

---

**DWSIM - Open Source Chemical Process Simulator**  
**User Guide**

---

**Version 6.2 Update 2**

October 2020

## **License**

DWSIM is released under the GNU General Public License (GPL) version 3.

## **Contact Information**

**Developer: Daniel Medeiros**

Website 1: <http://dwsim.inforside.com.br>

Website 2: <http://www.sourceforge.net/projects/dwsim>

E-mail: dwsim@inforside.com.br

# Contents

<b>I Introduction</b>	<b>6</b>
<b>II Classic User Interface (Classic UI)</b>	<b>10</b>
<b>1 Welcome Screen</b>	<b>10</b>
<b>2 Simulation</b>	<b>11</b>
2.1 User Interface . . . . .	11
2.2 Configuration . . . . .	11
2.2.1 Components/Compounds . . . . .	12
2.2.2 Basis . . . . .	13
2.2.3 Systems of Units . . . . .	17
2.2.4 Miscelaneous . . . . .	18
2.2.5 Property Tables . . . . .	19
2.3 Process modeling (Flowsheeting) . . . . .	20
2.3.1 Inserting Flowsheet Objects . . . . .	22
2.3.2 Process data management . . . . .	28
2.3.3 Undo/Redo actions . . . . .	30
2.3.4 Cut/Copy/Paste objects . . . . .	32
2.3.5 Simulation . . . . .	32
2.3.6 Results . . . . .	33
2.4 Sensitivity Analysis . . . . .	34
2.5 Flowsheet Optimization . . . . .	35
2.6 Utilities . . . . .	38
2.7 Chemical Reactions . . . . .	42
2.8 Characterization of Petroleum Fractions . . . . .	44
<b>3 Compound Creator</b>	<b>46</b>
<b>4 Data Regression</b>	<b>46</b>
<b>5 General Settings</b>	<b>47</b>
5.1 Solver . . . . .	47
5.2 Flowsheet . . . . .	47
5.2.1 Cut/Copy/Paste Flowsheet Objects . . . . .	47
5.2.2 Undo/Redo . . . . .	47
5.2.3 Object Editors . . . . .	47
5.3 Datasets . . . . .	48
5.4 Backup . . . . .	48
5.5 Other . . . . .	48
5.5.1 Messages . . . . .	48
5.5.2 Debug mode . . . . .	48

5.5.3 UI Language . . . . .	48
5.5.4 CAPE-OPEN . . . . .	48
5.5.5 Compound Constant Properties . . . . .	48
5.5.6 DWSIM/Octave Bridge Settings . . . . .	48
5.5.7 DWSIM/Python Bridge Settings . . . . .	49
<b>III Cross-Platform User Interface (CPUI)</b>	<b>50</b>
<b>6 Main Interface</b>	<b>50</b>
<b>7 Configuring a Simulation</b>	<b>51</b>
7.1 Components/Compounds . . . . .	51
7.2 Property Packages . . . . .	52
7.3 Flash Algorithms . . . . .	55
7.4 Systems of Units . . . . .	56
<b>8 Process Modeling (Flowsheeting)</b>	<b>58</b>
8.1 Inserting Flowsheet Objects . . . . .	58
8.2 Connecting/Disconnecting objects . . . . .	61
8.3 Process data management . . . . .	61
8.3.1 Entering process data . . . . .	61
8.4 Running a Simulation . . . . .	62
8.5 Viewing Results . . . . .	63
8.6 Flowsheet Utilities . . . . .	64
8.6.1 Pure Component Properties . . . . .	64
8.6.2 Phase Envelope . . . . .	65
8.6.3 Binary Envelope . . . . .	66
<b>IV Unit Operation Models</b>	<b>68</b>
<b>9 Streams</b>	<b>68</b>
9.1 Material Stream . . . . .	68
9.1.1 Calculation Method . . . . .	69
9.1.2 Output parameters . . . . .	69
9.2 Energy Stream . . . . .	69
<b>10 Unit Operations</b>	<b>70</b>
10.1 Mixer . . . . .	70
10.1.1 Splitter . . . . .	70
10.2 Separator Vessel . . . . .	70
10.3 Tank . . . . .	71
10.4 Pipe Segment . . . . .	71
10.5 Valve . . . . .	72
10.6 Pump . . . . .	73

10.7 Compressor/Expander . . . . .	74
10.8 Heater . . . . .	75
10.9 Cooler . . . . .	76
10.10 Shortcut Column . . . . .	76
10.11 Rigorous Column (Distillation / Absorption) . . . . .	77
10.12 Heat Exchanger . . . . .	80
10.13 Component Separator . . . . .	81
10.14 Orifice Plate . . . . .	82
10.15 Custom Unit Operation . . . . .	82
10.16 Solids Separator . . . . .	83
10.16.1 Continuous Cake Filter . . . . .	83
10.17 Excel Unit Operation . . . . .	84
10.18 Flowsheet Unit Operation . . . . .	85
10.19 Reactors . . . . .	86
10.20 Logical Operations . . . . .	87
10.20.1 Recycle . . . . .	87
10.20.2 Energy Recycle . . . . .	88
10.20.3 Adjust . . . . .	88
10.20.4 Specification . . . . .	88
<b>V Dynamic Modeling and Simulation</b>	<b>89</b>
<b>11 Introduction</b>	<b>89</b>
<b>12 Dynamic Simulation Structure and Configuration</b>	<b>90</b>
12.1 Dynamic Model Setup . . . . .	90
12.2 Event Sets . . . . .	91
12.3 Cause-and-Effect Matrices . . . . .	91
12.4 Integrators . . . . .	91
12.4.1 Monitored Variables . . . . .	91
12.5 Schedules . . . . .	91
<b>13 Dynamic Simulation Flowsheet Blocks</b>	<b>91</b>
13.1 Analog Gauge . . . . .	91
13.2 Digital Gauge . . . . .	91
13.3 Level Gauge . . . . .	92
13.4 PID Controller . . . . .	92
13.4.1 Introduction . . . . .	92
13.4.2 Fundamental Operation . . . . .	92
13.4.3 Mathematical form . . . . .	94
13.4.4 Selective use of control terms . . . . .	94
13.4.5 Overview of tuning methods . . . . .	94
13.4.6 DWSIM Implementation . . . . .	95

13.5 Input Box . . . . .	97
13.6 Switch . . . . .	97
<b>14 Dynamic Simulation Tools</b>	<b>97</b>
14.1 Control Panel Mode . . . . .	97
14.2 PID Controller Tuning . . . . .	97
<b>15 Supported Unit Operations</b>	<b>98</b>
<b>VI Technical Basis: Methods and Procedures</b>	<b>100</b>
<b>16 Introduction</b>	<b>100</b>
<b>17 Thermodynamic Properties</b>	<b>101</b>
17.1 Phase Equilibria Calculation . . . . .	101
17.1.1 Fugacity Coefficient calculation models . . . . .	102
17.1.2 Chao-Seader and Grayson-Streed models . . . . .	105
17.1.3 Calculation models for the liquid phase activity coefficient . . . . .	106
17.1.4 Models for Aqueous Electrolyte Systems . . . . .	111
17.2 Enthalpy, Entropy and Heat Capacities . . . . .	113
17.3 Speed of Sound . . . . .	116
17.4 Joule-Thomson Coefficient . . . . .	116
<b>18 Transport Properties</b>	<b>117</b>
18.1 Density . . . . .	117
18.2 Viscosity . . . . .	119
18.3 Surface Tension . . . . .	121
18.4 Isothermal Compressibility . . . . .	121
18.5 Bulk Modulus . . . . .	121
<b>19 Thermal Properties</b>	<b>122</b>
19.1 Thermal Conductivity . . . . .	122
<b>20 Aqueous Solution Properties</b>	<b>124</b>
20.1 Mean salt activity coefficient . . . . .	124
20.2 Osmotic coefficient . . . . .	124
20.3 Freezing point depression . . . . .	124
<b>21 Specialized Models / Property Packages</b>	<b>124</b>
21.1 IAPWS-IF97 Steam Tables . . . . .	124
21.2 IAPWS-08 Seawater . . . . .	125
21.3 Black-Oil . . . . .	126
21.4 CoolProp . . . . .	127
21.5 Sour Water . . . . .	127

<b>22 Reactions</b>	<b>128</b>
22.1 Conversion Reaction . . . . .	129
22.2 Equilibrium Reaction . . . . .	129
22.2.1 Solution method . . . . .	129
22.3 Kinetic Reaction . . . . .	130
<b>23 Property Estimation Methods</b>	<b>131</b>
23.1 Petroleum Fractions . . . . .	131
23.1.1 Molecular weight . . . . .	131
23.1.2 Specific Gravity . . . . .	132
23.1.3 Critical Properties . . . . .	132
23.1.4 Acentric Factor . . . . .	133
23.1.5 Vapor Pressure . . . . .	133
23.1.6 Viscosity . . . . .	134
23.2 Hypothetical Components . . . . .	134
<b>24 Other Properties</b>	<b>135</b>
24.1 True Critical Point . . . . .	135
24.2 Natural Gas Hydrates . . . . .	136
24.2.1 Modified van der Waals and Platteeuw (Parrish and Prausnitz) method . . . . .	136
24.2.2 Klauda and Sandler . . . . .	137
24.2.3 Chen and Guo . . . . .	138
24.3 Petroleum Cold Flow Properties . . . . .	139
24.3.1 Refraction Index . . . . .	139
24.3.2 Flash Point . . . . .	139
24.3.3 Pour Point . . . . .	139
24.3.4 Freezing Point . . . . .	140
24.3.5 Cloud Point . . . . .	140
24.3.6 Cetane Index . . . . .	140
24.4 Chao-Seader Parameters . . . . .	140
<b>VII How-To Tutorials</b>	<b>141</b>
<b>25 Methane Steam Reforming</b>	<b>141</b>
25.1 Introduction . . . . .	141
25.2 Background . . . . .	141
25.3 Problem Statement . . . . .	143
25.4 DWSIM Model (Classic UI) . . . . .	143
25.5 DWSIM Model (Cross-Platform UI) . . . . .	151
<b>26 Extractive Distillation</b>	<b>158</b>
26.1 Introduction . . . . .	158
26.2 Background . . . . .	158
26.3 DWSIM Model (Classic UI) . . . . .	159

---

26.4 DWSIM Model (Cross-Platform UI) . . . . .	167
<b>27 Flowsheet Control with Python Scripts</b>	<b>178</b>
27.1 Introduction . . . . .	178
27.2 DWSIM Model (Classic UI) . . . . .	178
27.3 DWSIM Model (Cross-Platform UI) . . . . .	182
<b>28 Basic Dynamic Simulation Tutorial</b>	<b>185</b>
28.1 Introduction . . . . .	185
28.2 DWSIM Model (Classic UI) . . . . .	185
28.2.1 Model Building . . . . .	185
28.2.2 Dynamic Simulation . . . . .	188
28.2.3 Adding a PID Controller . . . . .	188
28.2.4 PID Controller Tuning . . . . .	189
28.3 Real-Time Mode . . . . .	190
<b>References</b>	<b>197</b>

# Part I.

## Introduction

This document gives a detailed description about how to setup, run, modify and view results of a basic process simulation in DWSIM. The document is organized according to the sequence of execution of a simulation. Each step/task is explained with the help of images and descriptions of the associated windows.

DWSIM has two Graphical User Interfaces: Classic UI and Cross-Platform UI.

- The Classic UI is based on the **Windows Forms** graphical class library (link), which has been used since the initial versions of DWSIM. The Windows Forms graphical library was created for Windows applications. A port of this library exists for Linux, but it has some issues and, since it mimics the Windows look-and-feel, it doesn't look "native" on systems other than Windows:

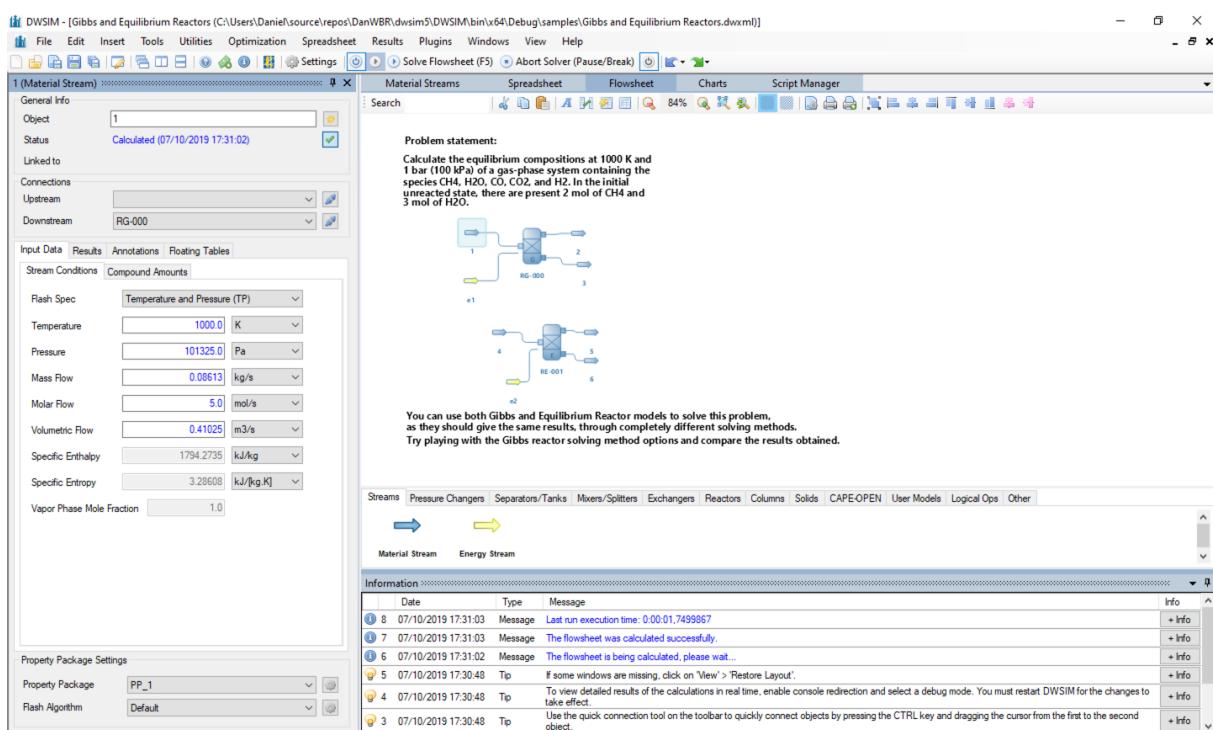


Figure 1: Classic UI on Windows 10 + .NET Framework 4.8.

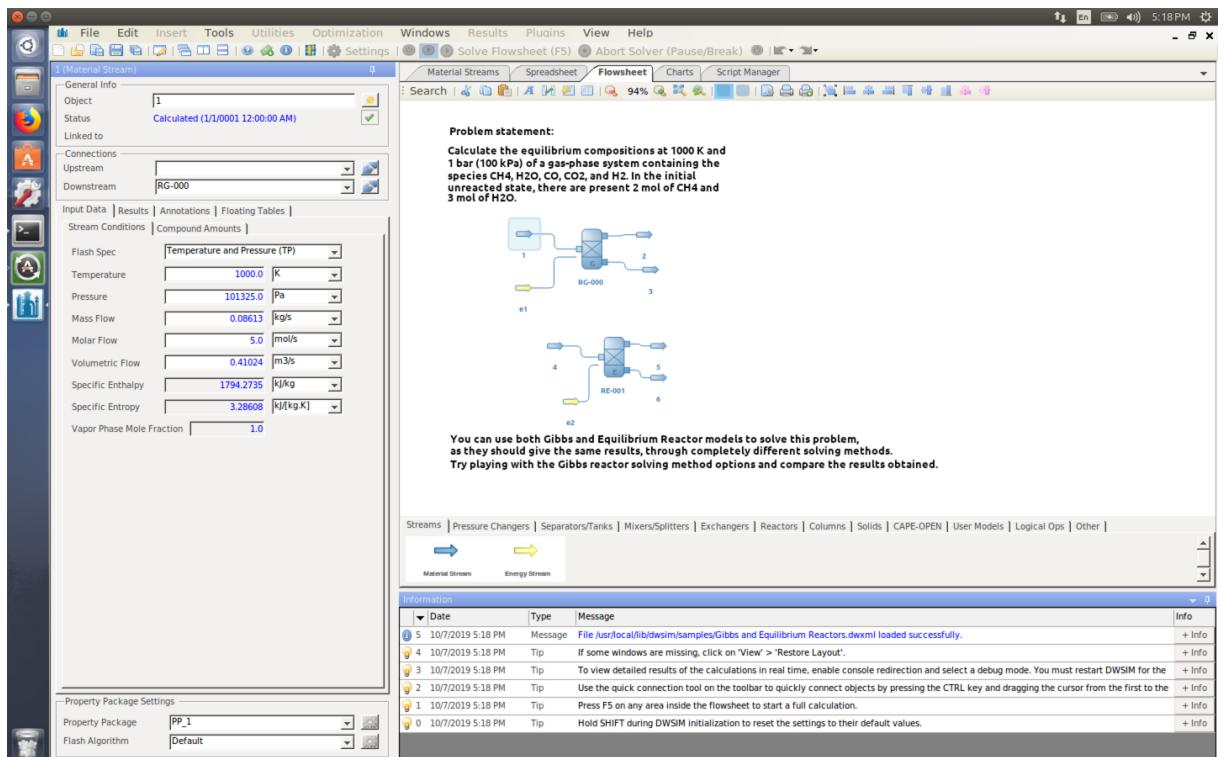


Figure 2: Classic UI on Ubuntu Linux 16.04 + Mono 5.10.

→ The Cross-Platform UI is based on a library called [Eto.Forms](#) (link), which is a cross-platform graphical class library, supporting Windows, Linux, macOS and some mobile systems. This UI was created from scratch using the C# language, and looks native when executed on systems other than Windows, since it runs under the *GTK* (link) backend on Linux and *Cocoa* (link) on macOS. On Windows, the Cross-Platform UI runs under the Windows Presentation Foundation backend, though it is recommended to use the Classic UI on Windows systems, since it is more stable and reliable due to the fact that it has been in development for much more time.

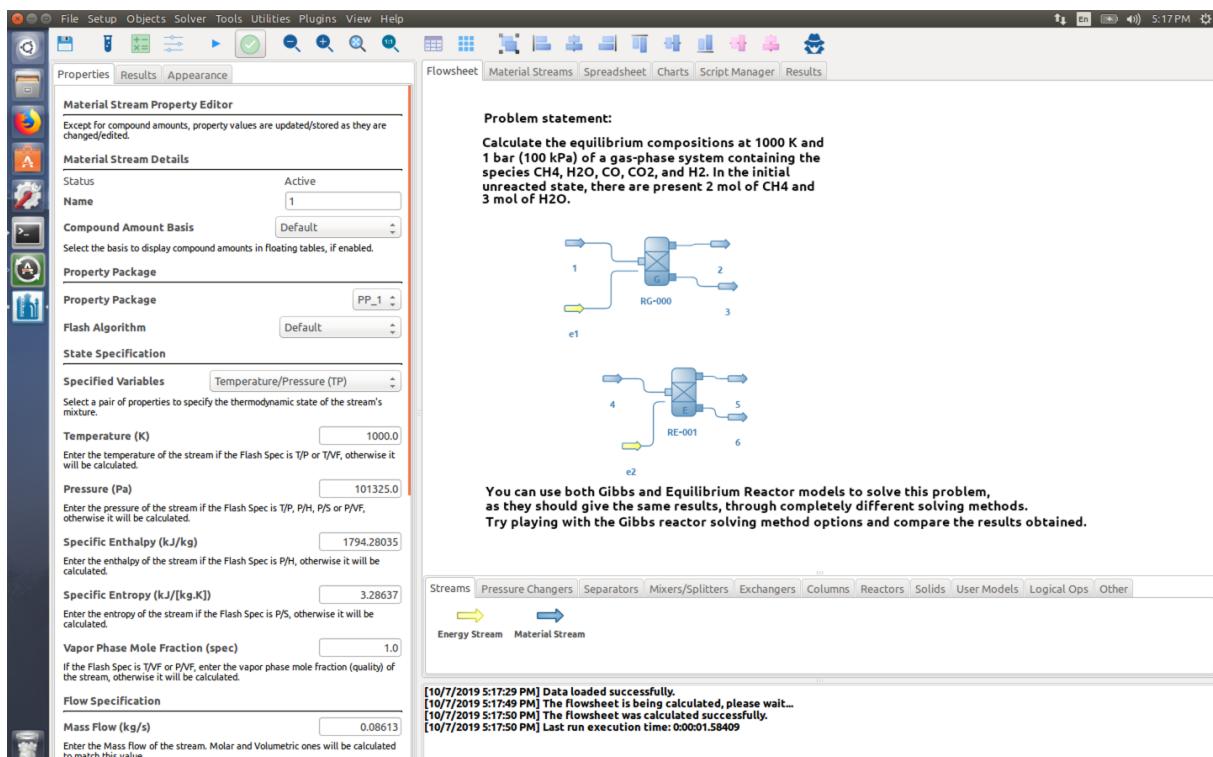


Figure 3: Cross-Platform UI on Linux 16.04 + Mono 5.10 (GTK backend).

