is only used as the Gex_model in one of the advanced equations of state
BIP data	Bip_set	Optional, Bip_set is the name for a set of BIPs defined by the BIPSET command; requires 2 BIPs, A_{ij} and A_{ji}
Poynting correction	POYNTING or NOPOYNTING	Optionally selects if Poynting correction should be added to activity coefficients (default) or not.

The following example first defines the rk eos model and then sets up the Wilson-A activity model with the identifier MWILSONAMRKS. BIP values are taken from the (previously defined) bipset WILSONBIP2

```
model MRKS rks rk;
model MWILSONAMRKS Wilson A MRKS WILSONBIP2;
```

Wilson activity method: E variant

This model may be used for vapour-liquid equilibrium calculations but it is not capable of prediction liquid-liquid immiscibility. It is necessary to supply BIP values to obtain accurate predictions.

$$\frac{G^E}{RT} = \sum_i n_i \ln\left(\frac{\sum_j G_{ij} n_j}{\sum_j n_j}\right)$$

$$G_{ij} = \frac{V_j^*}{V_i^*} \exp\left(-\frac{A_{ij}}{RT}\right)$$

part of model definition	Multiflash keyword	Comments
Model name	WILSON	
variant	* or E	the variant determines the way in which BIPs are defined
vapour phase model	V_model	model name for an equation of state model that is used to obtain the reference state vapour properties; not needed if the model is only used as the Gex_model in one of the advanced equations of state
BIP data	Bip_set	Optional, Bip_set is the name for a set of BIPs defined by the BIPSET command; requires 2 BIPs, A_{ij} and A_{ji}, plus pure component saturated liquid volume V_i^*
Poynting correction	POYNTING or NOPOYNTING	Optionally selects if Poynting correction should be added to activity coefficients (default) or not.

The following example first defines the rk eos model and then sets up the Wilson-E activity model with the identifier MWILSONEMRKS. BIP values can be taken from the databank INFOBIPS.BIN by defining the BIP databank as

```
bipdata infobips;
```

Then set the BIP values using the command, for example

```
bipset WILSONBIP2 2 constant activity J/mol;
```

followed by model definitions.

```
model MRKS rks rk;
model MWILSONEMRKS Wilson E MRKS WILSONBIP2;
```

UNIFAC liquid activity method

The UNIFAC method is similar to UNIQUAC but interaction parameters are predicted based on the molecular group structure of the components in a mixture. The model is completely predictive and does not require any BIPs.

part of model definition	Multiflash keyword	Comments
Model name	UNIFAC	
type of phase behaviour	* or VLE LLE	Vapour-liquid equilibrium or liquid-liquid equilibrium; determines which group interaction parameters are used
vapour phase model	V_model	Model name for an equation of state model that is used to obtain the reference state vapour properties; not needed if the model is only used as the *Gex_model* in one of the advanced equations of state
Poynting correction	POYNTING or NOPOYNTING	Optionally selects if Poynting correction should be added to activity coefficients (default) or not.

Using the more comprehensive UNIFACA keyword, the definitions are:

part of model definition	Multiflash keyword	Comments
Model name	UNIFACA	
Model Variant	ORIGINAL *, VLE or LLE ORIGINAL	Three model variant identifiers for selecting the form of the combinatorial term and the residual term; the third variant keyword is currently not used
vapour phase model	V_model	Model name for an equation of state model that is used to obtain the reference state vapour properties; not needed if the model is only used as the Gex_model in one of the advanced equations of state
Poynting correction	POYNTING or NOPOYNTING	Optionally selects if Poynting correction should be added to activity coefficients (default) or not.

The following example first defines the rk eos model and then sets up the VLE version of the UNIFAC activity model with the identifier MUNIFACVLEMRKS.

```
model MRKS rks rk;
model MUNIFACVLEMRKS UNIFAC VLE MRKS;
```

Or,

```
model MRKS rks rk;
model MUNIFACVLEMRKS UNIFACA ORIGINAL VLE ORIGINAL
MRKS;
```

UNIFAC for the PSRK equation

This version of Unifac has been extended to include a number of common gases and is designed for use in the Gex mixing rules of the PSRK equation of state. Although the user can define a vapour phase model as for all versions of Unifac, it is unnecessary when the model is used as part of the PSRK equation. In Multiflash, the PSRK model is taken to include the correction for hydrocarbons defined in *Prediction of vapor-liquid equilibria for asymmetric systems at low and high pressures with the PSRK model* by Li, Fischer and Gmehling from Fluid Phase Equilibria, **143**, 71, (1998). Using the simpler UNIFAC keyword, the definition follows.

part of model definition	Multiflash keyword	Comments
Model name	UNIFAC	
Model Variant	PSRK	Selects the PSRK variant
vapour phase model	V_model	Model name for an equation of state model that is used to obtain the reference state vapour properties; not needed if the model is only used as the Gex_model in one of the advanced equations of state
Poynting correction	POYNTING or NOPOYNTING	Optionally selects if Poynting correction should be added to activity coefficients (default) or not.

With the UNIFACA keyword, the definition is:

part of model definition	Multiflash keyword	Comments
Model name	UNIFACA	
Model Variant	ORIGINAL PSRK ORIGINAL	Three model variant identifiers for selecting the form of the combinatorial term and the residual term; the third variant keyword is currently not used
vapour phase model	V_model	Model name for an equation of state model that is used to obtain the reference state vapour properties; not needed if the model is only used as the Gex_model in one of the advanced equations of state

With the UNIFACA keyword, it is also possible to specify the PSRK model without the hydrocarbon correction of Li, Fischer and Gmehling:

part of model definition	Multiflash keyword	Comments
Model name	UNIFACA	
Model Variant	ORIGINAL PSRKNOHCC ORIGINAL	Three model variant identifiers for selecting the form of the combinatorial term and the residual term; the third variant keyword is currently not used
vapour phase model	V_model	Model name for an equation of state model that is used to obtain the reference state vapour properties; not needed if the model is only used as the Gex_model in one of the advanced equations of state
Poynting correction	POYNTING or NOPOYNTING	Optionally selects if Poynting correction should be added to activity coefficients (default) or not.

The following example defines PSRK Unifac without a gas phase model.

```
model MUNIFACPSRK UNIFAC PSRK;
```

Or,

```
model MUNIFACPSRK UNIFACA ORIGINAL PSRK ORIGINAL;
```

Or without the hydrocarbon correction,

```
model MUNIFACPSRK UNIFACA ORIGINAL PSRKNOHCC
ORIGINAL;
```

Dortmund Modified UNIFAC method

The Dortmund modified UNIFAC is a variant of UNIFAC that is better able to represent the simultaneous vapour-liquid equilibria, liquid-liquid equilibria and excess enthalpies of polar mixtures. Like original UNIFAC, however, it does not allow for the presence of light gases in the mixture.

part of model definition	Multiflash keyword	Comments
Model name	UNIFACA	
Model Variant	DORTMUND DORTMUND ORIGINAL	Three model variant identifiers for selecting the form of the combinatorial term and the residual term; the third variant keyword is currently not used
vapour phase model	V_model	Model name for an equation of state model that is used to obtain the reference state vapour properties; not needed if the model is only used as the Gex_model in one of the advanced equations of state
Poynting correction	POYNTING or NOPOYNTING	Optionally selects if Poynting correction should be added to activity coefficients (default) or not.

The following example first defines the ideal gas eos model and then sets up the ideal liquid model with the identifier MUNIFACADMMIDG

```
model MIDG idg;
model MUNFACADMMIDG UNIFACA DORTMUND DORTMUND
ORIGINAL MIDG;
```

UNIFAC groups and subgroups

In Multiflash, all variants of UNIFAC are handled using the same set of generic groups and subgroups. The current complete list of groups and subgroups which cover all the latest public-domain versions of original UNIFAC, PSRK and Dortmund modified UNIFAC are shown in the following table. The *q* and *r* values used in Multiflash are given for original UNIFAC and PSRK UNIFAC in columns 3 and 4, while values for Dortmund modified UNIFAC are given in columns 5 and 6. Entries marked not applicable (n/a) denote subgroups that are not part of the particular variant of UNIFAC. When two main group names are given, the first is the main group assignment for original and PSRK UNIFAC while the second is for Dortmund modified UNIFAC.

Subgroup	Main group	r	q	r Dortmund	q Dortmund
CH3	CH2	0.9011	0.848	0.6325	1.0608
CH2		0.6744	0.54	0.6325	0.7081
CH		0.4469	0.228	0.6325	0.3554
C		0.2195	0	0.6325	0
CH2=CH	C=C	1.3454	1.176	1.2832	1.6016
CH=CH		1.1167	0.8670	1.2832	1.2489
CH=C		0.8886	0.676	1.2832	0.8962
CH2=C		1.1173	0.988	1.2832	1.2489
C=C		0.6605	0.485	1.2832	0.4582
H2C=CH2		1.3564	1.3098	n/a	n/a
ACH	ACH	0.5313	0.4	0.3763	0.4321
AC		0.3652	0.12	0.3763	0.2113
ACCH3	ACCH2	1.2663	0.968	0.91	0.949
ACCH2		1.0396	0.66	0.91	0.7962
ACCH		0.8121	0.348	0.91	0.3768
OH	OH	1.0	1.2	1.2302	0.8927
s-OH		n/a	n/a	1.063	0.8663
t-OH		n/a	n/a	0.6895	0.8345
CH3OH	CH3OH	1.4311	1.432	0.8585	0.9938
H2O	H2O	0.92	1.4	1.7334	2.4561
ACOH	ACOH	0.8952	0.68	1.08	0.975
CH3CO	CH2CO	1.6724	1.488	1.7048	1.67
CH2CO		1.4457	1.18	1.7048	1.5542
CH3O	CHO	0.998	0.948	0.7173	0.771
CH2COO	CCOO	1.9031	1.728	1.27	1.6286
CH3COO		1.6764	1.42	1.27	1.4228
CH3O	CH2O	1.145	1.088	1.1434	1.6022
CH2O		0.9183	0.78	1.1434	1.2495
CH-O		0.6908	0.468	1.1434	0.8968
FCH2O		0.9183	1.1	n/a	n/a
CH3NH2	CNH2	1.5959	1.544	1.6607	1.6904
CH2NH2		1.3692	1.236	1.6607	1.3377
CHNH2		1.1417	0.924	1.6607	0.985
m-CNH2		n/a	n/a	1.6607	0.985
CH3NH	CNH	1.4337	1.244	1.368	1.4332
CH2NH		1.207	0.936	1.368	1.0805
CHNH		0.9795	0.624	1.368	0.7278