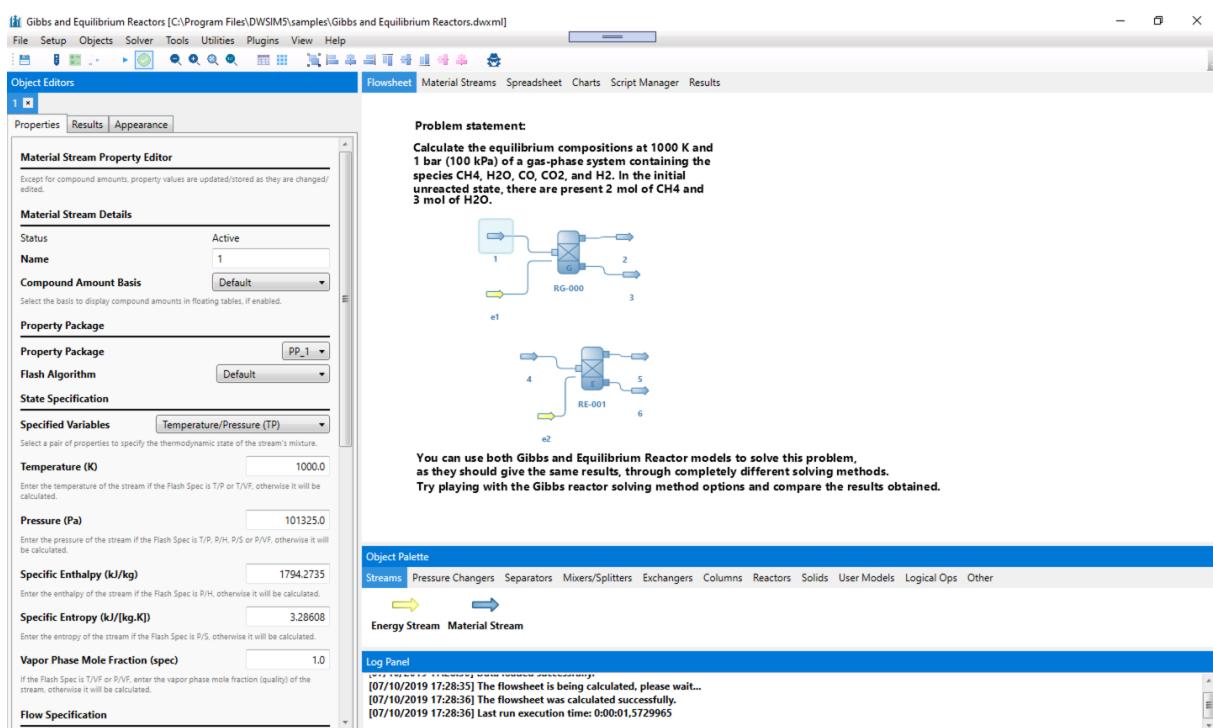


Figure 4: Cross-Platform UI on Windows 10 (WPF backend).

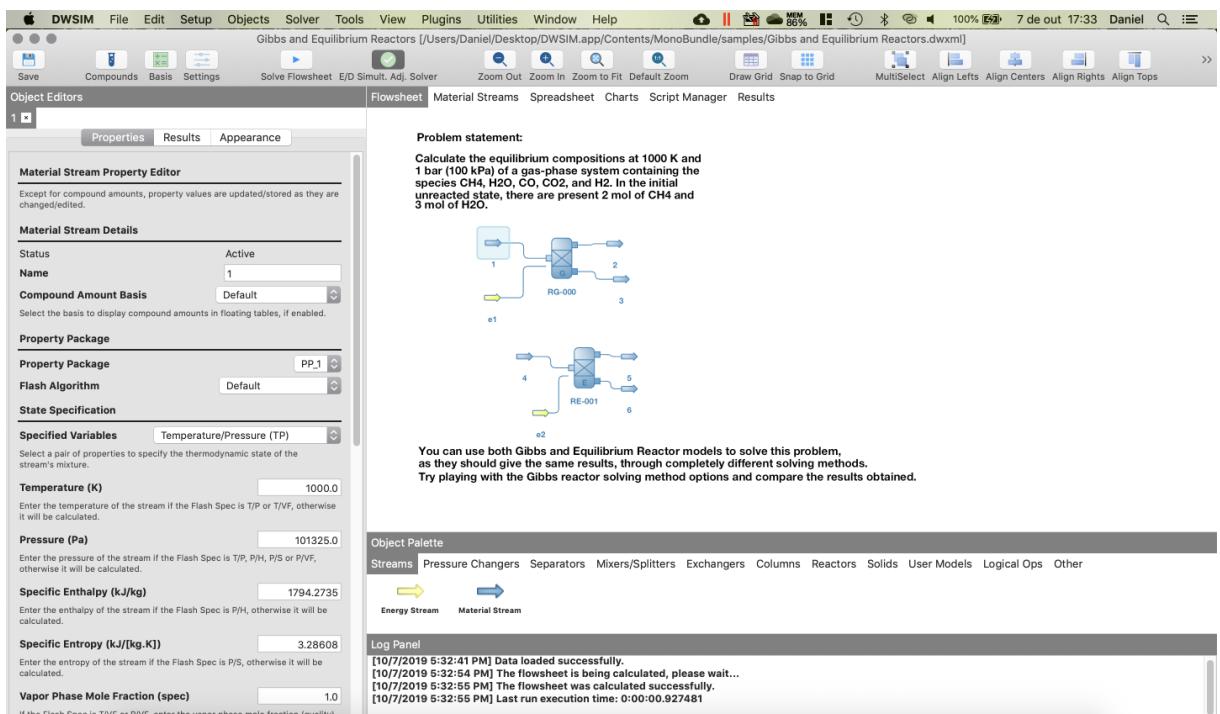


Figure 5: Cross-Platform UI on macOS Mojave (Cocoa backend).

## Part II.

# Classic User Interface (Classic UI)

### 1. Welcome Screen

When DWSIM is opened, the welcome screen is shown (Figure 6):

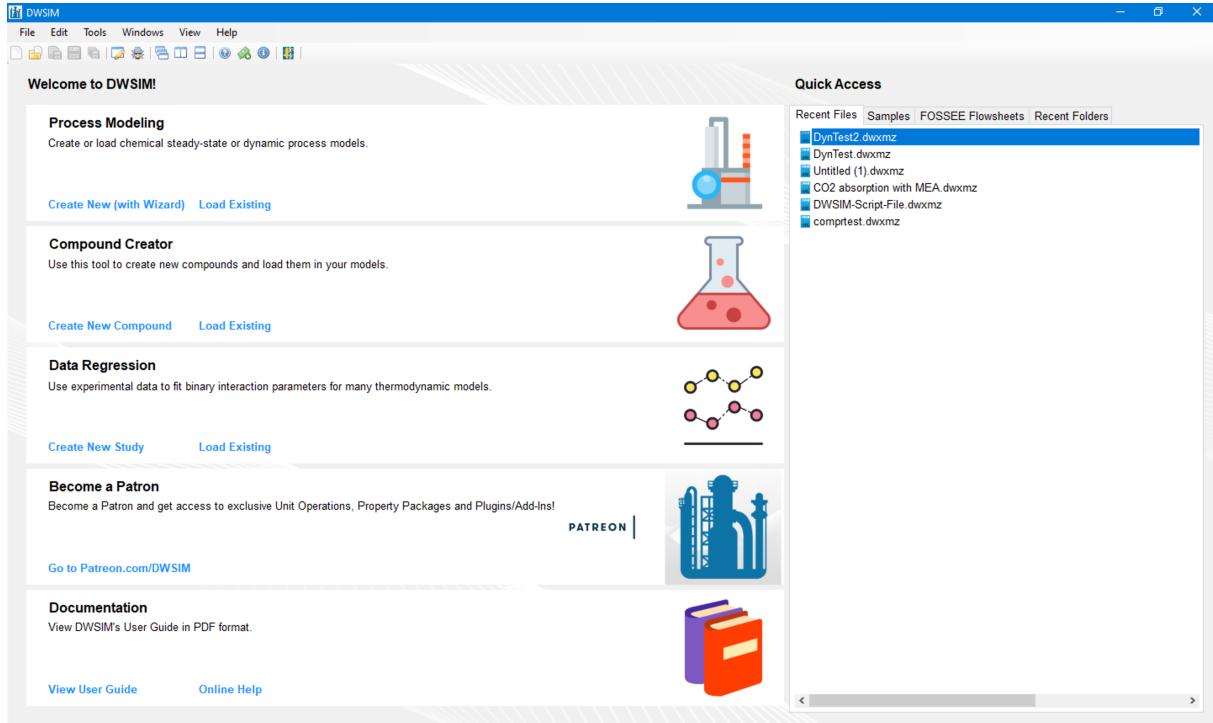


Figure 6: DWSIM's welcome screen.

The welcome screen provides the user with shortcuts to open existing simulations, create new ones, create new compound creator and data regression cases and open the samples folder.

The following items are displayed on DWSIM's main window:

- **Menu bar**, with buttons to open/save/create simulations, component creator and data regression cases, configure the active simulation, general preferences, launch tools, configure the child windows view mode, etc.;
- **Button strip**, to open, save and create new steady-state simulations, component creator and data regression cases.

There are various ways to access the most commonly operations with simulation files and component creator/data regression cases - open, save and create. In the next sections you will be guided through some necessary steps to create and configure a steady-state simulation, a compound creator and/or a data regression case.

## 2. Simulation

### 2.1. User Interface

The "Create a new steady-state simulation" button in the welcome window can be used to create a new simulation. After the simulation is created, the **configuration window** (Figure 8) is shown. The simulation configuration interface consists in a tabbed window:

- **Compounds** - Add or remove compounds to/from the simulation and petroleum fractions (pseudo-compounents) utilities.
- **Basis** - Property Package configuration, phase equilibrium flash algorithm selection and other advanced thermodynamic model settings.
- **System of Units** - Management of Systems of Units.
- **Miscelaneous** - Simulation info (title, author and description), number formatting and password settings.
- **Property Tables** - Definition of objects properties to be shown on flowsheet floating tables.

### 2.2. Configuration

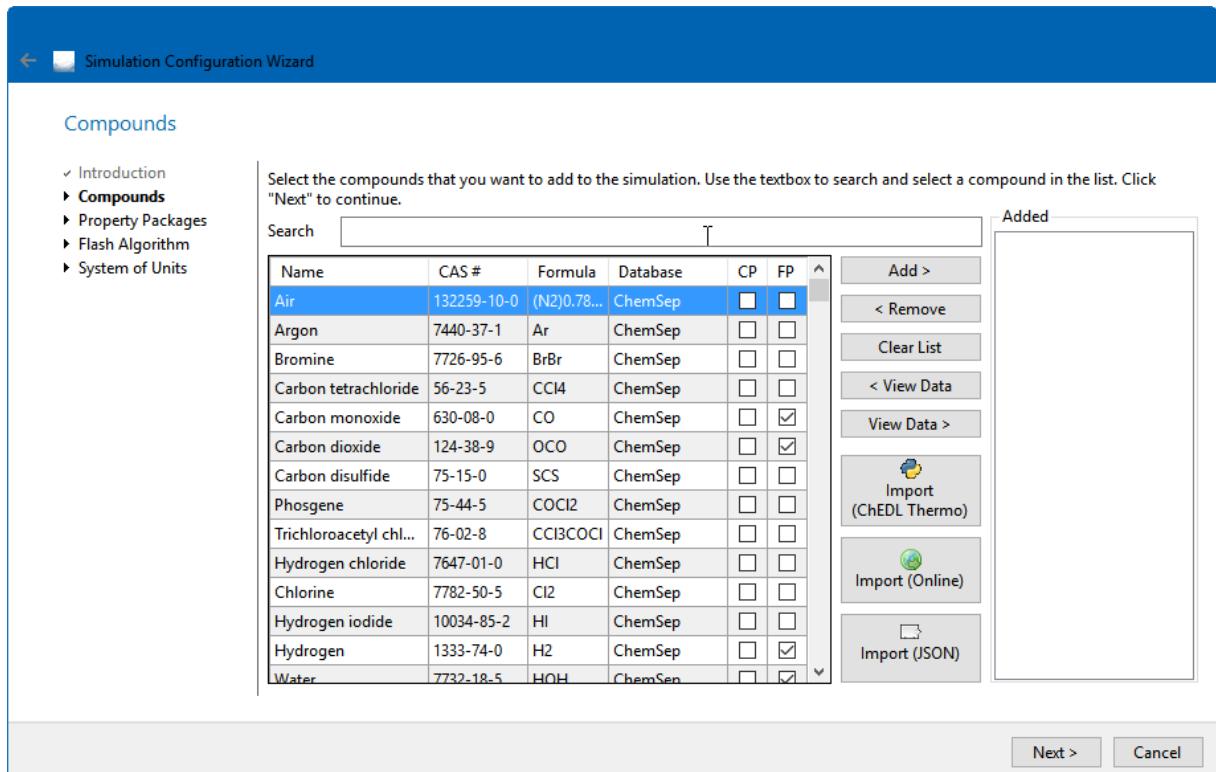


Figure 7: Simulation Configuration Wizard.

Since DWSIM 3.3, a new Simulation Configuration Wizard (Figure 7) is opened as soon as a new simulation is created, and will display the interfaces described in the following sections in a more streamlined

way. The older simulation configuration window can be accessed anytime during the simulation or through a button located in the first page of the config wizard.

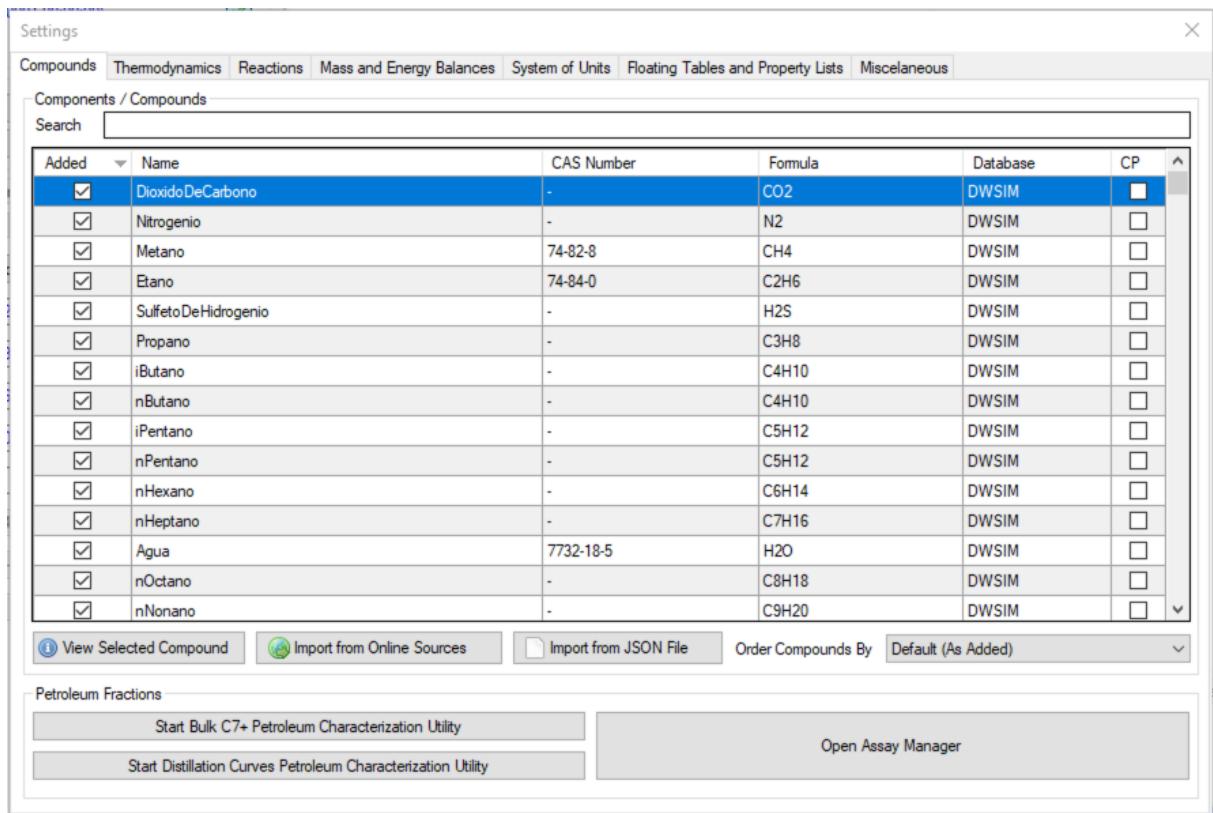


Figure 8: Simulation Configuration window.

The simulation configuration window (Figure 8) is the interface where all the functions for configuration and personalization of a simulation in DWSIM can be found. In this window, the user can manage the simulation components, the property package (thermodynamic model), configure the reactions environment, units system and number format, among other options.



The simulation configuration window can be accessed anytime when a simulation is opened in DWSIM. The changes made through it have immediate effect on the simulation.

### 2.2.1. Components/Compounds

There are two essential information required by DWSIM in order to correctly start a simulation. The first refers to the available **components** (or **compounds**). DWSIM comes with six default compound databases (DWSIM, ChemSep, Biodiesel, CoolProp, ChEDL and Electrolytes), with a total of more than 1500 compounds available for your simulation.

To add a compound to the simulation, select it from the list on the left and click on **Add >**. To remove an added compound, select it on the right-hand list and click < **Remove**. To view the data from a compound from a list, click on the appropriate **View Data** button.

DWSIM also features full compound data importing from **Online Sources** or from **JSON files**, using

the appropriate buttons on the Simulation Configuration Wizard or on the Simulation Settings panel. If you manage to find a compound from these sources with a minimum set of data, they can be added directly to the simulation without further action.

**JSON files** are exported from the Compound Creator utility or from the Pure Compound Property Viewer tool.

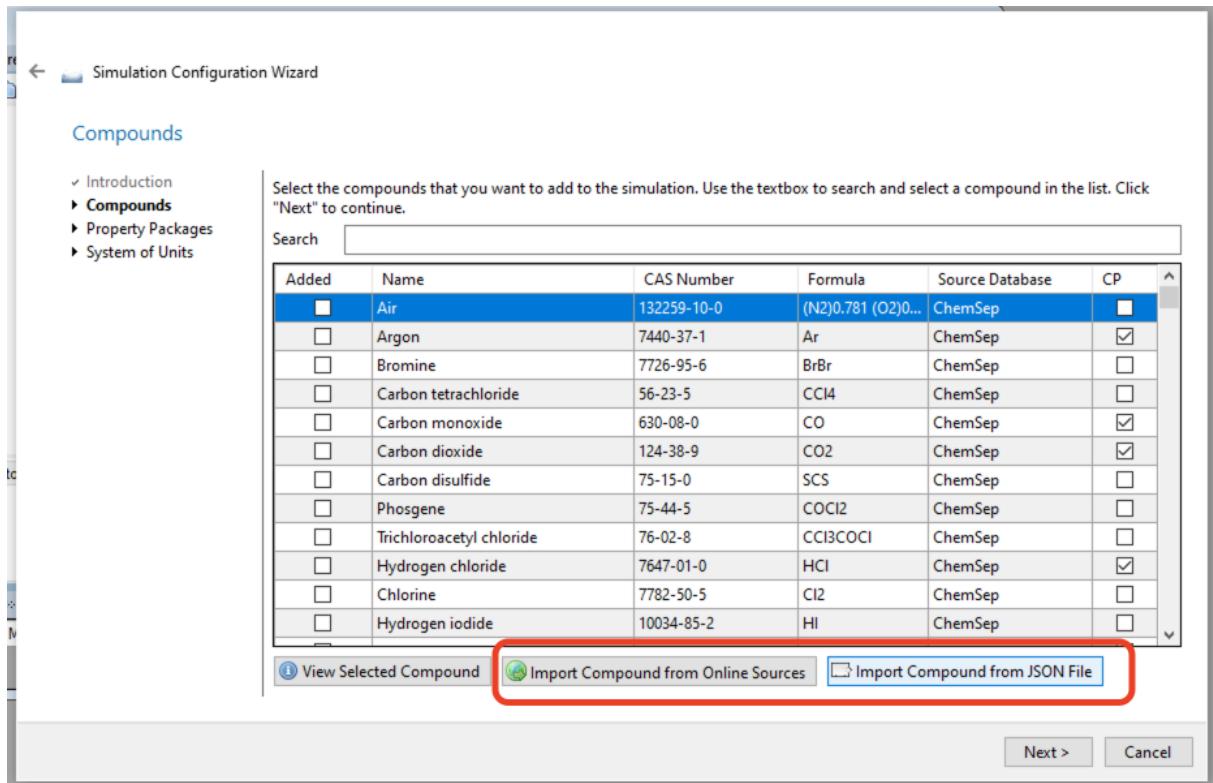


Figure 9: Compound Data Importing tools.