Subgroup	Main group	r	q	r Dortmund	q Dortmund
ACNH2	ACNH2	1.06	0.816	1.1849	0.8067
CH3CN CH2CN	CCN	1.8701 1.6434	1.724 1.416	1.5575 1.5575	1.5193 1.1666
COOH	COOH	1.3013	1.224	0.8	0.9215
HCOOH	COOH/ m-COOH	1.528	1.532	0.8	1.2742
CH2CL CHCL CCL	CCL	1.4654 1.238 1.0106	1.264 0.952 0.724	0.9919 0.9919 0.9919	1.3654 1.0127 0.66
CH2CL2 CHCL2 CCL2	CCL2	2.2564 2.0606 1.8016	1.988 1.684 1.448	1.8 1.8 1.8	2.5 2.1473 1.7946
CHCL3	CCL3/ m-CCL3	2.87	2.41	2.45	2.8912
CCL3	CCL3	2.6401	2.184	2.65	2.3778
CCL4	CCL4	3.39	2.91	2.618	3.1836
ACCL	ACCL	1.1562	0.844	0.5365	0.3177
CH3NO2 CH2NO2 CHNO2	CH3NO2	2.0086 1.7818 1.5544	1.868 1.56 1.248	2.644 2.5 2.887	2.5 2.304 2.241
ACNO2	ACNO2	1.4199	1.104	0.4656	0.3589
CS2	CS2	2.057	1.65	1.24	1.068
CH3N CH2N	(C)3N	1.1865 0.9597	0.94 0.632	1.0746 1.0746	1.176 0.824
HCOO	HCOO	1.242	1.188	1.9	1.8
I	I	1.264	0.992	1.076	0.9169
BR	BR	0.9492	0.832	1.209	1.4
CH3SH CH2SH CHSH CSH	CH3SH	1.877 1.651 1.425 1.199	1.676 1.368 1.06 0.752	1.289 1.535 n/a n/a	1.762 1.316 n/a n/a
CH2CH2OH *CHOHCH3* *CHOHCH2* *CH3CH2OH* *CHCH2OH*	CCOH	1.8788 1.878 1.6513 2.1055 1.6513	1.664 1.66 1.352 1.972 1.352	n/a n/a n/a n/a n/a	n/a n/a n/a n/a n/a
FURFURAL	FURFURAL	3.168	2.484	1.299	1.289
C5H5N *C5H4N* *C5H3N* AC2H2N AC2HN AC2N r1-AC2H2N r2-AC2H2N r1-AC2HN	PYRIDINE	2.9993 2.8332 2.667 n/a n/a n/a n/a n/a n/a	2.113 1.833 1.553 n/a n/a n/a n/a n/a n/a	2.5 2.8882 3.2211 1.4578 1.2393 1.0731 1.4578 1.4578 1.2393	2.1477 2.2496 2.5 0.9022 0.633 0.353 0.9022 0.9022 0.633
(CH2OH)2	DOH	2.4088	2.248	2.088	2.4
CH***C C***C	C***C	1.292	1.088	0.9214	1.3

Subgroup	Main group	r	q	r Dortmund	q Dortmund
CH***CH		1.0613 0.791	0.784 0.72	1.303 n/a	1.132 n/a
DMSO	DMSO	2.8266	2.472	3.6	2.692
ACRY	ACRY	2.3144	2.052	1.0	0.92
CL-(C=C)	CLCC	0.791	0.724	0.5229	0.7391
ACF	ACF	0.6948	0.524	0.8814	0.7269
DMF-1 DMF-2	DMF	3.0856 2.6322	2.736 2.12	2.0 2.381	2.093 1.522
CF3 CF2 CF	CF2	1.4060 1.0105 0.615	1.38 0.92 0.46	1.284 1.284 0.8215	1.266 1.098 0.5135
COO	COO	1.38	1.2	1.6	0.9
SIH3 SIH2 SIH SI	SIH2	1.6035 1.4443 1.2853 1.047	1.263 1.006 0.749 0.4099	n/a n/a n/a n/a	n/a n/a n/a n/a
SIH2O SIHO SIO	SIO	1.4838 1.303 1.1044	1.062 0.764 0.465	n/a n/a n/a	n/a n/a n/a
NMP	NMP	3.981	3.2	n/a	n/a
CCL3F CCL2F HCCL2F HCCLF CCLF2 HCCLF2 CCLF3 CCL2F2	CCLF	3.0356 2.2287 2.406 1.6493 1.8174 1.967 2.1721 2.6243	2.644 1.916 2.116 1.416 1.648 1.828 2.1 2.376	n/a n/a n/a n/a n/a n/a n/a n/a	n/a n/a n/a n/a n/a n/a n/a n/a
CONH2 CONHCH3 CONHCH2	CON	1.4515 2.1905 1.9637	1.248 1.796 1.488	n/a 1.5 1.5	n/a 1.08 1.08
CON(CH3)2 CONCH3CH2 CON(CH2)2	CON/ m-CON	2.8589 2.6322 2.4054	2.428 2.12 1.812	2.4748 2.2739 2.0767	1.9643 1.5754 1.1866
C2H5O2 C2H4O2	OCCOH	2.1226 1.8952	1.904 1.592	n/a n/a	n/a n/a
CH3S CH2S CHS	CH2S	1.613 1.3863 1.1589	1.368 1.06 0.748	n/a n/a n/a	n/a n/a n/a
MORPH	MORPH	3.474	2.796	n/a	n/a
C4H4S *C4H3S* *C4H2S*	THIOPHEN	2.8569 2.6908 2.5247	2.14 1.86 1.58	n/a n/a n/a	n/a n/a n/a
CO2†	CO2	1.3	0.982	n/a	n/a
CH4†	CH4	1.1292	1.124	n/a	n/a
N2†	N2	0.856	0.93	n/a	n/a
H2S†	H2S	1.235	1.202	n/a	n/a

Subgroup	Main group	r	q	r Dortmund	q Dortmund
H2†	H2	0.416	0.571	n/a	n/a
CO†	CO	0.711	0.828	n/a	n/a
NH3†	NH3	0.851	0.778	n/a	n/a
AR†	AR	1.177	1.116	n/a	n/a
O2†	O2	0.733	0.849	n/a	n/a
H2COCH2† H2COCH† H2COC† HCOCH† HCOC† COC†	EPOXY	1.5926 1.3652 1.1378 1.1378 0.9104 0.6829	1.32 1.008 0.78 0.696 0.468 0.24	n/a n/a n/a n/a n/a n/a	n/a n/a n/a n/a n/a n/a
HF†	HF	1.016	1.216	n/a	n/a
HI†	HI	1.393	1.208	n/a	n/a
COS†	COS	1.6785	1.316	n/a	n/a
SO2†	SO2	1.343	1.164	n/a	n/a
NO†	NO	0.716	0.62	n/a	n/a
N2O†	N2O	0.98	0.888	n/a	n/a
SF6†	SF6	2.374	2.056	n/a	n/a
HE†	HE	0.885	0.985	n/a	n/a
NE†	NE	0.886	0.986	n/a	n/a
KR†	KR	1.12	1.12	n/a	n/a
XE†	XE	1.13	1.13	n/a	n/a
HCL†	HCL	1.056	1.256	n/a	n/a
HBR†	HBR	1.058	1.258	n/a	n/a
F2†	F2	0.75	0.88	n/a	n/a
CL2†	CL2	1.53	1.44	n/a	n/a
BR2†	BR2	1.9	1.66	n/a	n/a
HCN†	HCN	1.2	1.19	n/a	n/a
NO2†	NO2	1.0	1.1	n/a	n/a
CF4†	CF4	1.78	1.82	n/a	n/a
O3†	O3	1.1	1.27	n/a	n/a
CLNO†	CLNO	1.48	1.34	n/a	n/a
C2H6	C2H6	n/a	n/a	n/a	n/a
c-CH2 c-CH c-C	c-CH2	n/a n/a n/a	n/a n/a n/a	0.7136 0.3479 0.347	0.8635 0.1071 0
c-CH2OCH2 c-CH2OC/2H c-C/2HOC/2H	c-CH2O	n/a n/a n/a	n/a n/a n/a	1.7023 1.4046 1.0413	1.8784 1.4 1.0116
c-CONCH3 c-CONCH2 c-CONCH c-CONC	c-CONC	n/a n/a n/a n/a	n/a n/a n/a n/a	3.981 3.7543 3.5268 3.2994	3.2 2.892 2.58 2.252

Subgroup	Main group	r	q	r Dortmund	q Dortmund
AC2H2S	ACS	n/a	n/a	1.7943	1.34
AC2HS		n/a	n/a	1.6282	1.06
AC2S		n/a	n/a	1.4621	0.78
r1-AC2H2S		n/a	n/a	1.7943	1.34
r2-AC2H2S		n/a	n/a	1.7943	1.34
r1-AC2HS		n/a	n/a	1.6282	1.06

Notes on the table: the identifiers shown are as used in Multiflash; they are not case-sensitive. If the user introduces an identifier for a subgroup or group which is not present in the table, it is taken to be a new user-defined subgroup or group. Multiflash allows up to 10 user-defined subgroups and 10 user-defined groups to be present at any time. Subgroups shown in italics are not generic identifiers and should be avoided. Subgroups marked with a † symbol only occur in PSRK UNIFAC; if they are present when other variants of UNIFAC are used, they will cause an error to occur.

When a particular variant of UNIFAC is invoked, the generic UNIFAC subgroups are translated to other subgroups within the actual UNIFAC model. The translation rules for original UNIFAC and PSRK UNIFAC are as follows.

c-CH2→CH2
c-CH→CH
c-C→C
s-OH→OH
t-OH→OH
c-CH2OCH2→CH2 + FCH2O
c-CH2OC/2H→½ CH2 + FCH2O
c-C/2HOC/2H→FCH2O
AC2H2S→C4H4S – 2 ACH
r1-AC2H2S→C4H3S – ACH – AC
r2-AC2H2S→C4H2S – 2 AC
AC2HS→C4H3S – 2 ACH
r1-AC2HN→C4H3S – 2 ACH
AC2S→C4H2S – 2 ACH
AC2H2N→C5H5N – 3 ACH
r1-AC2H2N→C5H4N – 2 ACH – AC
r2-AC2H2N→C5H3N – ACH – 2 AC
AC2HN→C5H4N – 3 ACH
r1-AC2HN→C5H3N – 2 ACH – AC
AC2N→C5H3N – 3 ACH
C2H6→2 CH3

For Dortmund modified UNIFAC the translation table is:

NMP→c-CONCH3 + 3 c-CH2
r1-AC2H2S→AC2H2S
r2-AC2H2S→AC2H2S
r1-AC2HS→AC2HS
r1-AC2H2N→AC2H2N
r2-AC2H2N→AC2H2N
r1-AC2HN→AC2HN
C2H6→2 CH3

User-modified UNIFAC groups and subgroups

For each definition of a UNIFAC model the user can amend the subgroup and group definitions using the UNIFACA keyword in combination with the data keyword. For example,

```
model MUNIFAC UNIFACA ORIGINAL VLE ORIGINAL MIDG
 POYNTING DATA
 SUBGROUP HCOOH 1.528 1.532 HCOOH
 SUBGROUP S-CH3 0.89 0.84 CH2
 SUBGROUP CH3 0.91 0.85 CH2
 GROUP HCOOH CH2 330 -0.1 0
 GROUP CH2 HCOOH 570 0.2 0
;;
```

Note that if a gas phase model is not required, the 6th keyword can be replaced by a * to indicate it is undefined.

After the data keyword repeated SUBGROUP and GROUP keywords can be used to modify the subgroup and group definitions used in the instance of UNIFAC referred to by the MUNIFAC identifier. The keyword SUBGROUP must always be followed by four further keywords giving the name of the subgroup in question, its *r* value, its *q* value and the name of the main group to which the subgroup belongs. The keyword GROUP must always be followed by the name of the first and second groups to which it applies followed by three coefficients *a, b, c* defining the group interaction parameter according to the relation,

$$A_{ij} = a + b\,T + c\,T^2$$

where T is the absolute temperature in kelvin. Note that group interactions between any pair of groups requires two GROUP keywords for a complete definition of A_{ij} and A_{ji}.

In the UNIFAC example above, the user has decided that the formic acid subgroup should have its own main group which will be called HCOOH. The *r* and *q* values are in fact set to the normal values for subgroup HCOOH. Because there is no standard main group called HCOOH, Multiflash will create a new user-defined main group of this name with group interaction parameters between HCOOH and all other main groups set to zero. Next the user has decided to introduce a new subgroup S-CH3 to denote a secondary methyl group which is defined as belonging to the existing CH2 main group. The existing CH3 subgroup is amended to have non-standard *r* and *q* values of 0.91 and 0.85 respectively. Finally the user has defined the group interaction parameters between HCOOH and CH2 to be linear functions of temperature.

It is possible to add approximately 100 SUBGROUP and GROUP keywords to a Unifac model definition before Multiflash's internal storage limits are reached.

Activity coefficient model for gas phase

The ACG (activity coefficient model for gas phase model) model has been added to make it possible to calculate enthalpy, entropy and heat capacity from the liquid heat capacity and heat of vaporisation instead of from the perfect gas heat capacity. The ACG model depends on a gas phase model and an activity coefficient model for the liquid phase. It is also necessary to set the pure component property CPIDLIQ = 2 for any component for which this method should be used.

part of model definition	Multiflash keyword	Comments
Model name	ACG	
vapour phase model	V_model	Model name for an equation of state model that is used to obtain the reference state vapour properties.
liquid phase model	Gex_model	Model name for an activity coefficient model that is used to obtain liquid enthalpy, entropy and heat capacity.

Other thermodynamic models for fluids

Electrolyte model

The new electrolyte model is designed to be added on to any equation of state. The models selection form allows it to be selected for use with the Advanced RKS equation and the CPA model. It represents salts as equivalent mixtures of sodium, potassium, calcium, chloride and bromide ions.

The following example sets up models MHYD1MRKSANRTLand MHYD2MRKSANRTL for hydrate structures I and II using the Electrolyte model.

```
model MRKSANRTL RKSA PSAT LDEN NRTL RKSABIP3;
model MDH LRELECTROLYTE DH BORN VIRIAL ELECBIP
ELECBIP-2 ELECBIP-3;
model MTEST LREPHCHECK;
model MADDMRKSANRTLMDH ADD ZANAL MTEST MRKSANRTL MDH;
model MHYD1MRKSANRTL HYDRATE I MRKSANRTL;
model MHYD2MRKSANRTL HYDRATE II MRKSANRTL;
```

The following example sets up models MHYD1MASSOC and MHYD2MASSOC for hydrate structures I and II using the Electrolyte and CPA model .

```
bipset ASSOCBIP 1 constant;
bipset ASSOCBIP-2 1 constant;
bipset ELECBIP 1 constant;
bipset ELECBIP-2 1 constant;
bipset ELECBIP-3 1 constant;
model MASSOC ASSOC PSAT ASSOCBIP ASSOCBIP-2;
model MDH LRELECTROLYTE DH BORN VIRIAL ELECBIP
ELECBIP-2 ELECBIP-3;
model MTEST LREPHCHECK;
model MADDMASSOCMDH ADD ZANAL MTEST MASSOC MDH;
model MHYD1MASSOC HYDRATE I MASSOC;
model MHYD2MASSOC HYDRATE II MASSOC;
```

COSTALD liquid density model

The COSTALD model is a corresponding states method for predicting the density of liquid mixtures. It is valid for liquids on the saturation line and for compressed liquids up to a reduced temperature of 0.9.

The volume of a liquid on the saturation line is defined by:

$$\frac{V^{sat}}{V^*} = V_R^{(0)}\left[1 - \omega V_R^{(1)}\right]$$

where V^{sat} is the saturated liquid volume, V^* is a characteristic volume for each substance, ω is the acentric factor and $V_R^{(0)}$ and $V_R^{(1)}$ are generalised functions of reduced temperature. In the Infochem implementation V^* is obtained by matching the saturated liquid volume stored in the databank at 298 K or a reduced temperature of 0.7, whichever is the lower.

The volume of a compressed liquid is given by:

$$\frac{V}{V^{sat}} = \left[1 - C\ln\left(\frac{B' + p}{B' + p^{sat}}\right)\right]$$

where B' is a generalised function of reduced temperature and ω, C is a generalised function of ω, and p^{sat} is the saturation pressure at the given temperature.

part of model definition	Multiflash keyword	Comments
Model name	COSTALD	
BIP data	bipset	optional, bipset is the name for a set of BIPs defined by the BIPSET command; requires 1 BIP, k_{ij}

Henry's law model for water

A specialised model for predicting solubility of light gases in water at low concentrations.

part of model definition	Multiflash keyword	Comments
Model name	HLW	
liquid phase model	L_model	model name for a model that is used to obtain the properties of pure water in the liquid phase

Thermodynamic models for solids

Gas hydrate model

The original Infochem model uses a modification of the RKS equation of state for the fluid phases plus the van der Waals and Platteeuw model for the hydrate phases. The model introduced in Multiflash 2.9 uses the CPA model for the fluid phases. The hydrate models have also been extended to include hydrate structure H in addition to structures I and II. The model can explicitly represent all the effects of the presence of inhibitors, although parameters for the CPA model are only provided for methanol, MEG and salt. CPA parameters for additional thermodynamic inhibitors will be added in future versions.

The hydrate model can be used in conjunction with either the RKSA equation of state or CPA model and the OILANDGAS BIP correlations.

part of model definition	Multiflash keyword	Comments
Model name	HYDRATE	
structure type	* or II I H	hydrate structure II hydrate structure I hydrate structure H
liquid phase model	L_model	model name for a model that is used to obtain the properties of pure water in the liquid phase

The following example sets up models MHYD1MRKSANRTL and MHYD2MRKSANRTL for hydrate structures I and II using the RKSA eos with Infochem mixing rules to get the properties of the liquid phase.

```
model MRKSANRTL RKSA PSAT LDEN NRTL RKSABIP3;
model MHYD1MRKSANRTL HYDRATE I MRKSANRTL;
model MHYD2MRKSANRTL HYDRATE II MRKSANRTL;
```

The following example sets up models MHYD1MASSOC and MHYD2MASSOC for hydrate structures I and II using the CPA model. There are two sets of BIPs required by CPA model which have to be defined before the model specification.

```
bipset ASSOCBIP 1 constant;
bipset ASSOCBIP-2 1 constant;
model MASSOC ASSOC PSAT ASSOCBIP ASSOCBIP-2;
model MHYD1MASSOC HYDRATE I MASSOC;
model MHYD2MASSOC HYDRATE II MASSOC;
```

Nucleation model

This model is an extension of the existing thermodynamic model for hydrates described above. In order to extend the nucleation model into the Multiflash program, the following enhancements to the nucleation model were made:

- The model was extended to cover the homogeneous nucleation of ice and fitted to available ice nucleation data.
- The model was generalised to cover in principle nucleation from any liquid or gas phase.
- A correction for heterogeneous nucleation was included that was matched to available hydrate nucleation data.
- An improved expression was adopted for fluid diffusion rates.
- More robust numerical methods were introduced into the program.

The nucleation model provides an estimate of the temperature or pressure at which hydrates can be realistically expected to form. The model is based on the statistical theory of nucleation in multicomponent systems. Although there are limitations and approximations involved in this approach it has the major benefit that a practical nucleation model can be incorporated within the framework of a traditional thermodynamic hydrate modelling package.

With the existing Infochem hydrate model and the nucleation model, the hydrate formation and dissociation boundaries can be predicted between which is the hydrate formation risk area.

The following example sets up the nucleation model in association with the hydrate model MHYD1MRKSANRTL and MHYD2MRKSANRTL

described above for hydrate structures I and II and the MRKSNRTL for the fluid phases.

```
model MNUCL1MHYD1MRKSANRTL BPNUC MHYD1MRKSANRTL;
model MNUCL2MHYD2MRKSANRTL BPNUC MHYD2MRKSANRTL;
```

Solid freezeout model

This model is used to calculate the thermodynamic properties of solid phases formed by freezing one or more of the components in the fluid mixture. It may be applied to any component where this may be a consideration. The freeze-out model in general is defined by:

$$\ln \phi_i = \alpha \ln \phi_i^{liq} - \left(\frac{\Delta H - T_{ref} \Delta C_p}{R} \right) \left(\frac{1}{T} - \frac{1}{T_{ref}} \right) + \frac{S_{corr}}{R} + \frac{\Delta C_p}{R} \ln \left(\frac{T}{T_{ref}} \right) - \frac{(p - p_{atm}) \Delta V}{RT}$$

For the freeze-out of pure solids, the parameters are as follows: ϕ_i is the fugacity coefficient of pure solid component i, φ_i^{liq} is the fugacity coefficient of the same component as a pure liquid at the same pressure p and temperature T (calculated from the liquid phase model associated with the freeze-out model), $\alpha = 1$, ΔH, ΔC_p and ΔV are the changes in molar enthalpy, molar heat capacity and molar volume respectively on fusion at the melting point, T_{ref} is a reference temperature which corresponds to the normal melting point when $S_{corr} = 0$ which is assumed in this case. p_{atm} is atmospheric pressure. ΔH, ΔC_p and ΔV are constants, which are normally obtained from the chosen data source.