### 2.2.2. Basis

**2.2.2.1. Property Packages** The Property Package consists in a set of methods and models for the calculation of physical and chemical properties of material streams in the simulation. It is composed of a thermodynamic model - an equation of state or a hybrid model - and methods for property calculation, like the surface tension of the liquid phase. The figure 10 shows the interface for configuration of the property package.

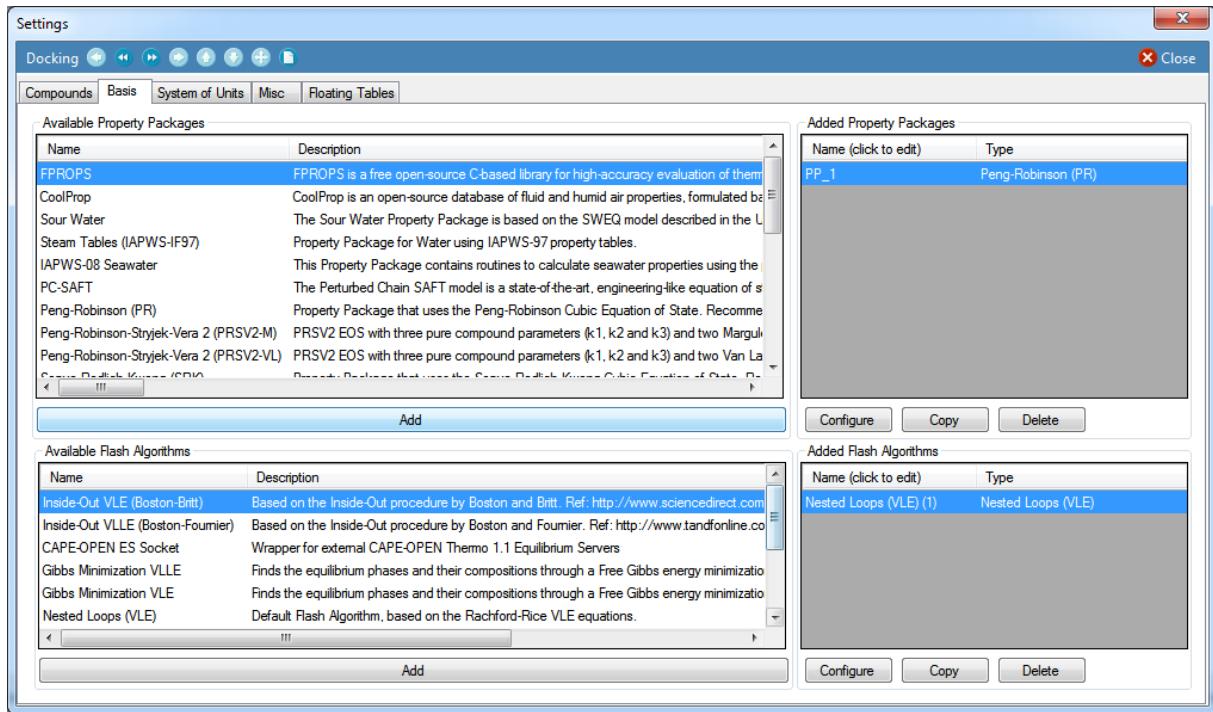


Figure 10: Property Package configuration interface.

If the selected property package has any editable property, the "Configure" button becomes clickable and the user can click on it to show the property package configuration window.

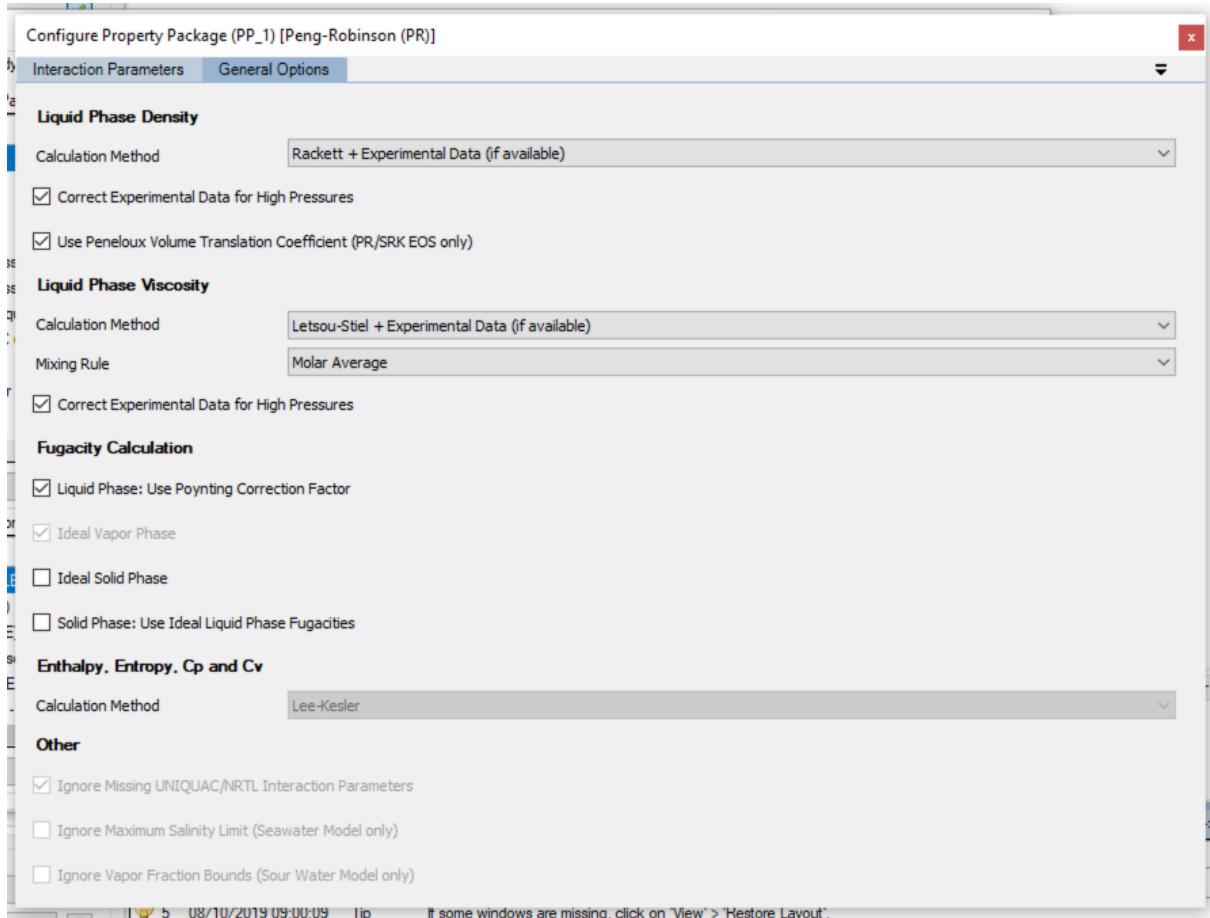


Figure 11: Property package configuration window (1).

**Property Package configuration options** Some Property Packages have extra configuration options in order to allow a deeper control of the thermodynamic calculations for the user. They are:

→ *Use Peneloux Volume Translation correction (PR/SRK EOS only)*

This option is available for PR and SRK Property Packages. It enables correction of EOS-calculated densities by the inclusion of a correction factor named *volume translation coefficient*. This option will be effective only if the EOS is selected as the calculation method for Liquid Density.

→ *Ignore maximum salinity limit (IAPWS-08 Seawater Property Package only)*

Ignores the maximum supported salinity value (0.12 kg/kg) for calculations and doesn't display any warnings. Use 0 to disable, 1 to enable this option. If enabled, the calculated salinity will be sent directly to the property calculation routines without further check. If disabled, the maximum value of 0.12 will be used if the calculated salinity is higher, and a warning message will be displayed in the flowsheet log window.

**2.2.2.2. Flash Algorithms** The Flash Algorithms in DWSIM are the components responsible for determining a particular set of phases at thermodynamic equilibrium, their amounts (and the amounts of the compounds on each phase) at the specified conditions like Temperature, Pressure, Total Enthalpy and Total Entropy. Some Flash Algorithms are capable of predicting equilibrium between one vapor and one

liquid phase, while others support another co-existing liquid and/or solid phase. As the amount of phases considered in equilibrium increases, the calculation time/complexity also increases while the results' reliability decreases. Some flash algorithms are more capable/reliable than others, depending on the mixture for which the flash calculation request is being requested. DWSIM features a selection of flash algorithms that are capable of calculating VLE, VLLE and SLE. They are:

1. **Nested Loops (VLE)**: recommended for the vast majority of VLE systems;
2. **Nested Loops (VLLE)**: recommended for systems where the liquid phase may be unstable (will split in two liquid phases with different compositions);
3. **Inside-Out (VLE/VLLE)**: recommended for petroleum simulations with many pseudocomponents. The 3-phase option must be used when a second liquid phase is expected (i.e. free water);
4. **Gibbs Minimization (VLE/VLLE)**: recommended for difficult chemical systems;
5. **Nested Loops for Eutectic Solid Systems (SLE-E)**: Calculates Solid-Liquid Equilibria for eutectic systems considering the solid phase as being ideal.
6. **Nested Loops for Solid Solution Systems (SLE-SS)**: Calculates Solid-Liquid Equilibria for solid solution systems considering the solid phase as being ideal.
7. **Nested Loops (3-phase immiscible VLLE)**: For systems with an immiscible second liquid phase (VLLE). The first compound selected in the key compound list will be the immiscible one.

You can configure the added Flash Algorithms by selecting the item to configure and clicking on the "Configure" button on the "Added Flash Algorithms" section.

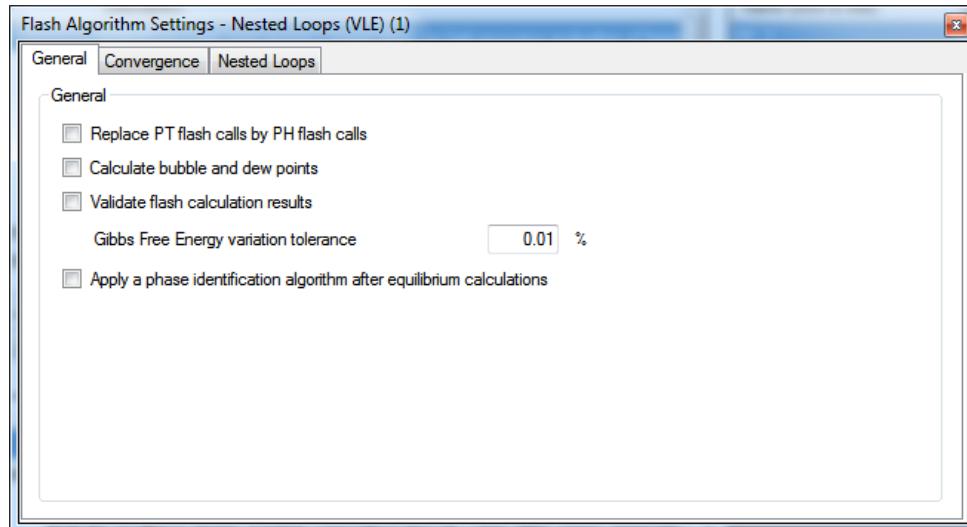


Figure 12: Property Package configuration interface.

#### → Force Pressure-Enthalpy (PH) Flash calculations

If enabled, all requests by unit operations for PT Flashes will be replaced by PH ones. This option must be activated if you are working with only one component (steam simulation, for example), otherwise you won't have partial vaporization/liquefaction in valves, compressors and expanders.

→ *Calculate Bubble and Dew points at stream conditions*

Check this box if you want the DWSIM to calculate bubble and dew points at conditions specified on each material stream. The calculated values will be shown only if the stream is at VLE equilibrium. The calculations are not exactly fast, so use this option with caution and only if needed.

→ *Validate Equilibrium Flash Calculation Results*

If enabled, DWSIM will check the mixture Gibbs energy before and after the equilibrium flash calculation. If the gibbs energy *increases* after the calculation (it should always *decrease* when there is a phase split), an error message will be shown and the flowsheet calculation will be aborted.

→ *Apply a Phase Identification Algorithm after Equilibrium Calculations*

Check this to apply an identification algorithm to each phase after the equilibrium calculation is finished. This can be useful for supercritical compounds which behave as liquid at high pressures and temperatures, or special mixtures which exhibit LLE behavior at low temperatures, incorrectly identified as VLE by the flash algorithms.

**This procedure is enabled by default and will override the flash result regarding phase identification.** For instance, the result of a flash calculation may give an all-vapor solution and, after the phase identification algorithm is applied, the phase may, in fact, behave as liquid. In this case, DWSIM will show this phase as liquid in the Material Stream property window.

Visit DWSIM's wiki for more information about the phase identification algorithm.

→ *Stability Test*

When a flash algorithm with a three-phase capability is selected, DWSIM needs to know which components are most likely to be present in a second liquid phase. You can do that by selecting the ones you think that will be in a higher amount - that is, the key components for the second liquid phase. That doesn't necessarily imposes that the other, unselected components cannot be present in the second liquid phase - only the equilibrium calculation will tell you that.

The search severity setting controls how much effort DWSIM should put into searching for a second liquid phase. In the *Low* setting, only one trial phase is created with a mixture of the key components and the convergence tolerance is somewhat high. In the *High* setting, DWSIM will include additional trial phases, one for each key component and do additional checks using incipient phases' gibbs energy, with the smallest convergence tolerance value (1E-06).

**2.2.2.3. Multiple Property Packages & Flash Algorithms** DWSIM allows multiple Property Packages and Flash Algorithms to be added to a single simulation, which can be associated to each unit operation and material stream on a individual basis. Each property package and flash algorithm has its own settings, even if two or more packages/algorithms are of the same type.

### 2.2.3. Systems of Units

Three basic units systems are present in DWSIM: **SI System** (selected by default), **CGS System** and **English (Imperial) System**. The simulation's units system can be viewed/modified in the "Units System" section of the "Options" tab in the simulation configuration window (Figure 13).

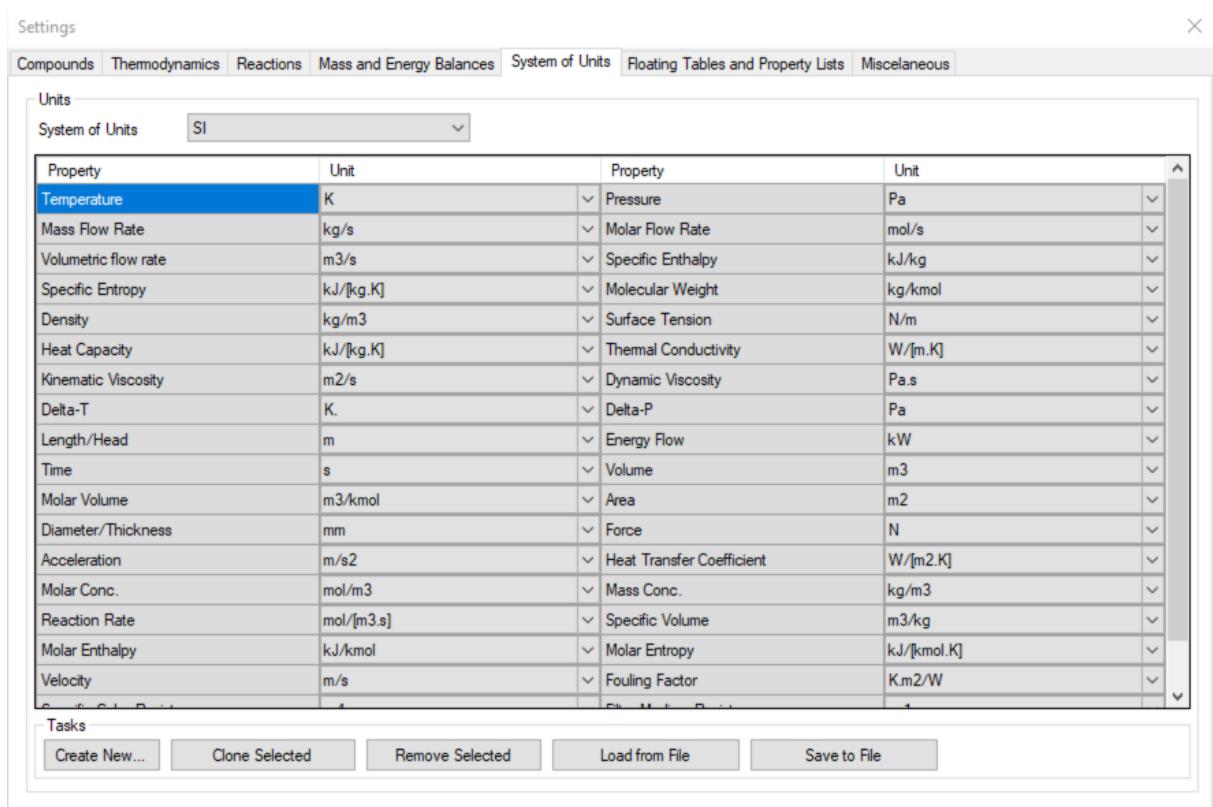


Figure 13: System of Units configuration interface.

There are buttons available on this interface to create custom units systems and save/load them. It is worth remembering that the units systems can also be modified at any time during the simulation - the changes are applied immediately.

#### 2.2.4. Miscelaneous

In the "Misc" section it is possible to define the number formatting in the simulation (Figure 14). In the "Description" group box it is possible to edit some information about the active simulation (title, author and description). You can also define a password to prevent the simulation of being opened by anyone, but this feature only works with the Compressed XML simulation file format (\*.dwxmz).

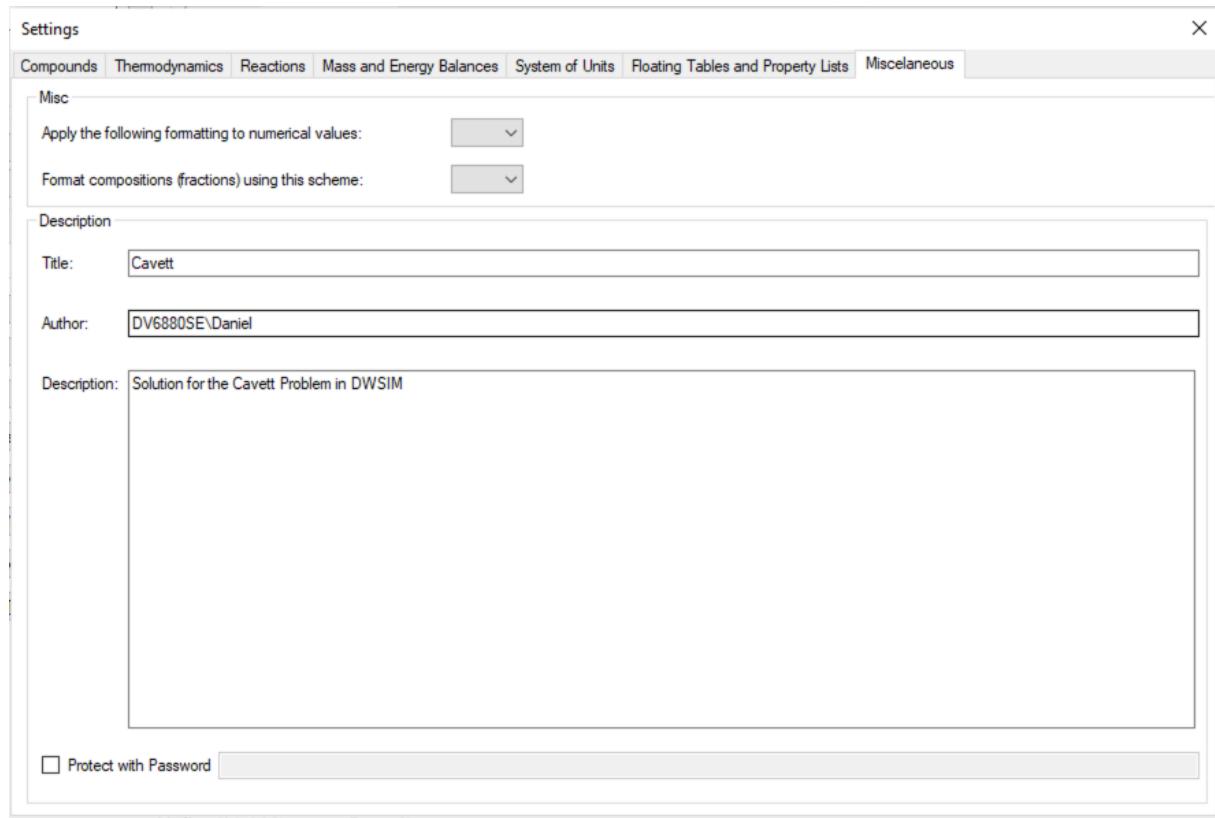


Figure 14: Misc settings interface.