Solid freeze-out can be used to model the solidification of compounds such as water, carbon dioxide or methane, for example in natural gases. It can also be used to model eutectics. In its general form, the freeze-out model can be applied to any solid phase of fixed composition, which must be defined. The model can for example be applied to hydrated salts such as monoethylene glycol (MEG) monohydrate or to crystalline mineral salts, i.e. scales. If $\alpha = 1$, the solid fugacity coefficient is defined relative to the liquid phase of the same composition whereas, if $\alpha = 0$, the solid fugacity coefficient is an absolute value. S_{corr} is a molar enthalpy correction factor that allows the reference temperature T_{ref} to be different from the normal melting point. For solid phases that are not pure substances, the parameters ΔH, ΔC_p ΔV, S_{corr} and T_{ref} must be defined specifically for the phase in question.

The basic pure component form of the model can be defined as follows.

part of model definition	Multiflash keyword	Comments
Model name	FREEZEOUT	
solid component	comp_name comp_no	name or serial number (in order of component definition) of the component in the pure solid phase
liquid phase model	L_model	model name for a model that is used to obtain the properties of the pure component in the liquid phase

The following example sets up a model for an ice phase (MICEMRKSANRTL) using the RKSA eos with Infochem mixing rules to get the properties of the liquid phase.

```
model MRKSANRTL rksa psat lden nrtl;
model MICEMRKSANRTL FREEZEOUT water MRKSANRTL;
```

The advanced form of the model is defined as follows:

part of model definition	Multiflash keyword	Comments
Model name	FRZA	
reference state	LIQUID or ACTIVITY	α=1 α=0
liquid phase model	L_model	Model name for a model that is used to obtain the properties of the pure component(s) in the liquid phase.
variants keyword	DATA	Used to specify model parameters for freeze-out phase.
model variants repeated (order optional)	COMPONENT VMELT HMELT SMELT CPMELT TMELT	Model parameter keywords each followed by appropriate values.

Each model variant must be followed by associated values as given in the following table.

Multiflash keyword	Associated value(s)
COMPONENT	Name or CAR number of component followed by its abundance
VMELT	ΔV
HMELT	ΔH
SMELT	S_{corr}
CPMELT	ΔC_p
TMELT	T_{ref}

This example defines a solid phase of calcium chloride dihydrate:

```
model MRKSANRTL rksa psat lden nrtl;
model MNACL.2(H2O) FRZA ACTIVITY MRKSANRTL DATA
COMPONENT Ca++ 1 COMPONENT Cl- 2 COMPONENT water 2
VMELT 1.4e-4 HMELT -34745 SMELT 103.32 CPMELT -264.89
TMELT 298.15;
```

Wax model

The model can be used to predict the wax appearance temperature and the wax deposition from crude oils. The Coutinho wax model has to be used in associated with the normal-paraffin pseudocomponents and can predict the wax appearance temperature and deposition more accurately.

Waxes are mainly formed from normal paraffins but isoparaffins and naphthenes are also present. Some waxes also have an appreciable

aromatic content. Waxes phase formation shows a very complex range of behaviour. When waxes form from crude oils, the individual fractions do not form a solid solution but they individually freeze out to form a mixture of solids, defined as a multi-solid phase.

Part of model definition	Multiflash keyword	Comments
Model name	coutinho	Infochem new wax model, called Multisolution (coutinho) wax model.
variant	Wilson or uniqac	Used together to define the couthino wax model, which can be wilson or unifac.
liquid phase model	L_model	model name for a model that is used to model the fluid phase

The following example for Coutihno wax model sets up a model for a wax phase using RKSA eos to get the properties of the liquid phase.

```
Model MRKSA RKSA PSAT LDEN VDW RKSABIP;
model MWAXMRKSA COUTINHO WILSON MRKSA;
```

Asphaltene model

Asphaltenes, heavy aromatic residues of the oil are another type of solid deposition that can occur from crude oils. They are thought to be polycyclic aromatics of very high molecular weight. They also contain nitrogen and sulphur. Asphaltenes are polar compounds and they are stabilised in crude oil by the presence of resins which are also polar compounds and have a strong tendency to associate with asphaltenes.

Part of model definition	Multiflash keyword	Comments
Model name	RAEQUIL	
variants keyword	DATA	Used to specify model parameters by users for asphaltene phase.
model variants repeated (order optional)	AAP AAE RAP RAE	Model parameter keywords each followed by appropriate values.

The following example sets up a model for gas and liquid phases using RKSA model and a model for a asphaltene phase using asphaltene association model and default model parameters. Then combine those two models together to create a general model for the phase equilibrium calculations.

```
model MREFFLUID RKSA PSAT LDEN VDW ASPHALTBIP;
model MREFASPHALTENE RAEQUIL;
model MADD ADD ZANAL MREFFLUID MREFASPHALTENE;
```

There are four model parameters. The following is the example to change and modify the model parameters.

```
model MREFASPHALTENE RAEQUIL DATA AAP value_aap
AAE value_aae RAP value_rap RAE value_rae;
```

where AAP and AAE are the keywords for asphaltene-asphaltene interaction parameters, followed by the corresponding values, their positions indicated in italics; RAP and RAE are for those for resin-asphaltene interactions. DATA marks the start of model parameters entry and must be present if model parameters are entered directly.

Viscosity models

SuperTRAPP viscosity model

The SuperTRAPP method is a predictive extended corresponding states model that uses propane as a reference fluid. It can predict the viscosity of petroleum fluids and well-defined components over a wide range of thermodynamic states from the dilute gas to the dense fluid. The basic idea behind this model is that the viscosity of a mixture can be equated to the viscosity of a hypothetical pure fluid which is then related to the viscosity of a reference fluid at a corresponding-state point (ρ_0 ,T_0). In order to improve the viscosity prediction for cycloalkanes and highly branched alkanes, the concept of mass shape factor in introduced in this method. In order to apply the method, the following are required:

- An equation of state for the reference fluid,
- Correlation for the viscosity of the reference fluid,
- The critical parameters, acentric factor and molecular weight of the fluid of the interest or for each component of the mixture of the interest.

The original method is not applicable to aqueous mixtures of alcohols, glycols or salts. The Infochem implementation of Super TRAPP model includes modification to ensure that the viscosity of aqueous solutions of methanol, ethanol MEG, DEG and TEG or salts and ions are predicted reasonably well.

Two variants of the SuperTRAPP model are implemented:

- Standard SuperTRAPP model as described above.
- Liquid viscosity fitting. An internal model parameter is adjusted to match the experimental measurements.

Overall the SuperTRAPP method is the most versatile method for viscosity predictions and its performance is generally better than the other methods available in Multiflash. We would recommend this method for oil and gas application. It is the default viscosity model for use with equations of state.

Reference: Huber, M. L. & Hanley, H.J.M. (1996) The corresponding-states principle: Dense Fluids. In J. Millat, J. H. Dymond & C. A. Nieto (Eds.), *Transport properties of Fluids: Their correlation, Prediction and Estimation*. Cambridge University Press.

part of model definition	Multiflash keyword	Comments
Model name	`SPVISC`	
model variant	`* or SPVISC LFIT`	Standard SuperTRAPP method Fits SuperTRAPP parameters to match reference viscosity for each component

Pedersen model

This is a predictive corresponding states model originally developed for oil and gas systems. It is based on accurate correlations for the viscosity and density of the reference substance which is methane. The model is applicable to both gas and liquid phases. The Infochem

implementation of the Pedersen model includes modifications to ensure that the viscosity of liquid water, methanol, ethanol, MEG, DEG and TEG and aqueous solutions of these components or salt are predicted reasonably well. We would recommend this method for oil and gas applications.

Multiflash includes two variants of the PDV model. The first uses the Infochem implementation of the original PDV method. The second variant fits parameters in the model to reproduce the saturated liquid viscosity of each component at its boiling point.

Reference: Pedersen, Fredenslund and Thomassen, *Properties of Oils and Natural Gases*, Gulf Publishing Co., (1989).

part of model definition	Multiflash keyword	Comments
Model name	PDVISC	
model variant	* or PDVISC	Standard PDV method
	LFIT	Fits PDV parameters to match reference viscosity for each component
	HEAV	Heavy fluids variant

Twu model

This is a predictive model suitable for oils. It is based on a correlation of the API nomograph for kinematic viscosity plus a mixing rule for blending oils. It is only applicable to liquids.

Multiflash includes two variants of the TWV model. The first uses the Infochem implementation of the original PDV method. The second variant fits parameters in the model to reproduce the saturated liquid viscosity of each component at its boiling point.

Reference: Twu, *Generalised method for predicting viscosities of petroleum fractions*, AIChE Journal, **32**, 2091, (1986).

part of model definition	Multiflash keyword	Comments
Model name	TWVISC	
model variant	* or TWVISC	Standard TWV method
	LFIT	Fits TWV parameters to match reference viscosity for each component

Lohrenz-Bray-Clark viscosity model

The LBC method is a predictive model for gas or liquid viscosity. It is mainly applicable to the types of components in oil and gas processing operations.

This model is a predictive model which relates gas and liquid densities to a fourth degree polynomial in reduced density. $\rho_r = \rho/\rho_c$:

$$\left[\left(\eta - \eta^*\right)\xi + 10^{-4}\right]^{1/4} = a_1 + a_2\rho_r + a_3\rho_r^2 + a_4\rho_r^3 + a_5\rho_r^4$$

where a_1, a_2, a_3, a_4 and a_5 are constants. For pure components

the viscosity reducing parameter ξ is defined by

$$\xi = T_c^{1/6} / MW^{1/2} p_c^{2/3}$$

where T_c and p_c are respectively the critical temperature and critical pressure and MW is the component molecular weight. For a mixture these properties are calculated using mole fraction averages.

In Multiflash the fluid densities are derived from any chosen equation of state, rather than the correlations proposed by Lohrenz et al. This has the advantage that there is no discontinuity in the dense phase region when moving between liquid-like and gas-like regions.

Multiflash also allows two variants of the LBC model. The first uses the original LBC method to estimate the critical volume of petroleum fractions and takes the critical volume of other components from the chosen data source. The second variant fits the critical volume of each component to reproduce the liquid viscosity at the boiling point.

It is also possible to specify a critical volume specifically for the LBC model by setting the pure component property VCLBC.

part of model definition	Multiflash keyword	Comments
Model name	LBC	
model variant	* or LBC LFIT	Standard LBC method Fits LBC parameters to match reference viscosity for each component
fluid phase model	fluid_model	model name for a model that is used to obtain the density of the mixture in a fluid phase
Variant keyword	DATA	Used to specify LBC model parameters
Model variant repeated (order optional)	A1 A2 A3 A4 A5	Model parameter keywords each followed by appropriate values.