### 2.2.5. Property Tables

In the "Property Tables" section you can define which properties are going to be shown for each object type when you hover the mouse over the objects on the flowsheet. The property list is saved in a per-simulation basis.

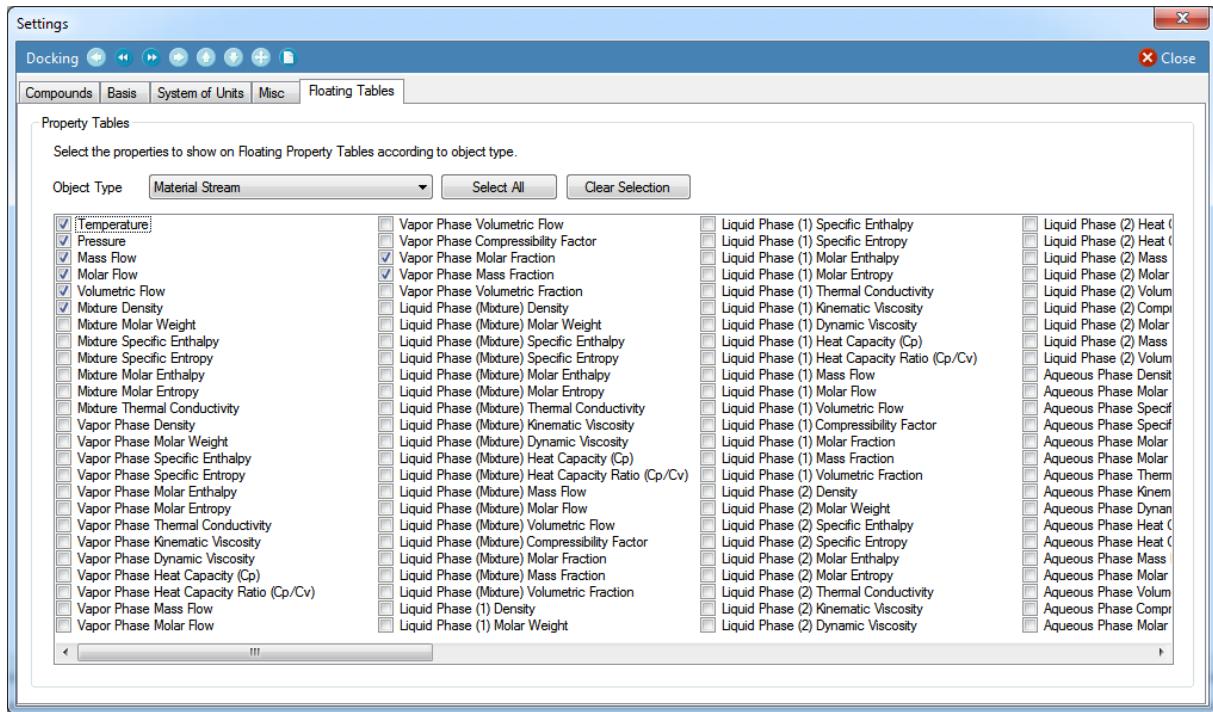


Figure 15: Property Tables settings interface.

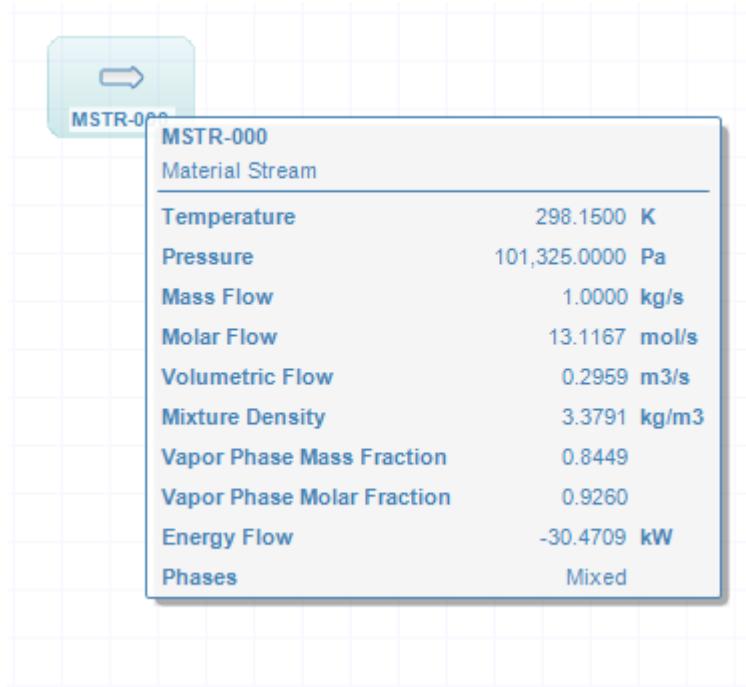


Figure 16: Selected properties on the previous image are shown on the flowsheet for the Material Streams.

### 2.3. Process modeling (Flowsheeting)

After configuring the simulation, the user is taken to the main simulation window (Figure 17). In this window we can highlight the following areas:

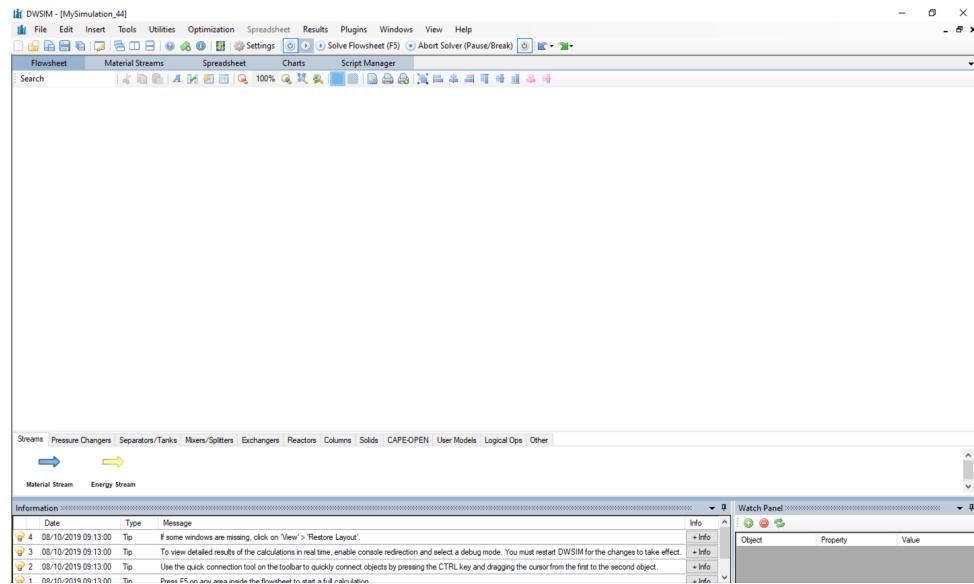


Figure 17: DWSIM simulation window.

- **Menu Bar**: file handling, window arrangement, help, simulation settings, solver controls, undo/redo buttons.
- **Object Palette** window: shows objects which can be added by dragging them into the PFD;
- **Material Streams** window: lists the material streams in the flowsheet and their calculated properties;
- **Flowsheet** window: process flowsheet building and editing area;
- **Information** window: general information about the active simulation;
- **Spreadsheet** window: shows the spreadsheet, a utility to do math operations with data provided by the objects in the current simulation;
- **Charts** window: used to create and view charts from flowsheet objects or from spreadsheet data;
- **Script Manager** window: displays the script manager, which can be used to write Python scripts to automate certain simulation tasks.

When running DWSIM on a Windows platform, the simulation windows can be freely repositioned, with the arrangement information being saved together with the rest of simulation data. To reposition a window, the user should click with the left mouse button in the window's top bar and drag it to the desired place. A preview of how the window will be is shown in blue (Figure 18).

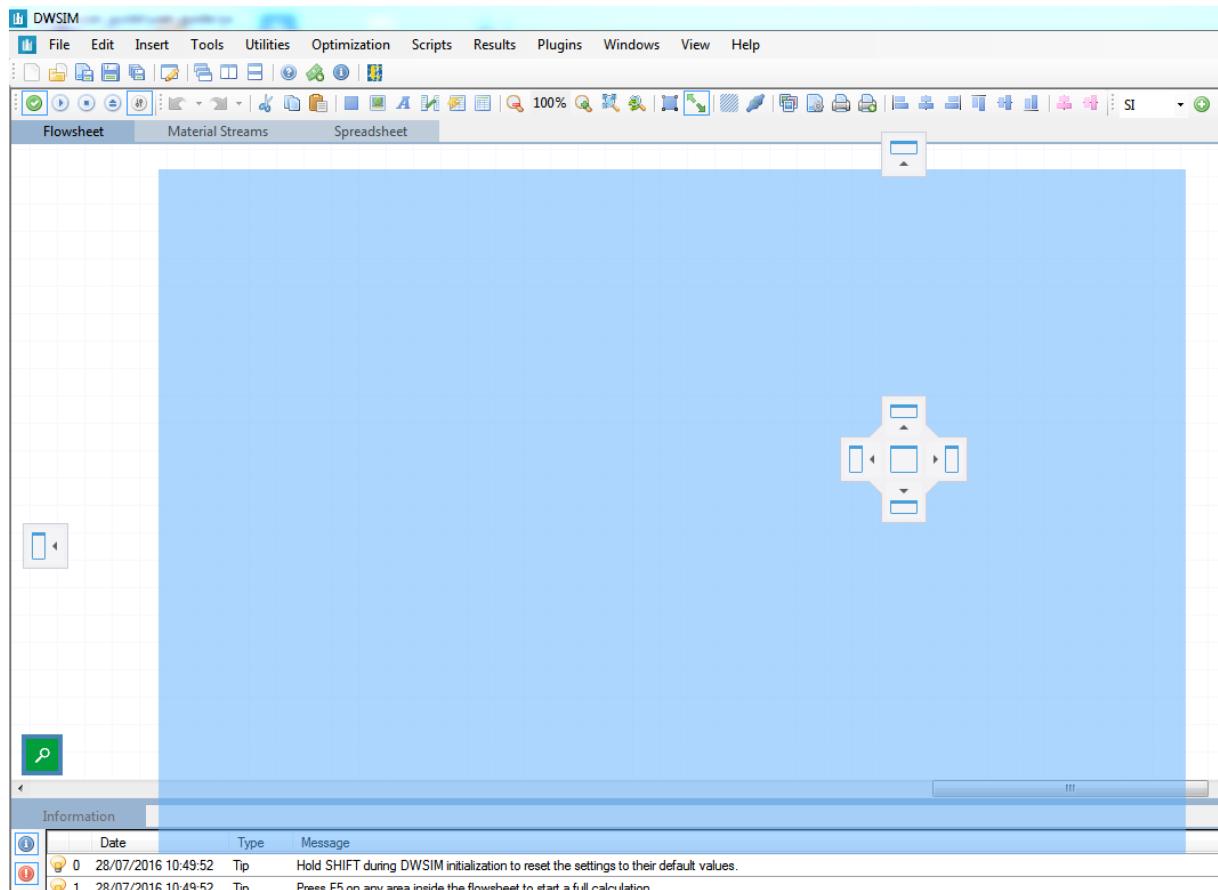


Figure 18: Window repositioning.

When running DWSIM on Mono (Linux), use the context menus (right-click with the mouse on the window caption bar) on each window to reposition/dock its contents.

### 2.3.1. Inserting Flowsheet Objects

To add an object to the flowsheet, you can:

- Use the **Insert > Flowsheet Object** menu item (keyboard shortcut: Ctrl+A):

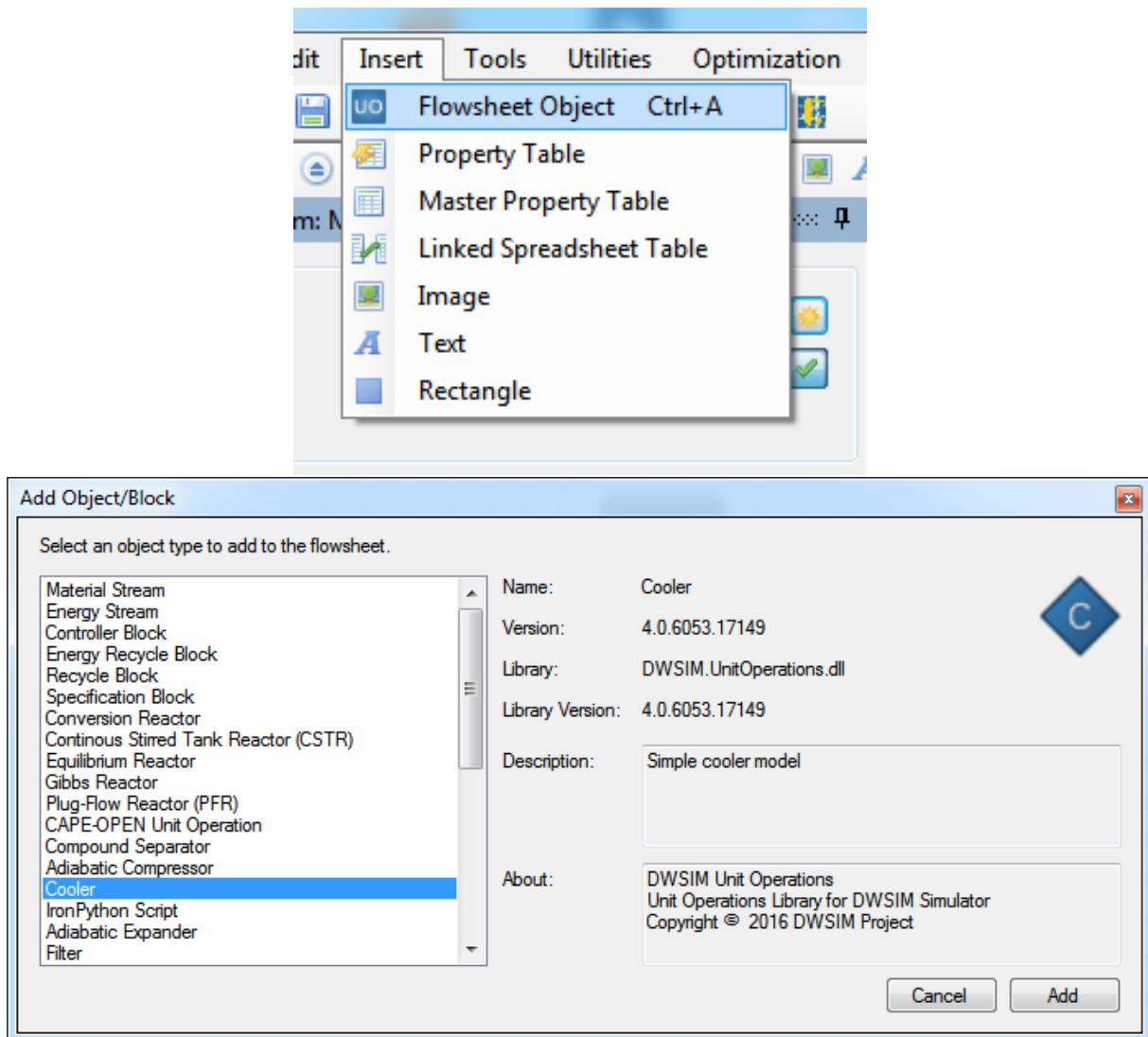


Figure 19: Inserting an object to the flowsheet.

→ Drag an item from the **Object Pallette** window located on the bottom of the flowsheet panel:

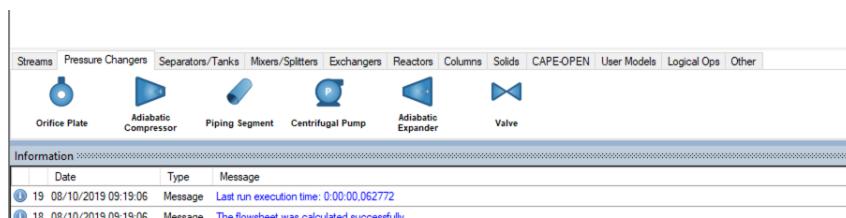


Figure 20: Inserting an object to the flowsheet by dragging from the Object Pallette window.

The elements of a simulation (objects) which can be added to the flowsheet are:

→ **Material Stream:** used to represent matter which enters and leaves the limits of the simulation and passes through the unit operations. The user should define their conditions and composition in order for DWSIM to calculate their properties accordingly;

- **Energy Stream:** used to represent energy which enters and leaves the limits of the simulation and passes through the unit operations;
- **Mixer:** used to mix up to three material streams into one, while executing all the mass and energy balances;
- **Splitter:** mass balance unit operation - divides a material stream into two or three other streams;
- **Valve:** works like a fixed pressure drop for the process, where the outlet material stream properties are calculated beginning from the principle that the expansion is an isenthalpic process;
- **Pipe:** simulates a fluid flow process (mono or two-phase). The pipe implementation in DWSIM provides the user with various configuration options, including heat transfer to environment or even to the soil in buried pipes. Two correlations for pressure drop calculations are available: Beggs & Brill and Lockhart & martinelli. Both reduces to Darcy equation in the case of single-phase flow;
- **Pump:** used to provide energy to a liquid stream in the form of pressure. The process is isenthalpic, and the non-idealities are considered according to the pump efficiency, which is defined by the user;
- **Tank:** in the current version of DWSIM, the tank works like a fixed pressure drop for the process;
- **Separator Vessel:** used to separate the vapor and liquid phases of a stream into two other distinct streams;
- **Compressor:** used to provide energy to a vapor stream in the form of pressure. The ideal process is isentropic (constant entropy) and the non-idealities are considered according to the compressor efficiency, which is defined by the user;
- **Expander:** the expander is used to extract energy from a high-pressure vapor stream. The ideal process is isentropic (constant entropy) and the non-idealities are considered according to the expander efficiency, which is defined by the user;
- **Heater:** simulates a stream heating process;
- **Cooler:** simulates a stream cooling process;
- **Conversion Reactor:** simulates a reactor where conversion reactions occur;
- **Equilibrium Reactor:** simulates a reactor where equilibrium reactions occur;
- **PFR:** simulates a Plug Flow Reactor (PFR);
- **CSTR:** simulates a Continuous-Stirred Tank Reactor (CSTR);
- **Shortcut Column:** simulates a simple distillation column with approximate results using shorcut calculations;
- **Distillation Column:** simulates a distillation column using rigorous thermodynamic models;
- **Absorption Column:** simulates an absorption column using rigorous thermodynamic models;
- **Refluxed Absorber:** simulates a refluxed absorber column using rigorous thermodynamic models;

- **Reboiled Absorber:** simulates a reboiled absorber column using rigorous thermodynamic models.
- **Heat Exchanger:** simulates a countercurrent heat exchanger using rigorous thermodynamic models.
- **Orifice Plate:** model to simulate an orifice plate, used for flow metering.
- **Component Separator:** model to simulate a generic process for component separation.
- **Custom Unit Operation:** an user-defined model based on IronPython/IronRuby scripts.
- **CAPE-OPEN Unit Operation:** External CAPE-OPEN Unit Operation socket for adding CO Unit Operations in DWSIM.
- **Spreadsheet Unit Operation:** Unit Operation where the model is defined and calculated in Spreadsheet (XLS/XLSX/ODS) files.
- **Solids Separator:** model to simulate a generic process for solid compound separation.
- **Continuous Cake Filter:** continuous cake filter model for solids separation.

Additionally, the following logical operations are available in DWSIM:

- **Adjust:** used to make a variable to be equal to a user-defined value by changing the value of other (independent) variable;
- **Specification:** used to make a variable to be equal to a value that is a function of other variable, from other stream;
- **Recycle:** used to mix downstream material with upstream material in a flowsheet,
- **Energy Recycle:** used to mix downstream energy with upstream energy in a flowsheet.

Figure 21 shows a material stream added to the flowsheet by one of the method described above. It can be observed that the stream is selected and that its property editor is shown as a panel on the left part of the main window.

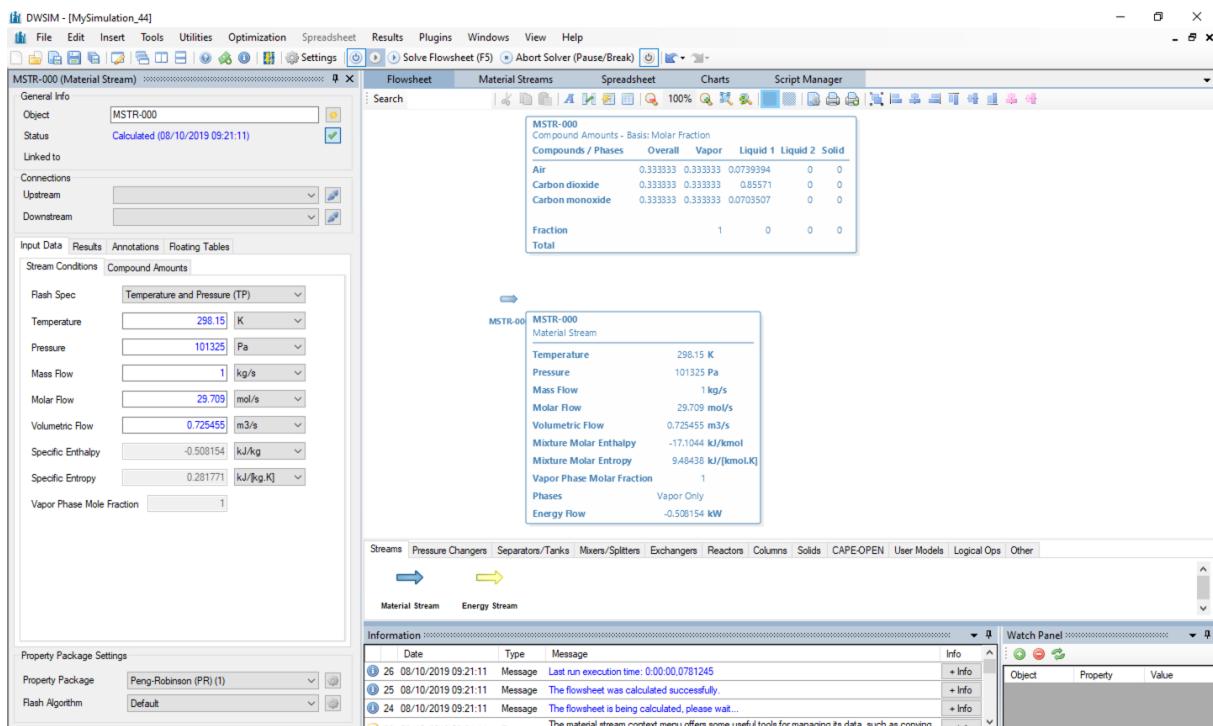


Figure 21: A material stream in the flowsheet.

**Connecting objects** The material streams represent mass flowing between unit operations. There are two different ways in which a material stream can be connected to a unit operation (or *vice-versa*):

- Through the context menu activated with a right mouse button click over the object (Figure 22);

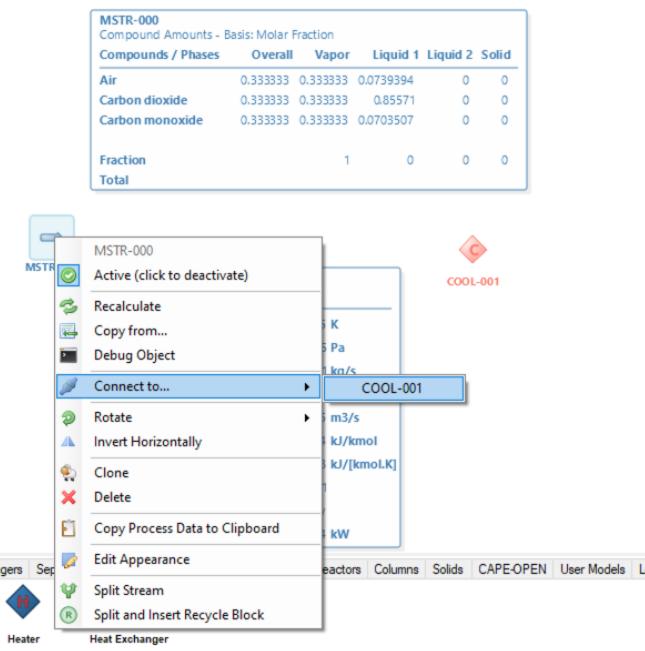


Figure 22: Selected object context menu.

- Through the property editor window - Connections section.

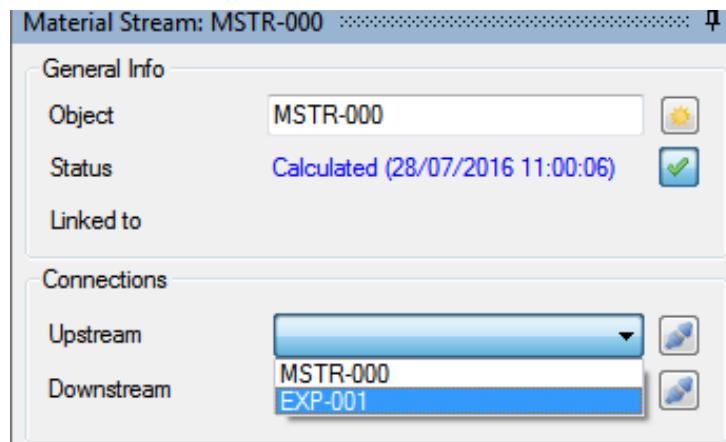


Figure 23: Connection selection menu.

- Through the "Create and Connect" buttons on the object editors. When you click on these buttons, DWSIM will automatically create and connect streams to the associated ports on the selected object.

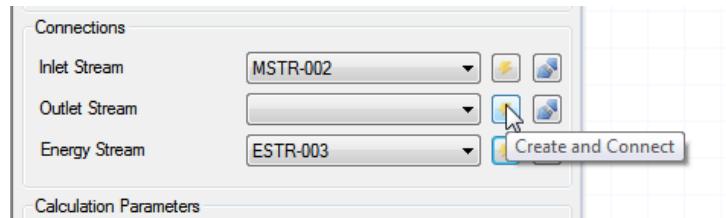


Figure 24: Create and Connect tool.

An expander system with its connections is shown on Figure 25.

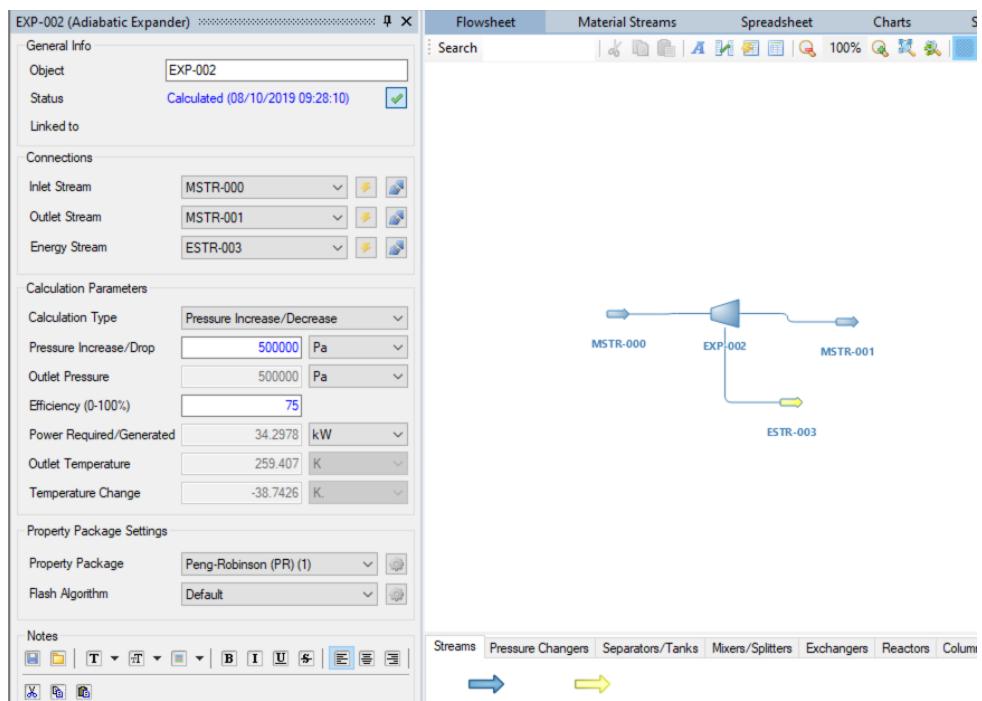


Figure 25: Expander with all connections correctly configured.

**Disconnecting objects** Tools to disconnect objects from each other can be found on the same locations as the connecting ones.

**Removing objects from the flowsheet** The selected object can be removed from the flowsheet by pressing the DEL keyboard button or by using the context menu - "Delete" item (Figure 22).

### 2.3.2. Process data management

**Entering process data** The objects' process data (temperature, pressure, flow, composition and/or other parameters) can be entered in the property editor window (Figure 26). Properties that cannot be edited (read-only) are grayed-out.

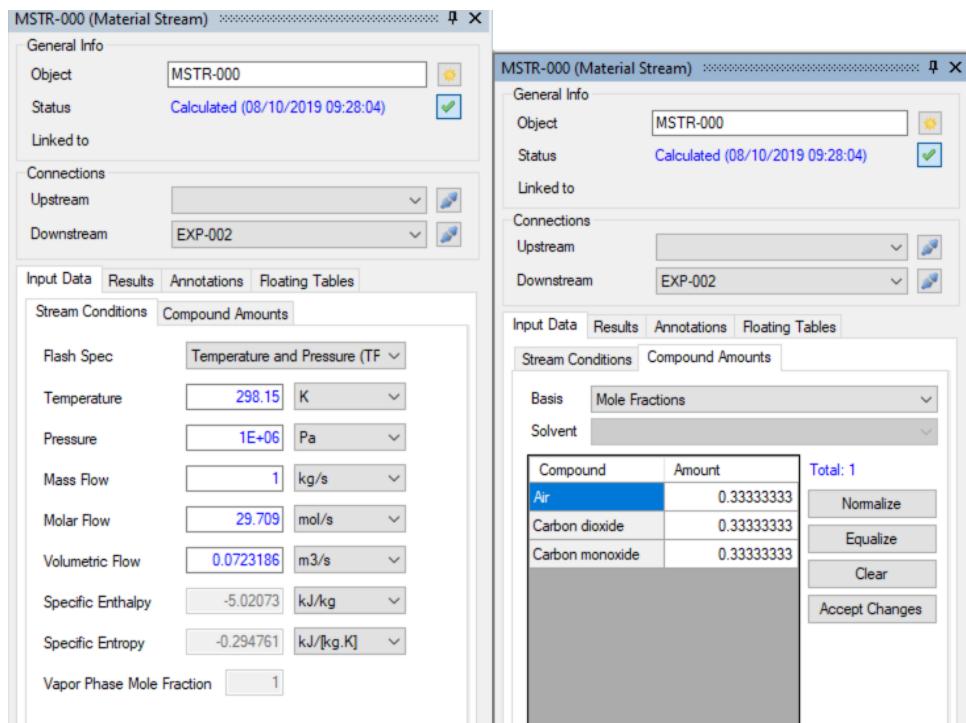


Figure 26: Viewing object properties in the editor window.

Most properties can be edited directly by typing a value in the textbox and pressing ENTER. DWSIM will then commit the new property value and trigger the flowsheet solver.

Property	Value	Unit
Temperature	300	K
Pressure	101325	Pa
Mass Flow	1	kg/s

Figure 27: Direct editing of a property.

You can also use the inline units converter to convert the value of a property from the desired units to the current selected units. Type the value of the property on the textbox and select the unit to convert from at the combobox on the right. DWSIM will then convert the value from the selected units on the combobox to the actual units of the simulation system of units.

Input Data	Phase Compositions	Phase Properties	Annotations
Flash Spec	Temperature and Pressure (TP) <input type="button" value="▼"/>		
Temperature	50	K	<input type="button" value="▼"/>
Pressure	101325	K R C F	<input type="button" value="▼"/>
Mass Flow	1	<input type="button" value="▼"/>	
Molar Flow	<input type="button" value="▼"/>		

Figure 28: Converting 50 C to the current temperature units (K).

Input Data	Phase Compositions	Phase Properties	Annotations
Flash Spec	Temperature and Pressure (TP) <input type="button" value="▼"/>		
Temperature	323.15	K	<input type="button" value="▼"/>
Pressure	101325	Pa	<input type="button" value="▼"/>
Mass Flow	1	kg/s	<input type="button" value="▼"/>

Figure 29: Converted temperature value (323.15 K).

If all object properties were correctly defined, it will be calculated by DWSIM and its flowsheet representation will have a blue border instead of a red one, indicating that the object was calculated successfully (Figure 30).

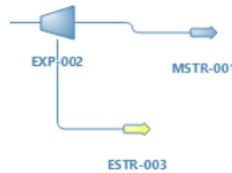


Figure 30: Calculated objects.

### 2.3.3. Undo/Redo actions

DWSIM supports Undo and Redo for certain actions, like:

- Changing a property of an object (model and graphic object properties)
- Connecting, disconnecting objects
- Adding and removing objects
- Adding and removing compounds

- Adding, removing and editing Property Packages
- Changing Spreadsheet cell values

To undo or redo an action, click on the "Edit" menu item and then on the corresponding menu items (Figure 32). You can also access a list of actions to undo or redo through the arrow buttons located in the menu strip, near to the "Flowsheet" label (Figure 31).

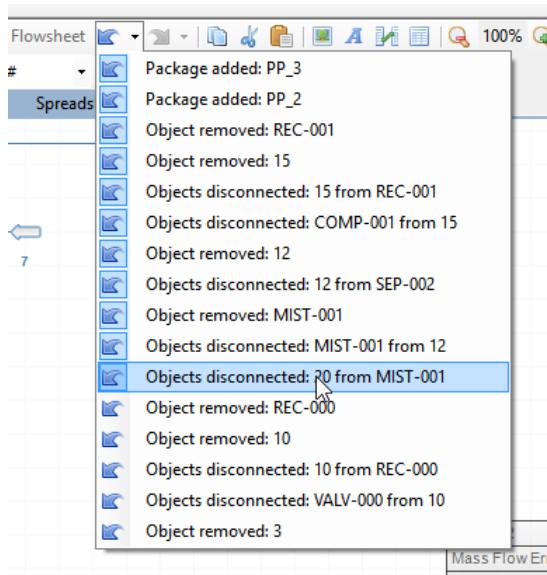


Figure 31: Undo action list.

The "Edit" menu also contains some handy tools for handling flowsheet objects, as well as simulation and application configuration menu items (Figure 32).

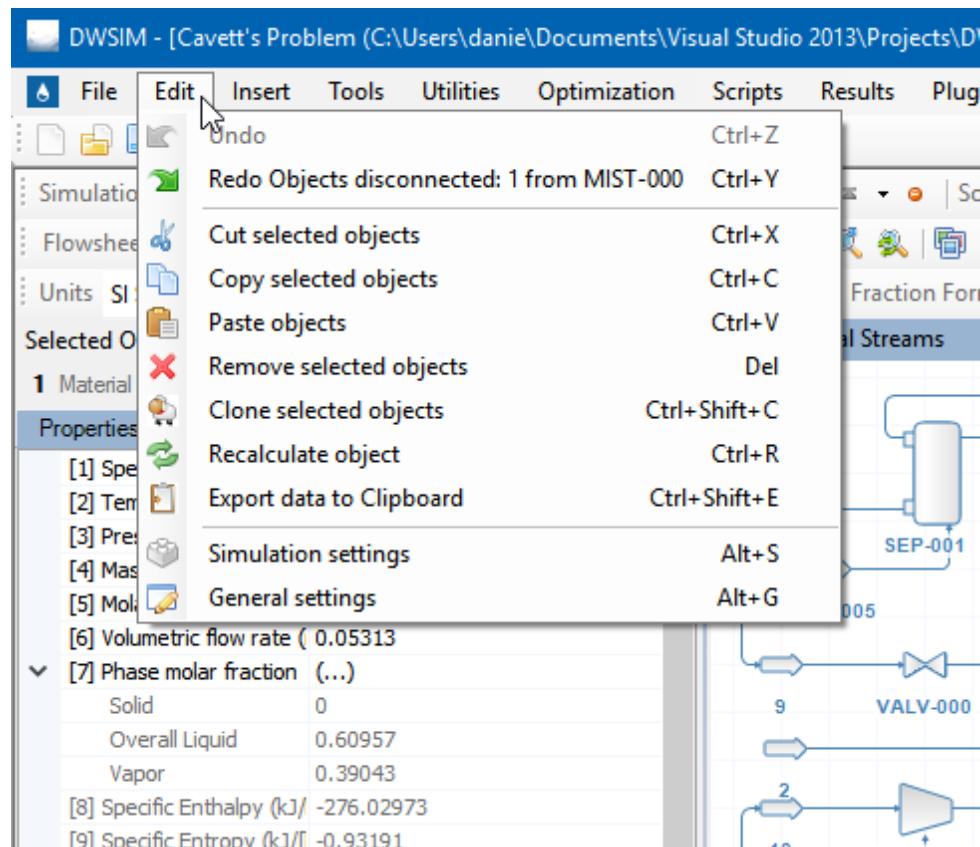


Figure 32: Edit menu.

### 2.3.4. Cut/Copy/Paste objects

DWSIM also supports cutting, copying and pasting flowsheet objects inside a flowsheet or between different flowsheets. When copying objects between flowsheets, DWSIM may also copy compounds and property packages from one flowsheet to another. Cut/Copy/Paste behavior is an application setting and can be set in the General Settings menu (Section 5.5).

### 2.3.5. Simulation

DWSIM is a sequential modular process simulator, that is, all calculations are made in a per-module basis, according to the connections between the objects. The calculator checks if an object has all of its properties defined and, if yes, passes the data for the downstream object and calculates it, repeating the process in a loop until it reaches an object that doesn't have any of its downstream connections attached to any object. This way, the entire flowsheet can be calculated as many times as necessary without having to "tell" DWSIM which object must be calculated. In fact, this is done indirectly if the user define all the properties and make all connections between objects correctly.

**DWSIM's calculation starts when the user edits a property which defines an object.** For example, editing a stream mass flow when its temperature, pressure and composition are already well-defined activates DWSIM's calculator.

It is possible to control DWSIM's calculator by using its button bar (Figure 33). Clicking on the button activates or deactivates the calculator. The button performs a full flowsheet recalculation.

DWSIM's calculator is enabled by default - if it is disabled, modifying of a property is accepted, but **does not** recalculate the object nor the ones that are downstream in the flowsheet. The  button stops the any ongoing calculation.



Figure 33: DWSIM's calculator control bar.

As DWSIM's calculator does its job, messages are added to the "Information" window. These messages tell the user if the object was calculated successfully or if there was an error while calculating it, among others (Figure 34).

	Date	Type	Message
⚠ 7	8/7/2008 11:21:53	Warning	COMP-000: No energy stream associated with the compressor.
ⓘ 6	8/7/2008 11:13:59	Message	File C:\Documents and Settings\DANIEL\Desktop loaded successfully

Figure 34: A DWSIM's calculator message.

### 2.3.6. Results

Results can be viewed in reports, generated (Figures 35 and 36) for printing. Report data can also be saved to ODT, ODS, XLS, TXT or XML files.