This example sets up an eos model (MRKSA) and then defines the LBC viscosity model using MRKSA to calculate the density.

```
model MRKSA rksa psat lden;
model MLBCMRKSA LBC * MRKSA;
```

There are five model parameters. The following is an example of how to modify these model parameters.

```
model MLBCMRKSA LBC * MRKSA DATA A1 value_a1 A2
value_a2 A3 value_a3 A4 value_a4 A5 value_a5;
```

Liquid viscosity mixing rule

This method obtains the liquid mixture viscosity by applying a simple mixing rule to the pure component saturated liquid viscosities obtained from a databank. The databank specified for each component must contain a liquid viscosity correlation.

$$\ln \eta = \frac{\sum_i n_i \ln \eta_i}{\sum_i n_i}$$

where η_i is the and pure liquid viscosity for component i and n_i is the number of moles of component i in the mixture.

part of model definition	Multiflash keyword	Comments
Model name	LVS1	

Vapour viscosity mixing rule

This method obtains the vapour mixture viscosity by applying a simple mixing rule to the pure component vapour viscosities at zero density obtained from a databank. The databank specified for each component must contain a gas viscosity correlation.

$$\eta = \frac{\sum_i n_i \sqrt{M_i}\,\eta_i}{\sum_i n_i \sqrt{M_i}}$$

where M_i and η_i are the molecular weight and pure gas viscosity for component i and n_i is the number of moles of component *i* in the mixture.

part of model definition	Multiflash keyword	Comments
Model name	VVS1	

Thermal conductivity models

Chung-Lee-Starling model

The CLS method is a predictive model for gas or liquid thermal conductivity. It is mainly applicable to the types of components in oil and gas processing operations.

$$\lambda = \lambda_\kappa + \lambda_p$$

$$\lambda_\kappa = \lambda_0\left[\frac{1}{H_2} + B_i Y\right]$$

where λ_0 is a generalised function of reduced temperature, B_i is a generalised function of ω, H_2 is a generalised function of ρ_r and Y is a function of volume.

part of model definition	Multiflash keyword	Comments
Model name	CLS	
fluid phase model	fluid_model	Model name for a model that is used to obtain the density of the mixture in a fluid phase

SuperTRAPP thermal conductivity model

The STRAPP method is an extended corresponding states model that can be used for petroleum fluids, well-defined components and pure substances.

In general the thermal conductivity is defined as the sum of the internal contribution and the translational contribution. The translational contribution can be defined as three parts: the dilute gas contribution, the residual and critical enhancement contribution.

$$\lambda_x(T,\rho,\{x\}) = \lambda_x^{\text{int}}(T,\{x\}) + \lambda_x^{dilute-gas}(T,\{x\}) + \lambda_x^{res}(T,\rho,\{x\}) + \lambda_x^{Crit}(T,\rho,\{x\})$$

For a mixture, the internal contribution λ_x^{int} can be estimated from the empirical mixing and combining rules using the modified Eucken correlation for the individual components.

For the dilute gas term, the expression for the translational contribution is formulated as

$$\lambda_x^{dilute-gas}(T,\{x\}) = \frac{15\,R\eta_x^o(T)}{4\mathrm{M}_x}$$

in units of *mW/m/K*. R is the molar gas constant in units of $J/mol/K$ and M_x is the molecular weight for the mixture in unit of g/mol and the expression of $\eta_x^o(T)$ for a mixture is given as follows.

$$\eta_x^o(T) = \frac{\sum_{i=1}^{n} x_i \eta_i^o(T) M_i^{1/2}}{\sum_{i=1}^{n} x_x M_i^{1/2}} \quad \text{and} \quad M_x = \sum_{i=1}^{n} x_i M_i$$

Where η_i^o is the dilute gas viscosity given by Lucas correlation in unit of μP.

The expression of the residual contribution term is evaluated using the extended corresponding state method and is related to the thermal conductivity of a reference fluid of propane at a corresponding state, (ρ_o, T_o).

$$\lambda_x^{res}(T,\rho) = \lambda_o^{res}(T_o,\rho_o) F_\lambda$$

where the F_λ calculation requires mixing and combining rules for the mixtures.

In the current version of STRAPP in Multiflash 3.9, the critical enhancement term, λ_x^{Crit} is omitted since the critical enhancement for mixtures is usually very small.

part of model definition	Multiflash keyword	Comments
Model name	SPTHCOND	

Liquid thermal conductivity mixing rule

This method obtains the liquid mixture thermal conductivity by applying a simple mixing rule to the pure component saturated liquid thermal conductivities obtained from a databank. The databank specified for each component must contain a liquid thermal conductivity correlation.

$$\frac{1}{\lambda^2} = \frac{\sum_i n_i M_i \frac{1}{\lambda_i^2}}{\sum_i n_i M_i}$$

where M_i and λ_i are the molecular weight and pure saturated liquid thermal conductivity of component i and n_i is the number of moles of component *i* in the mixture..

part of model definition	Multiflash keyword	Comments
Model name	LTC1	

Vapour thermal conductivity mixing rule

This method obtains the vapour mixture thermal conductivity by applying a simple mixing rule to the pure component vapour thermal conductivities at zero density obtained from a databank. Requires that the databank specified for each component contains a gas thermal conductivity correlation.

$$\lambda = \frac{\sum_i n_i \sqrt{M_i} \lambda_i}{\sum_i n_i \sqrt{M_i}}$$

where M_i and λ_i are the molecular weight and pure gas thermal conductivity of component i and n_i is the number of moles of component *i* in the mixture.

part of model definition	Multiflash keyword	Comments
Model name	VTC1	

Surface tension models

Linear Gradient Theory

This method predicts the interfacial tension between two phases. The possible pairs of phases are: Liquid / Gas and Liquid / Liquid.

The model uses the difference in densities between the two phases and the energy gradient that arrives from the fact that the phases are immiscible to predict the interfacial tension.

$$\sigma = \int_{\rho_I^{ref}}^{\rho_{ref}^{II}} \sqrt{2c\left(\sum_i \rho_i\left(\mu_i(\rho) - \mu_i^{eq}\right) - \left(P(\rho) - P^{eq}\right)\right)}\, d\rho_{ref}$$

Where ρ is the density, μ is the chemical potential, P is the pressure and σ is the interfacial tension. The parameter c is called the influence parameter and is a function of temperature.

part of model definition	Multiflash keyword	Comments
Model name	LGST	
liquid phase model	EOS_model	Model name for an equation of state model that is used to obtain the density and thermodynamic properties of the mixture.

Macleod-Sugden surface tension method

The MCS method is a predictive model for surface tension based on the *parachor* stored in the databank. It is mainly applicable to the types of components in oil and gas processing operations.

$$\sigma^{1/4} = \sum P_i(\rho_l x_i - \rho_v y_i)$$

where:

P_i is the parachor for component i

ρ_l is the liquid molar density

ρ_v is the vapour molar density

x_i is the liquid mole fraction

y_i is the vapour mole fraction

part of model definition	Multiflash keyword	Comments
Model name	MCS	
liquid phase model	liquid_model	Model name for a model that is used to obtain the density of the mixture in the liquid phase

The MCS model cal also be used in 2 phase mode, where the vapour phase is also calculated by the Equation of State rather than the ideal gas equation.

part of model definition	Multiflash keyword	Comments
Model name	MCSA	
Variant	MCS1P MCS2P	Equivalent to the MCS model 2 phase variant
Equation of state model	EOS_model	Model name for a model that is used to obtain the density of the mixture in the liquid phase and gas phase, if using the 2 phase variant

Sutton

The Sutton method predicts the interfacial tension between an aqueous phase and a gas or a liquid hydrocarbon phase. The method uses the density difference between the phases and the critical pressure of the hydrocarbon phase.

$$\sigma_{hw} = \left[\frac{c_1(\rho_w - \rho_h) + c_2}{\left(\frac{T}{T_c}\right)^{a_2 + a_3 T + a_4 T^2}} \right]^{c_3}$$

where

σ_{hw} is the surface tension between water and a hydrocarbon phase

T is the temperature

T_c is the critical temperature of the hydrocarbon phase

ρ_w is the density of the aqueous phase

ρ_h is the density of the hydrocarbon phase

It is mainly applicable to the type of components found in the oil and gas processing including hydrate inhibitors.

part of model definition	Multiflash keyword	Comments
Model name	SUT	
Variant	SUT	
liquid phase model	liquid_model	Model name for a model that is used to obtain the density of the mixture in the liquid phase

Surface tension mixing rule

This method obtains the surface tension of a liquid mixture by applying a simple mixing rule to the pure component saturated liquid surface tensions obtained from a databank. Requires that the databank specified for each component contains a liquid surface tension correlation.

$$\frac{1}{\sigma} = \frac{\sum_i n_i \frac{1}{\sigma_i}}{\sum_i n_i}$$

where σ_i is the surface tension of the pure saturated liquid for component i and n_i is the number of moles of component i in the mixture.

part of model definition	Multiflash keyword	Comments
Model name	ST1	

Binary interaction parameters

BIPDATA

The BIPDATA command sets the default databank (or correlation) for binary interaction parameter data for use by thermodynamic and/or transport property models for mixtures.

The command has the following format

```
BIPDATA  datasource_name1 datasource_name2;
```

Or

```
BIPDATA  datasource_name1;
```

datasource_names may be two of the following:

datasource_name	Meaning
OILANDGAS4	latest version of Infochem correlations for BIPs for cubic eos models (PR, PRA, RKS, RKSA) and LGST model for components in oil and gas mixtures. Components included are: hydrocarbons, petroleum fractions, water, methanol, glycols, H_2S, CO_2 , N_2, and saltcomponent (for RKSA)
OILANDGAS3	old version of correlations superseded by oilandgas4 to version 2.9 in January 2000 (included for backwards compatibility).
OILANDGAS2	old version of correlations superseded by oilandgas3 in November 1997 (included for backwards compatibility)
OILANDGAS1	old version of Infochem correlations superseded by oilandgas2 in June 1994 (included for backwards compatibility)
OILANDGAS	synonym for oilandgas4
INFOBIPS	BIPs databank for VLE with NRTL and UNIQUAC models and refrigerants.
INFOLLBIPS	BIPs databank for LLE with NRTL and UNIQUAC models.
ERASE	erases (removes) currently defined datasource

The following example sets up a model called MRKS that uses the RKS equation of state with the OILANDGAS correlations for BIPs.

```
BIPDATA INFOBIPS OILANDGAS;
MODEL MRKS RKS;
```

Other binary interaction parameter data, typically containing users' own data, in one of the standard formats may be specified as follows:

```
BIPDATA  datasource_name  file_name;
```

The databank_type, associated file_name are defined in the following table.