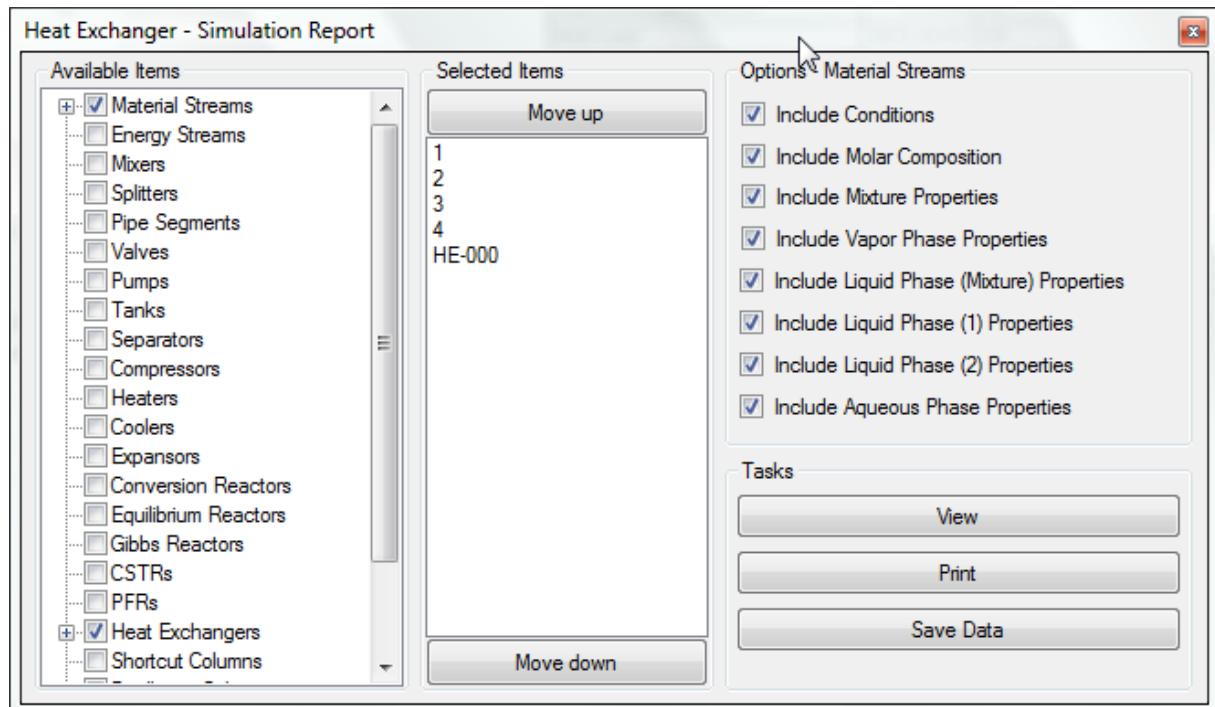


Figure 35: Results report configuration.

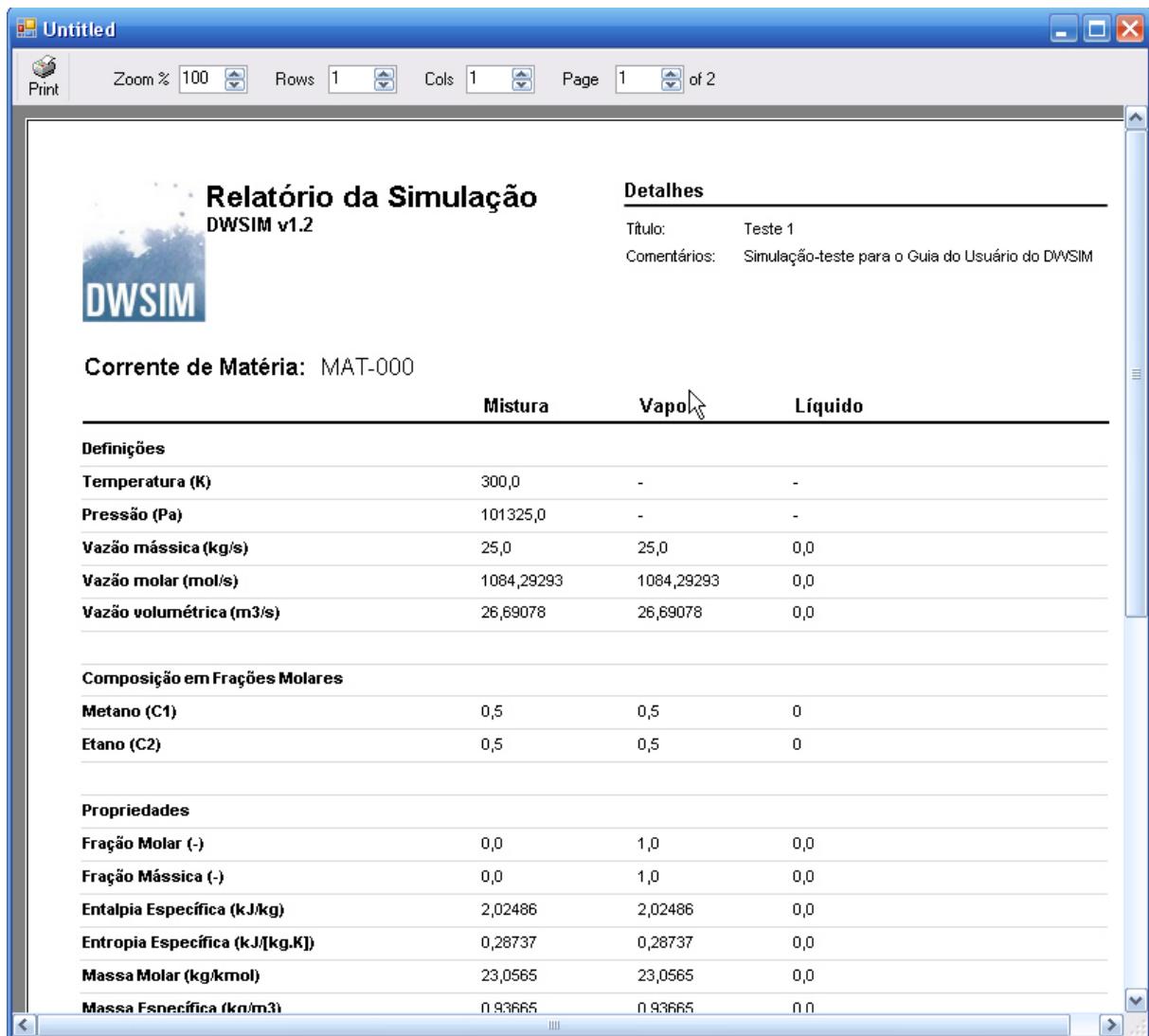


Figure 36: Results report.

## 2.4. Sensitivity Analysis

You can use the Sensitivity Analysis Utility in order to study the influence of up to 2 variables into other dependent flowsheet variables. The changes in variables are defined by a value range and a number of equally spaced points within this range. For example, you can analyze the influence of temperature and pressure in the enthalpy of a mixture, from 200 to 400 K and from 100 to 1000 kPa, nine points for temperature and 5 points for pressure, totaling 45 points on which the enthalpy will be calculated at different temperatures and pressures. This also means that the flowsheet will be recalculated 45 times (!), so be careful with the number of points you choose as the calculation time can be prohibitive.

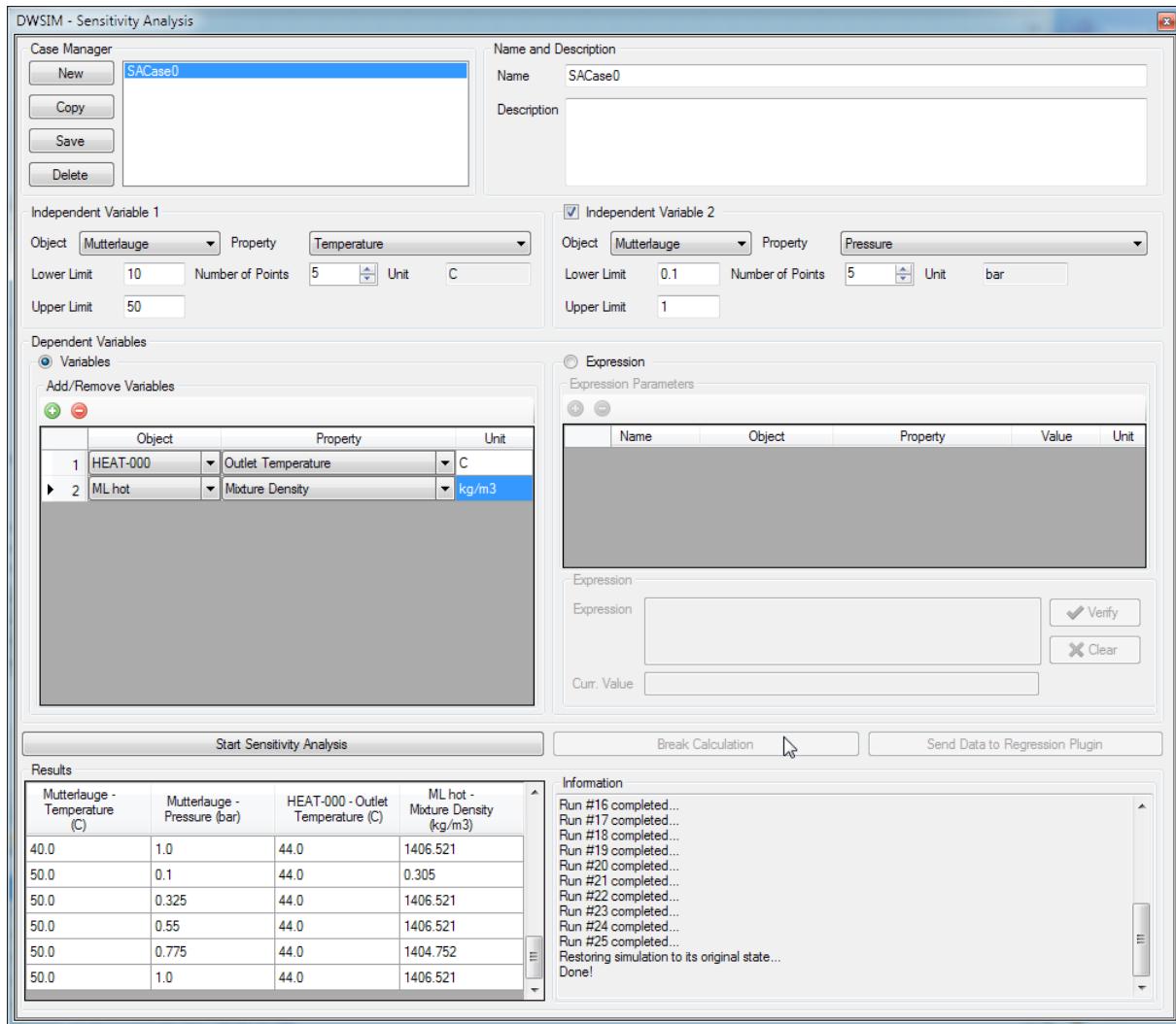


Figure 37: Sensitivity Analysis Utility (1).

The sensitivity analysis utility is based on case studies. In a single simulation one can define a number of cases, each one with its own variables, ranges and results. These cases will be saved together with the simulation, and cannot be exported to other ones. The results are shown in a table, so the data can be copied and pasted into another specialized data analysis software or sent directly to the data regression plugin.

## 2.5. Flowsheet Optimization

The new Optimizer in DWSIM handles single and multivariate optimization problems with or without bound constraints. The objective function can be either a variable in the flowsheet or an expression as a function of as many variables as you need.

The interface is very similar to Sensitivity Analysis's one. One can define a number of cases, each one with its own variables, ranges and results. These cases will be saved together with the current simulation, and cannot be exported to other simulations.

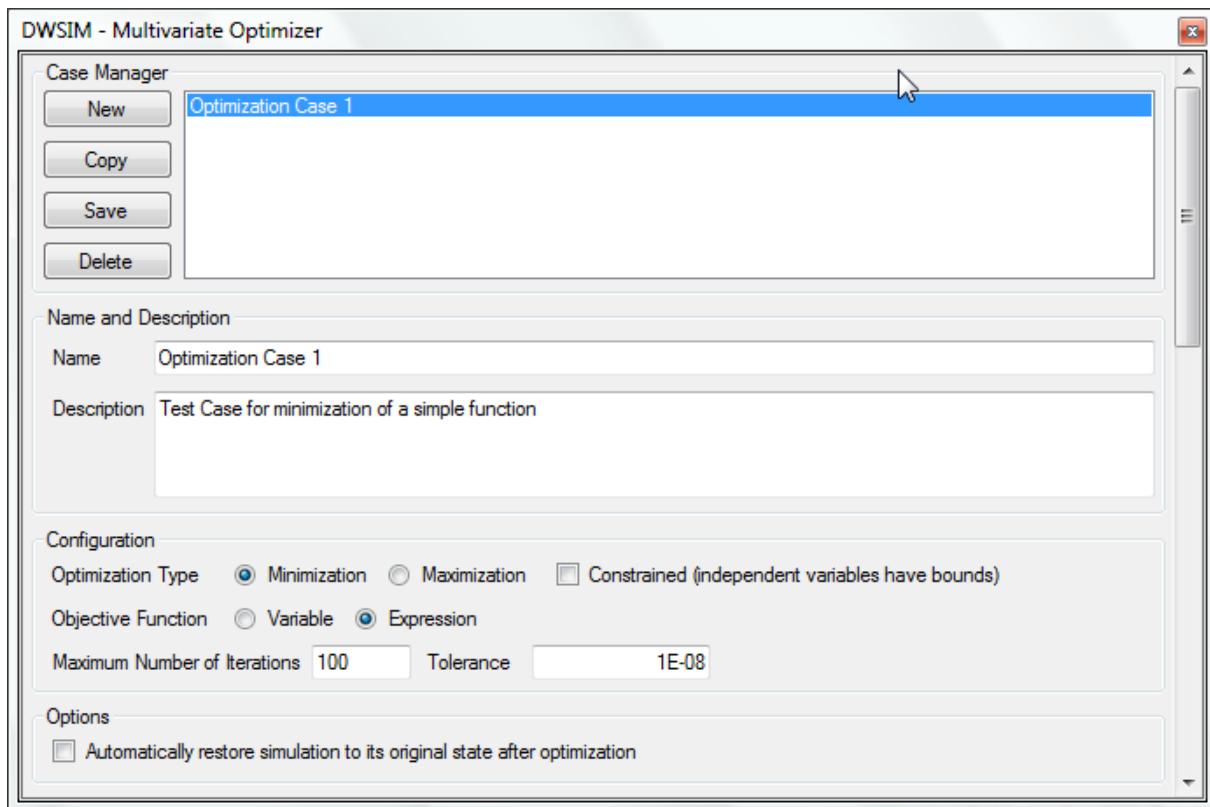


Figure 38: Multivariate Optimization Utility (1).

There are some options to choose from in DWSIM's Multivariate Optimizer. It is possible to select the type of the optimization (minimization or maximization of the objective function), choose if the independent variables will have lower and/or upper bounds and if the objective function will be a flowsheet variable or an expression based on flowsheet variables. One can also define a maximum number for the iterations and a tolerance for the variation of the calculated value for the objective function - if the variation is less than the defined value, the flowsheet is considered optimized and the process stops. There is also an option to choose if the flowsheet will be returned to its original state after optimization, so the results will be shown only in the current window, and the flowsheet initial configuration will remain intact.

In order to define variables to be used in the optimization process, a variable can be added by clicking on the "+" button. With the variable row added to the list, one chooses an object, then the desired property and the type of variable (IND for independent, AUX for auxiliary or DEP for dependent variables). If necessary, one can define a lower and/or upper limit for the IND variables, according to the current unit system. The variable name is the one which will be used in the expression.

DWSIM only considers bounds for independent variables. Also, if the objective function is a DEP variable, and you defined multiple DEP variables, only the first will be used. AUX variables are used in the context of an expression only. To remove a variable, a row must be selected by clicking at the row header before pressing the "-" button.

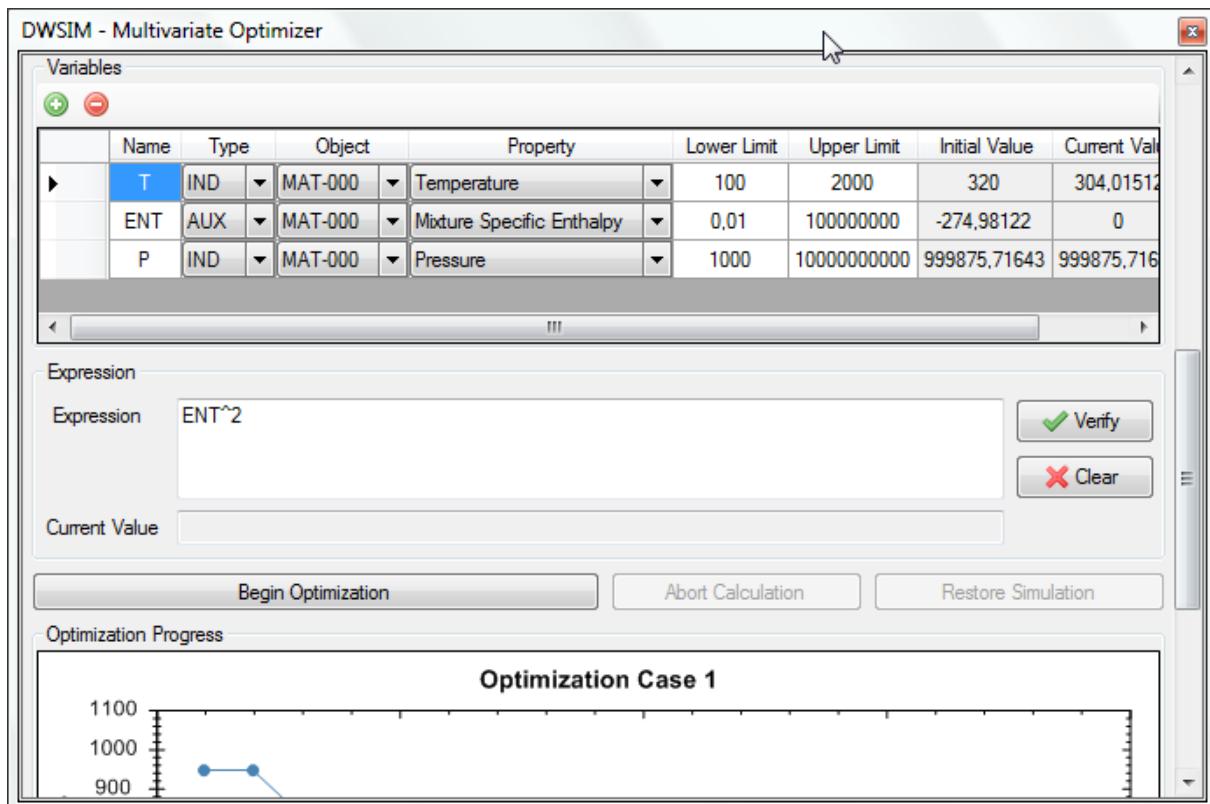


Figure 39: Multivariate Optimization Utility (2).

With all the variables defined and the case configured, the optimization can be carried out by clicking on the appropriate button - the button will become disabled. After some time, if the optimization converges, the button will become active again, indicating that the optimization process is over.

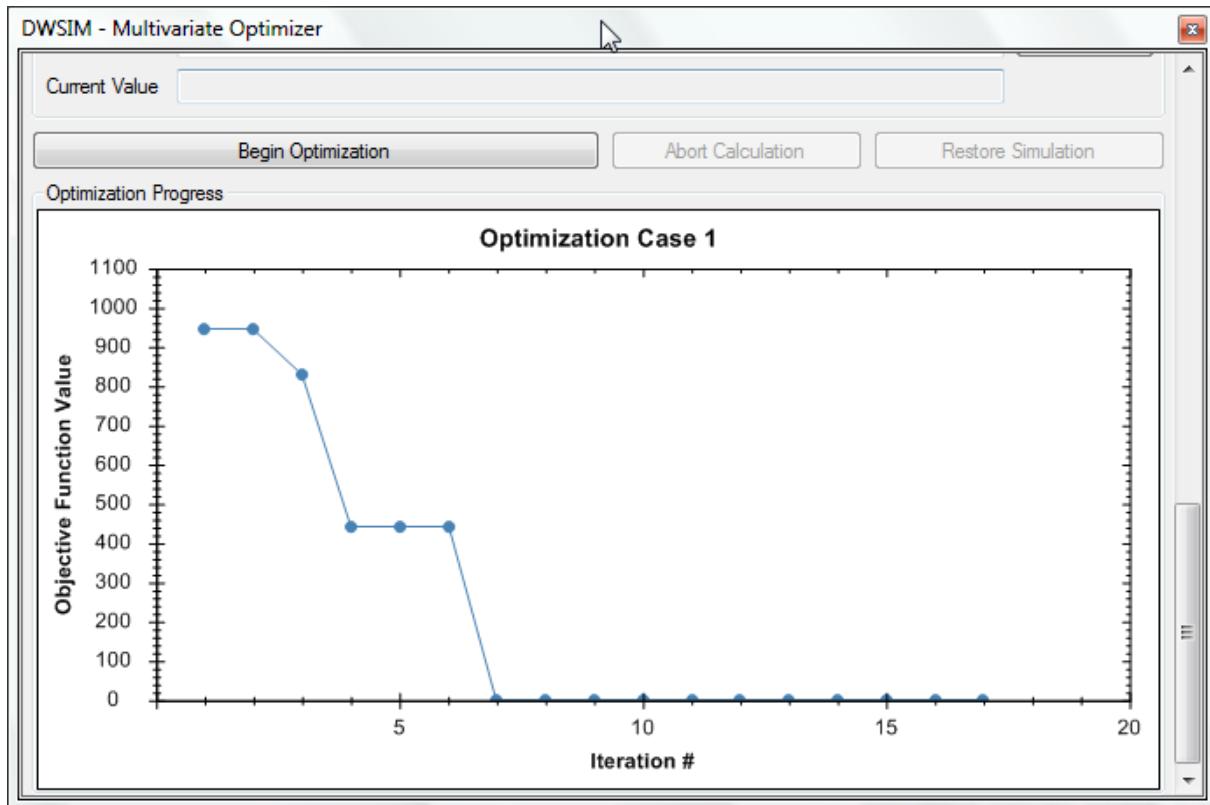


Figure 40: Multivariate Optimization Utility (3).

## 2.6. Utilities

DWSIM includes some utilities which provides the user with more information about the process being simulated.

Utilities can be added and attached to Flowsheet objects (**Utilities > Add Utility** menu item). After being attached, they will be saved together with simulation data and restored upon reopening. Some data from the attached utilities will be available to be displayed on property tables and used on sensitivity analysis and optimization studies.

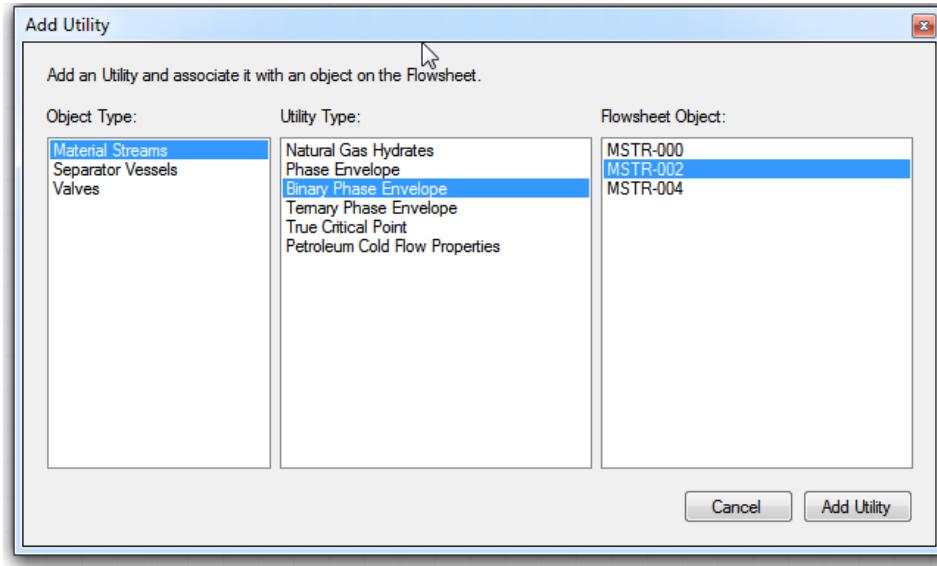


Figure 41: Attaching Utilities through the "Add Utility" window.

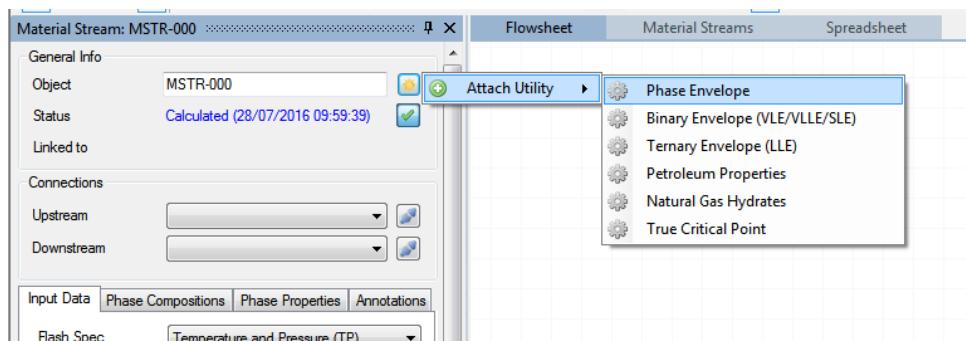


Figure 42: Attaching Utilities through the object editors.

Added/Attached Utilities will be visible on the context menu located on the object editors, on the right of the Object's Name textbox.

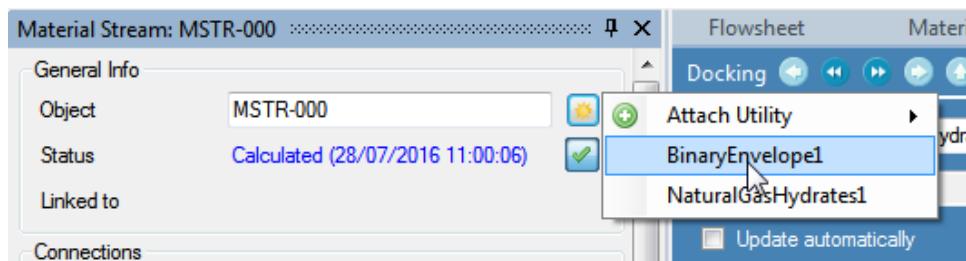


Figure 43: Accessing attached Utilities.

→ **True Critical Point** - utility to calculate the true critical point of a mixture (Figure 44).

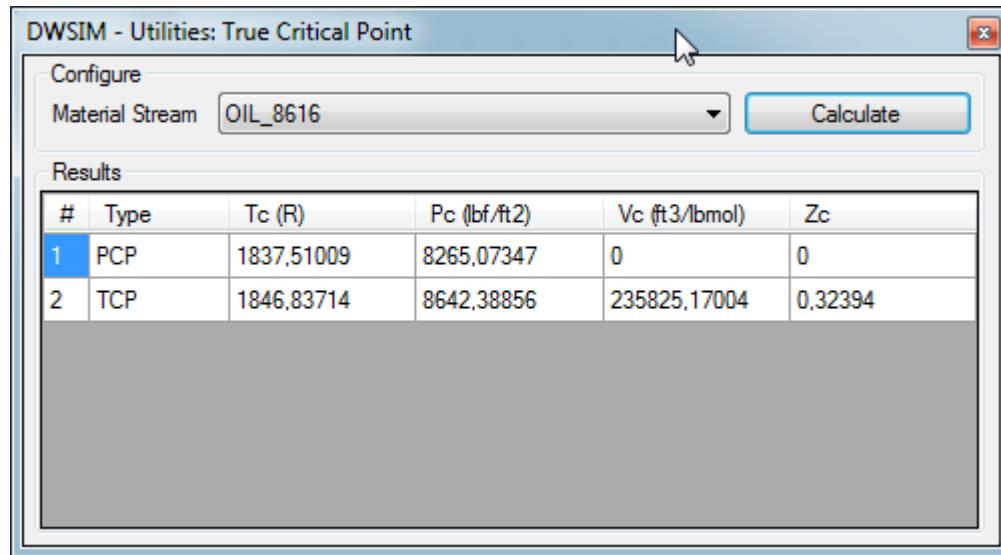


Figure 44: Utilities - True Critical Point.

→ **Hydrate Equilibrium/Dissociation Utility** - calculation of the equilibrium conditions for natural gas hydrates (Figure 45);

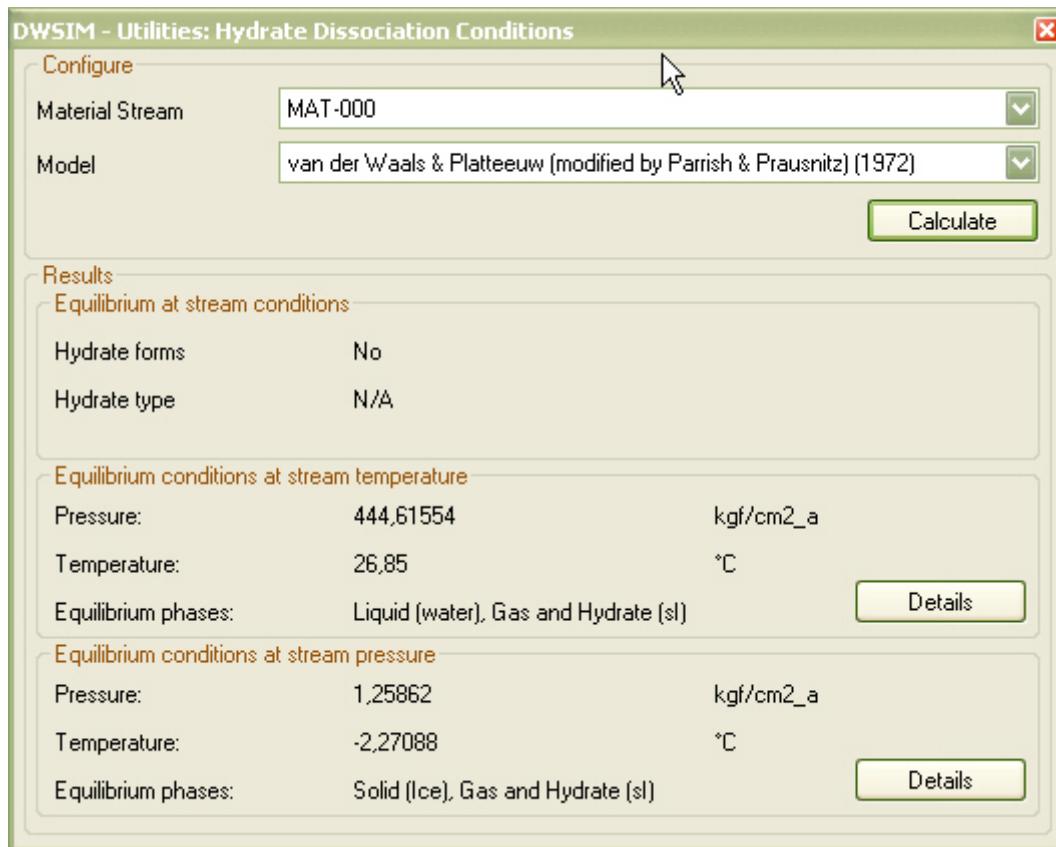


Figure 45: Utilities - Hydrate Calculations.

→ **Phase Envelope** - Material stream phase equilibria envelope calculation (Figure 46);

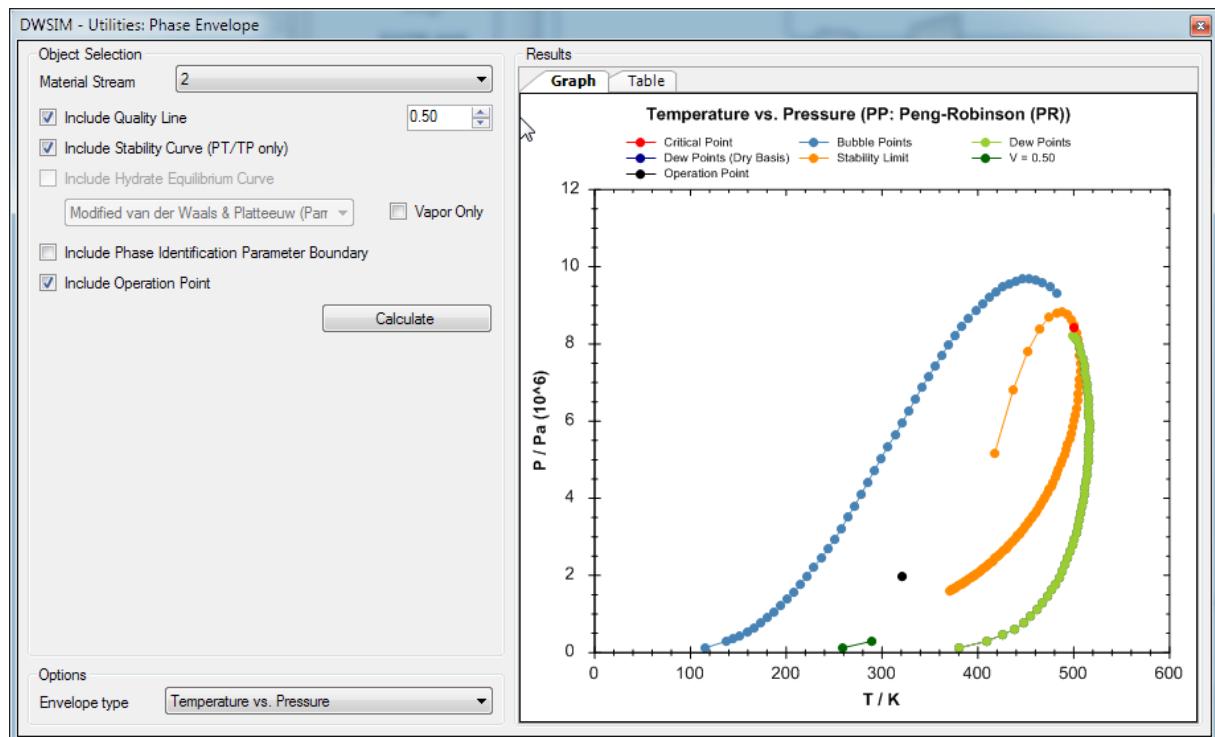


Figure 46: Utilities - Phase Envelope.

→ **Binary Envelope** - special envelopes for binary mixtures (Figure 47).

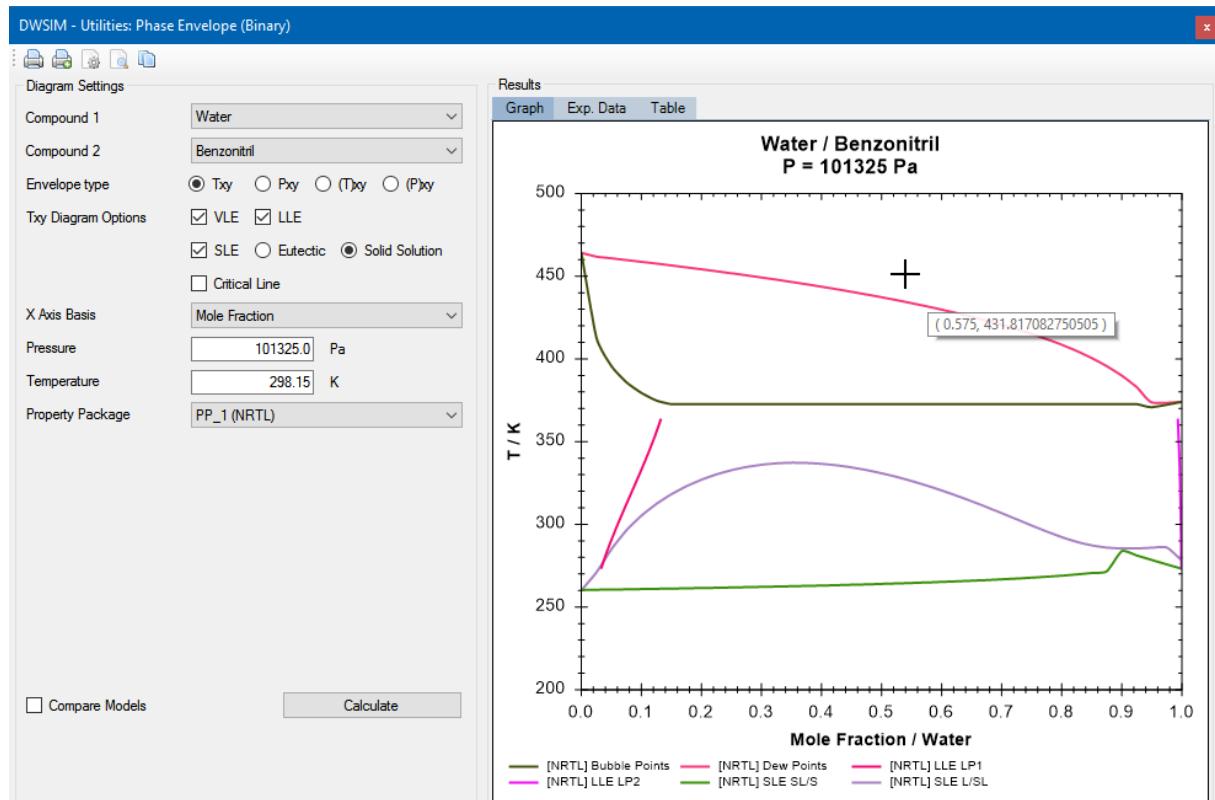


Figure 47: Utilities - Binary Envelope.

- **Petroleum Cold Flow Properties** - special properties of petroleum fractions, like cetane index, flash point, refraction index, etc. (Figure 48).

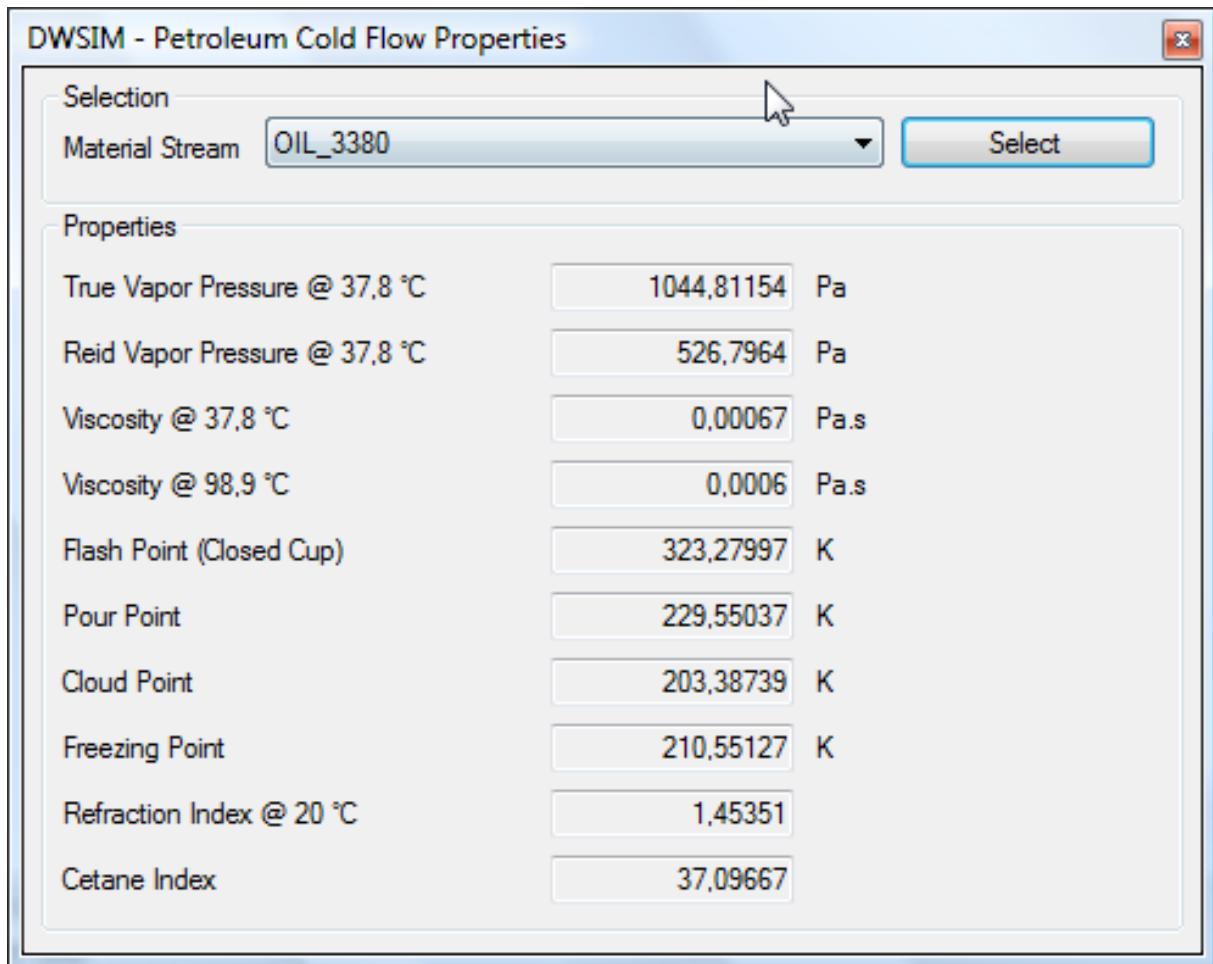


Figure 48: Utilities - Petroleum Cold Flow Properties.