Dataresource_name	File_name	Notes
OWNINFOBIPS	databank (.bin) file	The file must correspond to the structure and format used in Multiflash

BIPSET

The BIPSET command defines an identifier for a collection of BIP values and sets the numerical values of the BIPs. The BIPs may subsequently be associated with a mixture model for thermodynamic or transport properties.

The command has the following format

```
BIPSET  bipset_id  no_of_bips  T_dependence
Bipset_function_type bip_unit component_1
component_2  bip_values ....
```

The following table gives the valid options and settings:

command parameter	Description	Notes
bipset_id	user-defined name that will be used to refer to the collection of BIP values	any unique alphanumeric string, *e.g.* bipuser
no_of_bips	the number of BIP values that will be defined per binary pair of components. The number required is model-dependent and also depends on whether the BIPs are temperature-dependent	See the information on model definition for the number of BIPs required for each model
T_dependence	a keyword describing the T-dependence model used. May be: constant linear quadratic (constant is assumed if omitted).	The normal function used to describe temperature dependence is: $B = a_0 + a_1 T + a_2 T^2$ where B is the BIP and T is the temperature in K
Bipset_Function_Type	a keyword for describing the type of bip sets. May be: Eos activity association	Eos for EOS models activity for activity models association for CPA association models
Bip_unit	a keyword giving the dimensions of the BIPs. The available units for BIPs for activity models are J/mol, cal/mol, K, Aspen or none (denoting dimensionless). The units for eos models are all dimensionless. For association, the units are fixed as dimensionless, molar energy and molar heat capacity units respectively.	The BIPs may have units depending on the models specified. If it is omitted, the BIPs will be treated as dimensionless (or association units).
component_1 component_2	component identifiers for the BIP	These may be names or serial numbers. the number of BIP values entered must agree with the number specified above
bip_values	BIP values (real numbers)	The component identifiers and BIP values may be repeated as many times as required to specify all BIPs

The following example defines a BIP dataset called WILSONBIP2 and sets the values of two BIPs for the component pair methanol and isobutene. Because the temperature-dependence keyword is omitted the parameters are assumed to be constants.

```
bipset WILSONBIP2 2 constant activity J/mol methanol
isobutene 1.16 2.5;
```

If both BIPSET and BIPDATA are used, then the BIPSET data will be used first, then BIPs will be taken from the defined BIPDATA source and the remainder will be set to the default value(s) for the model. Remember that the user identifier for any defined BIPSET must be attached to the definition of the model.

Regarding activity methods, for the dimensionless and Aspen units options note that the actual temperature function becomes

$$b = a_0 + a_1/T + a_2 T$$

as the dimensionless BIP is actually *b=B/RT*. For the Aspen units, the NRTL α parameter is defined as a quadratic function of the Celsius temperature *t*:

$$\alpha = a_0 + a_1 t + a_2 t^2$$

It is possible to use the same BIP set for more than one model, provided the models use the same number of parameters. For instance if an RKS BIP set is provided then it could also be used with the Peng-Robinson EOS if the values are appropriate.

Troubleshooting - BIPs

Order of components

Equations of state with standard mixing rules usually have a single symmetric BIP. This means that the BIP is the same regardless of the order of components. However, activity coefficient models usually have two asymmetric BIPs. It is important that you enter **both** BIPs and in the correct cells so that the binary pair is consistent with BIPs supplied.

Number of BIPs for the model

If you are supplying BIPs in a .mfl file it is important that you define and supply the correct number of BIPs.

Wilson A, Wilson E and UNIQUAC require 2 BIPs. If you only specify 1 after the bipset name a warning message will appear;

```
Model has the wrong no. of BIPs for the existing
set
```

A similar message will appear if you only specify 2 BIPs for NRTL which requires 3.

The other possibility is that you specify the correct number of BIPs for the model, but fail to supply them all. The result will depend on the model and the default BIPs set.

If you specify NRTL with 3 BIPs and only supply 2 the default is to set the third parameter to 0.3, for NRTL VLE and to 0.2 for NRTL LLE. For other BIP sets the default value is 0. If no BIPs are specified the BIPs are read from the databank. If the databank does not hold any values the BIPs will be set to the default values.

For example if you specify UNIQUACBIP to have 2 constant BIPs and supply only one BIP value, the second BIP will be set to 0.0, and no warning message will be issued. We recommend that you always check your supplied BIPs using the Tools/BIPs option.

Naming the components

If you forget to enter the component names for a binary pair in a file before the BIP values you will upset the expected sequence of the command and the likely error message will be

```
MFL Syntax error.
```

Phase descriptors and stream types

The PD command

The PD command is used to define a *phase descriptor* (PD). The PD is a user-defined name that is used to refer to a phase. It is necessary to define a PD for all the phases that you wish Multiflash to consider when doing a flash calculation.. To exclude the formation of a particular phase type, *e.g.* gas, the corresponding PD should be omitted or erased. The `phasedescriptor` command is a synonym for `pd`.

The maximum number of PDs that may be defined is currently set at 20. The maximum number of phases that can coexist at equilibrium is limited to 7.

The command for defining a phase has the format:

```
PD  pd_id  phase_type  model_identifiers;
```

And the command for deleting an existing phase is:

```
PD  pd_id  erase;
```

The following table gives the valid options and settings:

command parameter	Description	notes
`pd_id`	user-defined name that will be used to refer to the particular instance of phase type and associated models	any unique alphanumeric string, *e.g.* `liquid1`
`phase_type`	a keyword that defines the phase type, valid settings are: condensed gas hydrate liquid multisolid solidsolution vapor vapour	1. gas, vapor and vapour are synonyms. 2. condensed means a pure solid phase. 3. Multisolid and solidsolution mean a mixed solid phase.

model_identifiers	identifiers for up to six models that will be used to evaluate the thermodynamic and transport properties of the phase. Models for the following properties may be specified in the order given: 1. fugacity (K-values) 2. Volume/density (optional) 3. Enthalpy/entropy (optional) 4. Viscosity (optional) 5. Thermal conductivity (optional) 6. surface tension (optional) 7. Nucleation model 8. Diffusivity model	1. the model identifiers are the user-defined names associated with the models (see description of MODELS command) 2. at least one thermodynamic model must be defined (the same model is then used for all thermodynamic properties) 3. transport property models need only be defined if output of these properties is required
Erase	erases (removes) the PD from the list of PDs available for Multiflash	1. the pd_id must have been previously defined 2. all information associated with the PD is lost

The streamtype command

The streamtype command is used to define a *stream type* in association with the phase descriptors (PDs) and components. The streamtype command is a synonym for st.

The command has the format:

```
ST  st_id composition nmol_1 nmol_2 …; components
comp_1&comp_alias_1 comp_2&comp_alias_2 … ; pds
pd_id_1&pd_alias_1   pd_id_2&pd_alias_2 …; ;
```

In the command there is an end-marker for keywords composition, components and pds, and the last end-marker must be present for terminating the streamtype command. If no phase descriptors are currently defined and you try to define a stream type using the command described above, you will see an error message on the screen:

```
*** ERROR     555 ***

Unrecognised entry - No phase descriptor of this name
has been defined
```

The composition field provides the flexibility to define multiple feeds into one MFL file. The amounts are assigned using the same order as the components are listed in the MFL file. If no composition is defined, the amounts for the components are taken from the amounts command.

The components are identified using either their names or component numbers *comp_1 … .* If the components keyword is omitted, all defined components are included in the stream type. For each component, an alias can be defined which is the name by which the component must be referred to when the stream type is set to be the current operative stream type. The alias can be omitted for any or all components, for example:

```
components  comp_1 comp_2 … ;
```

in which case the components will be referred to by their normal names.

The phases are designated by their phase identifiers *pd_id_1 … .* If the pds keyword is omitted, all defined phases are included in the stream type. For each phase, an alias can be defined which is the name by

which the phase must be referred to when the stream type is set to be the current operative stream type. The alias can be omitted for any or all phases, for example:

```
pds  pd_id_1 pd_id_2 … ;
```

in which case the phases will be referred to by their normal names.

The command to remove a stream type has the format:

```
ST  st_id  erase;
```

The following table gives the valid options and settings:

command parameter	Description	Notes
st_id	user-defined name that will be used to refer to a stream type	any unique alphanumeric string, *e.g.* stream1
composition	Keyword that defined compositions for the stream	
nmol_1	Component amount	Real or integer
components	keyword that defines the components for a stream type	optional
comp_1	component name or number	alphanumeric or integer
comp_alias_1	user-supplied component alias	
phasedescriptors or pds	keyword that defines the phase descriptors for a stream. type	optional
pd_id_1	phase identifier	
pd_alias_1	user-supplied phase alias	
Erase	erases (removes) the stream type from the list of those available for Multiflash	1. the st_id must have been previously defined 2. all information associated with the stream type is lost

The KEYS component command

The KEYS command is used to associate a set of *key components* to a PD. A key component helps identifying a particular phase when two or more PDs would otherwise be indistinguishable. It is not necessary to define key components unless a flash calculation needs to identify phases uniquely (*e.g.* a search for a particular phase fraction). The command has the format:

```
KEYS  pd_id  key_component_id1
key_component_id2 ... ;
```

or:

```
KEY  pd_id  not key_component_id1
not key_component_id2 ...  ;
```

pd_id is a previously-defined phase descriptor name.

key_component_id(i) is the name of the ith component used to identify the phase. The user can associate as many key components to a given PD as the maximum number of components allowed in Multiflash.

In general, the rule used is that the key components should represent a large (mass) fraction of the given phase. If the a given component name is preceded by the keyword `not` , this means that this component should be present in the minimum mass fraction.

For more flexibility, it is possible to combine “positive” and “negative” key components. For example,

```
KEY  pd_id  not key_component_id1

key_component_id2 ;
```

would identify phases that simultaneously have a large mass fraction of “key component 2” and low mass fraction of “key component 1”.

When a set of “positive” key components is defined, two criteria need to be met for a phase to be allocated to the associated phase descriptor: (1) The given phase must be the one with the highest mass fraction of the key components among all the eligible phases (i.e., phases using the same fugacity model, in the same state of matter, etc.) (2) The total mass fraction of key components must be above a given threshold which has a default value of 0.65 but that can vary when special name keywords are used for the key components.

Five special name keywords are available: ASPHALTENE , AQUEOUS , * , LIGHTEST and HEAVIEST. Only a special name keyword can be used for a given PD, and it cannot be used in combination with additional key components.

The ASPHALTENE keyword is used to identify asphaltene phases. It requires the asphaltene component to be present in a large fraction while the rest of the components in negligible fractions.

The AQUEOUS keyword includes as positive key components water as well as the most common components soluble in water (e.g., methanol, MEG, salts, etc.). In addition, it requires that the total mass fraction of aqueous components is greater than 0.5.

The * keyword is used to denote a phase which has negligible amounts of the positive key components of all the other phase descriptors. It is equivalent to specifying all the positive key components of the other phase descriptors but with the keyword `not` in front of each one of them. This keyword is useful to represent generic hydrocarbon liquids.

The keywords LIGHTEST and HEAVIEST select the lightest or heaviest component in the mixture as the single key component. The criterion used is that the heaviest component is the one with the largest acentric factor and the lightest is the one with the smallest acentric factor. Since this keyword is meant to identify the phase where the lightest(heaviest) component tends to preferentially go to, when these keywords are used the phase with the largest mass fraction of the lightest(heaviest) component is always identified with this phase descriptor as long as the total mass fraction of the key component exceeds 0.01.

Allocation of PDs with key components is always attempted before PDs with no key components. However, if when trying to allocate a PD with positive key components the requirement of minimum mass fraction is not met, then phase descriptors with no key components will be attempted to allocate. If after allocating the PDs with no key components there are still unallocated phases, Multiflash will try to use the PDs with positive key components even if the minimum mass fraction requirement is not met. Hence, note that when an insufficient number of PDs has been defined phases might be allocated to inadequate phase descriptors.

In versions previous to Multiflash 4.4, only a single key component could be associated to a PD. This was done using the command KEY. The use of this command is no longer recommended but it is still supported for compatibility reasons. However, when the KEY command is used Multiflash assumes the following: (1) If the key component is water, MF assumes that the keyword AQUEOUS was intended. (2) If the key component is preceded by the command NOT then it is assumed that the user intended the keyword * .

PVT Experiment Data

Storing PVT Experiment data

Since version 6.0, Multiflash stores experimental data that can be used for model tuning, in a different way. The exceptions were Wax and Asphaltenes experimental data, where the Match command is still used.

The new PVT experimental data command can store data for these experiments:

- Saturation points (dew and bubble points)
- Viscosity (liquid and gas)
- Constant Mass Expansion (CME)
- Constant Volume Depletion (CVD)
- Differential Liberation Experiment (DLE)
- Multi-stage separator test.

The entries for PVT experimental data are defined as follows:

```
PVTEXP exp_type [TEMPERATURE [units] value]
[STREAM steam_name]
{property [units] x_1 x_2 … ;} ; ;
```

The `TEMPERATURE` field is mandatory for experiments carried out at a fixed temperature: CME, CVD and DLE.

The units for each quantity are optional but recommended to avoid

The `STREAM` keyword tells Multiflash for which stream is the experiment referring to. It is only valid for MFL files that define multiple streams.

The property field can be repeated as many times as necessary to store all the data. It can take the following values:

- `pressures`: list of pressures in experiment
- `saturation`: list of saturation pressures.
- `OilDensity`: liquid density measurements
- `gasDensity`: gas density measurements.

- RelVol: relative volume defined as V/Vtot
- LiqVolFrac: liquid volume fraction defined as Vliq/Vsat
- YFAC: Y factor for CME
- ZFAC: Z factor for gas phase
- Compressibility
- gas_Depleted: gas depleted in CVD experiments
- Cum_gas_Depleted: cumulative gas depleted in CVD
- TWOPHASEZ: 2 phase Z factor measured in CVD
- gasVISC: gas viscosity values.
- GasGravity: gas gravity defined as Mw gas / Mw air
- GASCOMP: defined the gas composition measured in the several stages in the separator test and in DLE. Each stage is marked with the STAGE keyword.
- BO: oil volume formation factor
- BG: gas volume formation factor.
- GOR: is used for both Rs in DLE and GOR in separator test
- TLIST: list of temperatures. This is valid in saturation, viscosity and separator test experiments. Note: this keyword must be place instead of the TEMPERATURE keyword.

When an MFL created from a previous version of Multiflash is opened in version 6.0, the old experimental data commands are converted into new PVTEXP commands:

- viscosity: the old data is assumed to be liquid viscosity and viscosity experiment commands are created.
- Bubble/dew points: saturation experiment commands are created
- GOR: the GOR data is converted in a single stage separator test.
- Density: converted into CME experiments, one for each detected temperature, with density data only.

Conditions

Commands for setting conditions

State quantities

Each of the following keywords must be followed with the value of the quantity in the current input units:

keyword	notes
DENSITY	used as synonym for volume
ENTHALPY	
ENTROPY	
INTERNALENERGY	internal energy
PRESSURE	
TEMPERATURE	
VOLUME	

Amounts of components

The AMOUNTS command enters the amounts of each component in the current input units.

The command has the following format

```
AMOUNTS  x_1  x_2  x_3  ....
```

Where x_i is the amount of each component entered as a real number (with a decimal point). The amounts are defined in the order specified starting with the amount of the first component.

There are two alternative forms of the command.

```
AMOUNTS  component_id   x_n
```

This form defines amounts of components by name, *i.e.* component_id must be a valid component name and x_n is the amount of the component.

```
AMOUNTS  n   x_n
```

This form defines amounts of components by number, *i.e.* n must be an integer serial number and x_n is the amount of the nth component.

TOLAMOUNTS - amounts of components in second mixture for tolerance calculation

The TOLAMOUNTS keyword is used to set the amounts of components in a second mixture which is combined with the main mixture composition in a tolerance calculation with the FRACTOLERANCE command. The command has the same format as the AMOUNTS command except AMOUNTS is replaced by TOLAMOUNTS.

Matching wax data

The MATCH command can be used to store the wax data of wax precipitation curve or wax appearance temperatures by tuning melting temperature of petroleum fractions. The command format for matching the amount of wax phase at a T and P to the total fluid is:

```
match table waxpc overall temperatures x_1 x_2
x_3 ...;pressures x_1 x_2 x_3 ...; molefractions x_1
x_2 x_3;;;
```

If the wax data are in massfraction, the keyword `molefractions` should be replaced by `massfractions`.

Keyword `molefractions`, followed by mole fractions of the wax phase to either the total amount of the fluid or the amount of the liquid plus wax phase. If the data of the wax phase are related to the total liquid plus wax phase, the keyword `overall` should be replaced by `liquid` in the command.

The units for the temperatures and pressures in the command can be set by using `units` command shown below and have to be set before using the match command so that the values for the temperature and pressure are consistent with the units specified.

```
Units temperature unit_name_temp ;
Units pressure unit_name_press;
```

For the wax appearance temperature matching, the fractions for the wax phase at a T and P should be set to zero.

Three semicolons are required in the end of the commands.

No active matching is possible without using the GUI.

Matching asphaltene deposition point

The MATCH command can be used to store asphaltene experimental data model parameters to match the asphaltene deposition point or reservoir condition or get the default model parameters or match asphaltene titration. The command format is:

```
Fixedphase asphaltene unspecified .0 molefraction;
match asphaltene ADE;
```

The temperature and pressure to be matched are taken from the specified input conditions, *i.e.* they must have been set using the `temperature` and `pressure` commands.

If the temperature and pressure are not given from the specified input condition, the alternative way to use the MATCH the command which includes temperature and pressure. The command for matching one set of ADE is:

```
match asphaltene ADE temperature t_1 t_2 … pressure
p_1 p_2 ;
```

The match asphaltene command now accepts any number of temperature/pressure pairs for the ADE values. Any matching is done using the GUI.

The commands for storing reservoir condition or getting the default model parameters or asphaltene titration are:

```
match asphaltene reservoir temperature x_1 pressure
x_2;
```

Storing the mass fraction of heptanes in asphaltene titration under stock tank condition is available. This is an alternative way to tune the asphaltene model parameters. The command format is:

```
match asphaltene titration x_1;
```

The value `x_1` is the mass fraction of heptanes required to precipitate asphaltenes from oils.

PVT table files

Pipesim PVT files

Pipesim is a general purpose simulator for modelling fluid flow in oil and gas wells, flowlines and pipeline systems. It is a product of Baker Jardine & Associates Ltd.

Multiflash can produce a PVT data file for use by Pipesim. The file contains all the physical property information required by Pipesim. It consists of a series of flash calculations on a grid of pressure and temperature values. The information stored includes the stream composition and for each grid point: liquid volume fraction, watercut volume fraction, liquid density, gas density, gas compressibility factor, gas molecular weight, liquid viscosity, total enthalpy, total entropy, liquid heat capacity, gas heat capacity and liquid surface tension. For a complete definition of PVT files see the Pipesim documentation.

Before generating a PVT file it is necessary to define the components, compositions, models and phase descriptors. Note that models for viscosity and surface tension must be defined in addition to the usual thermodynamic models.

The command to store information needed to generate a PVT file is:

```
TABLE     PIPESIM  file_name
          PRESSURES  pressure_list;
          TEMPERATURES  temperature_list;
          TITLE  “title_text” ;
```

where:

file_name is the name of the PVT file to be written. The usual file extension is '.pvt'. If a file name has been entered previously then entering * means that it should be reused.

Pressure_list is a list of pressure values at which properties will be generated. Note that a ; is needed to end the list.

Temperature_list is a list of temperature values at which properties will be generated. Note that a ; is needed to end the list.

Title_text is the title that is written to the PVT file. It is optional. If present, it should be enclosed in quotation marks. No end marker is needed, but

A final end marker ; is necessary to terminate the complete TABLE command.

For example:

```
TABLE PIPESIM pipe5.pvt
PRESSURES 1e5, 5e5 10e5;
TEMPERATURES 350 400 450;
TITLE “Riser simulation”;
```

OLGA table generator

Olga is a general purpose transient simulator for modelling fluid flow in flowlines and pipeline systems. It is a product of Scandpower A/S.

Multiflash version 3.4 onwards can produce a PVT data file for use by Olga. The command to store information needed to generate a PVT data file is:

```
table Olga mode “File_name” title "Title_PVTData"
hydratefile “Hydrate_file_name” waxfile
“Wax_file_name” pressures  x_1  x_2  x_3 . . .;
temperatures x_1 x_2 x_3 . . .;;
```

mode is now ignored. It will be set by the GUI.

“File_name”, the file name is used to save the calculated PVT data from Multiflash which should be quoted by double quotes. It can include the full path where the file is saved.

"Title_PVTData" is used to specify the title of the data, can be omitted. If it is specified, it should be quoted by double quotes.

hydratefile is an optional keyword to signal that Multiflash should write out an Olga hydrate file. It is followed by “Hydrate_file_name”, the name of the file used to save the hydrate data. “Hydrate_file_name” should be quoted by double quotes and can include the full pathname where the file is saved.

waxfile is an optional keyword to signal that Multiflash should write out an Olga wax file. It is followed by “Wax_file_name”, the name of the file used to save the wax data. “Wax_file_name” should be quoted by double quotes and can include the full pathname where the file is saved.

pressures, followed by the values of pressure at which the physical properties are calculated.

temperatures, followed by the values of temperature at which the physical properties are calculated.

Prosper table generator

Prosper is a well performance, design and optimisation program for modelling the well configurations for the oil and gas industry. It is a product of the Petroleum Experts.

Multiflash 3.9 can produce a PVT data file for use by Prosper. The command to store information to generate a PVT data file is:

```
table prosper “File_name” title "Title_PVTData"
properties PRES GOR OFVF OVIS ODEN OCOM GFVF GVIS
ZFAC GDEN CGR WVIS WCOM; units temperature temp_unit
pressure pres_unit; temperature x_T1 pressure
```

```
x_value_p11, x_value_p12, x_value_p13…; temperature
x_T2 pressure x_value_p21, x_value_p22, x_value_p23…;
```

“File_name”, the file name is used to save the calculated PVT data from Multiflash which should be quoted by double quotes. It can include the full path where the file is saved.

"Title_PVTData" is used to specify the title of the data, can be omitted. If it is specified, it should be quoted by double quotes.

properties is a keyword to signal what properties are required for the output and should be ended with a semicolon. The maximum number of the properties in the output is 13 but the number can be less. For each property a unique keyword should be used and are listed as follows.

Keywords	Description
GOR	GOR in unit of scf/sbl
PRES	The pressure in unit of psi.
OFVF	Oil FVF in unit of RB/sbl
OVIS	Oil viscosity in unit of cPoise
ODEN	Oil density in unit of lb/ft3
OCOM	Oil compressibility in unit of 1/psi
GFVF	Gas FVF in unit of ft3/scf.
GVIS	Gas viscosity in unit of cPoise.
ZFAC	Gas Z factor
GDEN	Gas density in unit of lb/ft3.
CGR	Reservoir condensate-gas ratio in unit of bbl/MMscf.
WVIS	Water viscosity in unit of cPoise.
WCOM	Water compressibility in unit of 1/psi.

Units are used for specifying the units for both temperature and pressure, followed by a keyword, temperature with a unit, temp_unit or pressure with a unit, pres_unit. And it should be ended with a semicolon, ;.

temperatures, followed by the values of temperature at which the physical properties are calculated.

pressures, followed by the values of pressure at which the physical properties are calculated, should be finished with a semicolon. For each isotherm at a temperature, maximum of 15 different pressure values can be specified.

Other commands

The TITLE command

The TITLE command sets a descriptive title. This would typically be used to record some details or background for the problem definition stored in a Multiflash command file. The title command has the following format:

```
title “title text”;
```

The title text up to 256 characters is stored by Multiflash and can be displayed using the `show title;` command. If the title text is split over several lines, quotation marks are only needed in the beginning and the end of text as in the following example.

```
title “title text line 1
title text line 2
title text line 3”;
```

Setting Enthalpy and Entropy bases

The enthalpy and entropy can be calculated on different bases. By default Multiflash will calculate the enthalpy and entropy on the *compound* basis (see below for definition). For both properties It is also possible to use the *elements* basis and for the entropy the standard basis.

The following command can be used to set the options required:

```
Datum enthalpy h_basis entropy s_basis;
```

where `h_basis` is the identifier for the enthalpy basis and `s_basis` is the identifier for the entropy basis.

basis identifier	meaning
`compound`	enthalpy or entropy is set to zero for each compound in its perfect gas state at 298.15K and 1 bar
`elements`	enthalpy or entropy is set to zero for each element in its perfect gas state at 298.15K and 1 bar
`standard`	entropy is set to zero for each compound in the perfect crystal state at 0K

For example, the following command sets the *compound* basis for enthalpy and the *standard* basis for entropy:

```
Datum enthalpy compound entropy standard;
```

Commands for changing units

Multiflash uses separate sets of units for entering quantities (input units) and for displaying results of calculations (output units).

The commands for setting units have the following formats:

```
UNITS          property_name      unit_name
INPUTUNITS     property_name      unit_name
OUTPUTUNITS    property_name      unit_name
```

`UNITS` sets both input and output units for a property. The following table lists the property and unit name keywords. For some properties, *e.g.* thermal conductivity, the keyword recognised by Multiflash is different from the property name and is given in parentheses.

Property	Units	notes
temperature	K	
	degC	
	degF	
	degR	
pressure	Pa	*all values absolute except where indicated*
	kPa	
	Mpa	
	bar	
	barg	gauge
	atm	
	atmg	gauge
	psi	
	psig	gauge
	mmHg	
	inwg	inches water gauge
	at	
	atg	
amount	mol or mole	
	kmol	
	kg	
	g	
	lbmol	
	lb	
volume or density	mol/m3	
	kg/m3	
	lb/ft3	
	lbmol/ft3	
	mol/cm3	
	g/cm3	
	kmol/m3	
	m3/mol	
	m3/kg	
	ft3/lb	
	ft3/lbmol	
	cm3/mol	
	cm3/g	
	m3/kmol	
enthalpy internal energy (internalenergy) Gibbs energy (gibbs)	J/mol	the calorie unit is the thermo-chemical calorie defined as 4.184J exactly
	kJ/mol	
	MJ/mol	
	kJ/kg	
	cal/mol	
	kcal/mol	
	kcal/lbmol	
	BTU/lbmol	
	cal/g	
	kcal/lb	
	BTU/lb	
entropy heat capacity	J/mol/K	heat capacity units are always the same as the entropy units
	kJ/kmol/K	
	MJ/mol/K	
	GJ/kmol/K	
	J/kg/K	
	kJ/kg/K	
	cal/mol/K	
	kcal/lbmol/F	
	BTU/lbmol/F	
	kcal/lb/F	
	BTU/lb/F	
speed of sound	m/s	unit setting cannot be changed
viscosity	Pas	
	cP	centiPoise

Property	Units	notes
thermal conductivity (thcond)	W/m/K kW/m/K kJ/hr/m/K BTU/hr/ft/F kcal/hr/ft/F cal/s/cm/K kcal/hr/cm/K mW/cm/K W/cm/K mW/m/K	
surface tension	N/m mN/m dyne/cm	
Diffusivity Coefficients	m2/s cm2/s	

The SET command

The SET command is used to set a number of options that control the operation of Multiflash. The command has the format:

```
SET  option_name  option_setting
```

The following table gives the valid options and settings:

option name	operation	option settings
Amounts	displays output compositions as amounts (moles or mass). See fractions	
Fractions	displays output compositions as moles fractions or mass fractions. See amounts	
Physprops	calculation and display of physical properties of phases following equilibrium calculation	The setting may be made up of a numeric value followed by the characters A, E, F and/or T. 2 phases present, phase compositions and amounts 1 the above plus: volume, enthalpy, entropy, internal energy, Gibbs energy, average molecular weight (*default*) 2 adds Cp, Cv, speed of sound A adds activity coefficients E adds H/S/G/U relative to elements in standard states E adds fugacity coefficients Transport properties C adds thermal conductivity, V adds viscosity S add surface/interfacial tension
Nophysprops	equivalent to level 0	
Streamtype	Set the current stream type for any flash calculations.	Streamtype_id, the stream type identifier previous defined using streamtype command.

The REMOVE command

The REMOVE command is used to remove or undefine all or part of the current problem specification. The command has the format:

```
REMOVE  remove_option;
```

The following table gives the valid options:

option	Action
all	removes all databanks, components, bipsets, models, phase descriptors, and input quantities. The input and output units are redefined to the initial settings they had following processing of the mfconfig.dat file.
bipsets	removes all defined bipsets
components	removes all components and amounts
models	removes all models
pds	removes all phase descriptors
streamtypes	removes all stream types

Other related commands are:

```
PUREDATA ERASE;
COMPONENTS comp_no ERASE;
```

```
PETROFRACS comp_no ERASE;
BIPDATA ERASE;
BIPSET bipset_id ERASE;
MODEL model_id ERASE;
PD pd_id ERASE;
STREAMTYPE streamtype_id ERASE;
```

The INCLUDE command

The INCLUDE command reads in a file of Multiflash commands and processes each command. The command has the format:

```
INCLUDE  file_name;
```

the file_name must be a valid file access string for the computer system in use. The command may be abbreviated to INC . The default file extension is .mfl

Alphabetical list of selected commands

COMMAND	Purpose	definition on page
AMOUNTS	specify or change composition	9, 91
BIPDATA	specify the source databank for binary interaction parameters.	6, 79
BIPSET	set binary interaction parameters.	79
CHARDATA	Specify petroleum fraction correlations.	28,
COMPONENTS	specify components.	6, 14
DENSITY	set the input volume condition for fixed volume calculations.	9, 91
DATUM	Set the option for calculation of enthalpy and entropy.	98
ENTHALPY	set the input enthalpy condition *i.e.* for isenthalpic flash.	9, 91
ENTROPY	set the input entropy condition *i.e.* for isentropic flash.	9, 91
INCLUDE	read in a file of Multiflash commands	5, 103
INHIBITOR	add the content of inhibitors (methanol, MEG, DEG, TEG and ethanol), the salt content of aqueous streams to mixtures.	36
INPUTUNITS	change the input units	10, 98
INTERNALENERGY	set the input internal energy condition for calculations at constant internal energy	9, 91
KEYS	define the key component for a phase descriptor	8, 85
MODEL	define a model for a thermodynamic or transport property of a mixture.	6, 39
OUTPUTUNITS	change the output units	10, 98
PD	set up a phase descriptor for any phase (abbreviation for PHASEDESCRIPTOR)	7, 83
PETROFRACS	define a petroleum fraction pseudocomponent	29
PHASEDESCRIPTOR	set up a phase descriptor for any phase c.f. PD	83
PRESSURE	set the pressure	9, 91
PUREDATA	specify the data source for pure component data for components.	5, 13
PVTANALYSIS	Allows the entry of experimental PVT analysis data to specify an input stream.	30,
REMOVE	removes (part of) the current problem and allows the user to specify a new problem	10, 102
SALINITY	converts the measured salt compositions into an equivalent amount of a salt pseudocomponent or ion ratios for freezing point depression or hydrate inhibition calculations.	37
SET	set a configuration option for the program	101
STREAMTYPE	Define a stream type	84
TABLE	specify a group of calculations	95
TEMPERATURE	set the temperature	9, 91
TITLE	store descriptive title	98
TOLAMOUNTS	specifies component amounts for a second stream for use in a tolerance calculation	92
UNITS	sets both input and output units	10, 98
VOLUME	set the input volume condition for fixed volume calculations.	9, 91
,	the comma can be used is used to separate keywords as an alternative to a space.	

COMMAND	Purpose	definition on page
;	the semicolon is the end-of-command marker. It terminates a command or separates multiple commands.	
#	comment character for input files. Any text following the comments character is disregarded.	