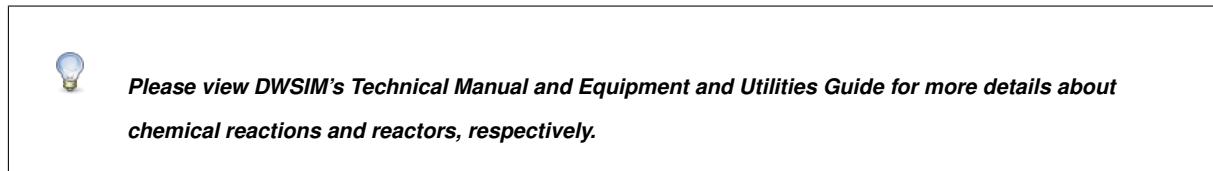
Utilities calculate their properties for one object only, which is selected inside their own windows. In the majority of cases, this object must be calculated in order to be available for selection in the utility window.



*Please view DWSIM's Technical Manual for more details about the models and methods used by the Utilities.*

## 2.7. Chemical Reactions

DWSIM classifies chemical reactions in three different types: Conversion, where the conversion of a reagent can be specified as a function of temperature; Equilibrium, where the reaction is characterized by an equilibrium constant K, and Kinetic/Heterogeneous Catalytic, where the reaction is led by a velocity expression which is a function of concentration of reagents and/or products and/or a catalyst.



Chemical reactions in DWSIM are managed through the **Chemical Reactions Manager** (**Simulation Settings > Reactions** panel) (Figure 49):

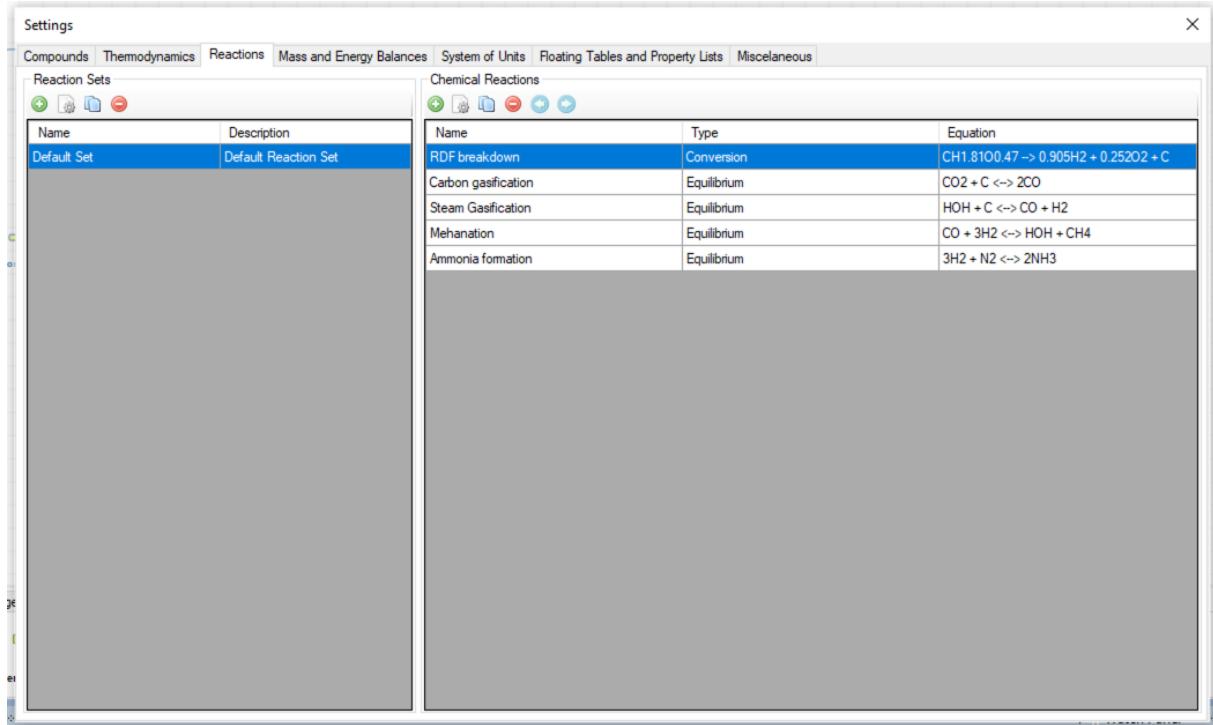


Figure 49: Chemical Reactions Manager.

The user can define various reactions which are grouped in *Reaction Sets*. These reaction sets list all chemical reactions, and the user must activate only those he/she wants to become available for one or more reactors, since the reactor's parameter is the **reaction set** and not the chemical reactions themselves. In the reaction set configuration window it is also possible to define the reaction ordering. Equal indexes define parallel reactions (Figure 50):

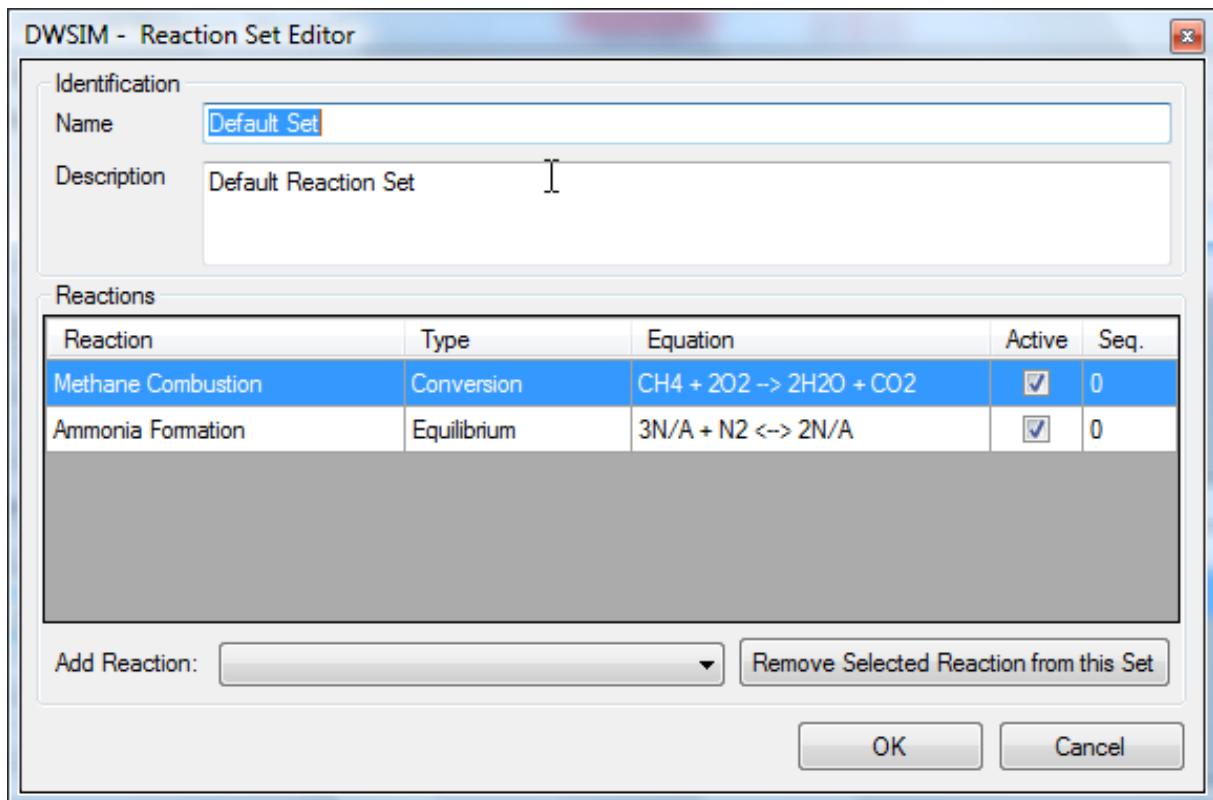


Figure 50: Reaction Set editor.

When the reactions and their respective reaction sets are correctly defined, the sets will be available for selection in the property window of a reactor in the simulation. When requested for a calculation, the reactor will then look for active reactions inside the selected set.

## 2.8. Characterization of Petroleum Fractions

DWSIM provides two tools for characterization of petroleum fractions ("Simulation Settings" > "Tools" > "Petroleum Characterization"). One of them characterizes C7+ fractions from bulk properties (Figure 51). The other characterizes the oil from an ASTM or TBP distillation curve (Figure 52).

In both tools, the characterization is done through the creation of various components with different boiling points (pseudocomponents) which together represent the assay as a whole.

- **Characterization from bulk properties** The method itself requires a minimum of information to generate the pseudocomponents, though the more data the user provides, the better will be the results (Figure 51). It is recommended that the user provides the specific gravity of the C7+ fraction at least. Viscosity data is also very important.

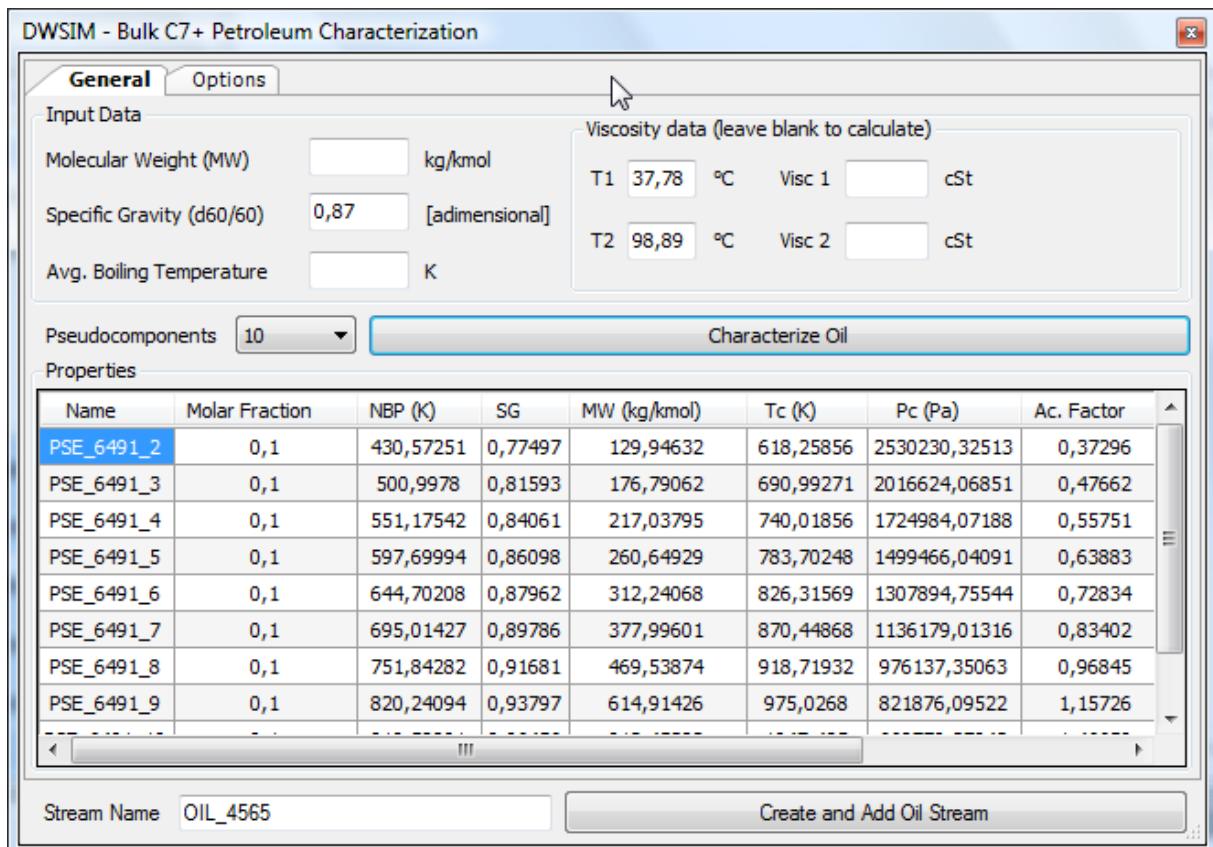


Figure 51: C7+ petroleum fraction characterization utility.

- **Characterization from distillation curves** This tool gets data from an ASTM or TBP distillation curve to generate pseudocomponents. It is also possible to include viscosity, molecular weight and specific gravity curves to enhance the characterization.

The interface has a wizard-like style, with various customization options (Figure 52):

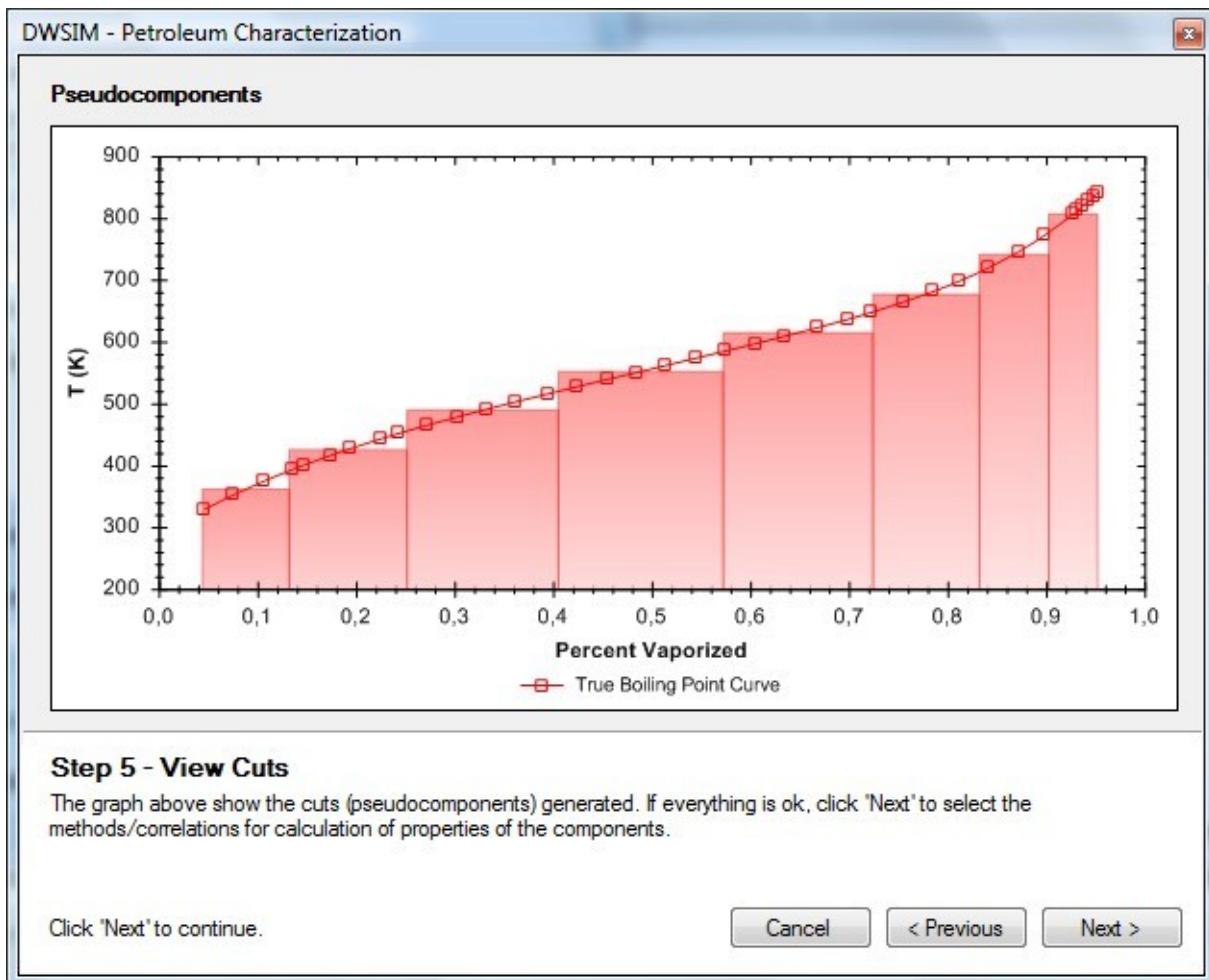


Figure 52: Characterizing petroleum from distillation curves.

After the pseudocomponents are created, a material stream with a defined composition is also created, which represents the characterized petroleum fraction.



*The hypo and pseudocomponents are available for use only in the simulation in which they were generated, even if there is more than one opened simulation in DWSIM. Nevertheless, the user can export these components to a file and import them into another simulation.*

### 3. Compound Creator

For detailed information about the Compound Creator, visit: Using the Compound Creator Utility (DWISM Wiki)

### 4. Data Regression

For detailed information about the Data Regression Utility, visit: Using the Data Regression Utility (DWSIM Wiki)

## 5. General Settings

The application settings can be accessed through the **Edit > General Settings** menu item (Figure 53):

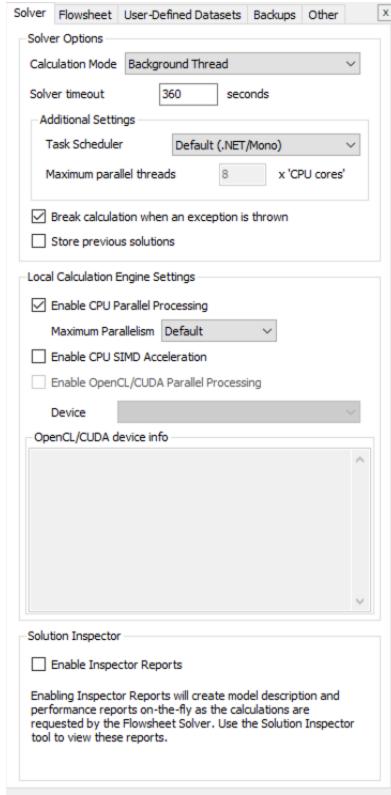


Figure 53: General Settings section.

### 5.1. Solver

The Solver configuration tab display a group of settings to control the behavior of DWSIM's solver. Check the Wiki article [Solver Configuration](#) for more details.

### 5.2. Flowsheet

#### 5.2.1. Cut/Copy/Paste Flowsheet Objects

- **Compounds:** controls how compounds are handled during cut/copy/paste operations.
- **Property Packages:** controls how Property Packages are handled during cut/copy/paste operations.

#### 5.2.2. Undo/Redo

- **Recalculate flowsheet:** defines if the flowsheet is to be recalculated after undo/redo operations.

#### 5.2.3. Object Editors

- **Enable multiple editors:** allows displaying of multiple object editors at once.

- **Close editors on deselecting:** closes the editors once the object being edited is deselected.
- **Default initial placement:** default location for displaying the object editors.

### 5.3. Datasets

In the database tab, you have options to remove, add and edit user-defined compound and interaction parameter datasets.

### 5.4. Backup

The Backup tab has options to control the frequency of the backup file saving. You can also configure the option to save an existing file with another name instead of overwriting it.

### 5.5. Other

#### 5.5.1. Messages

- **Show tips:** displays context-sensitive tips on the flowsheet information (log) window.
- **Show "What's New":** displays a window with information about what's new on the running version.

#### 5.5.2. Debug mode

- **Debug level:** controls the amount of information written to the flowsheet information (log) window when solving the simulation.
- **Redirect console output:** redirects the output of the console to the console window inside DWSIM.

#### 5.5.3. UI Language

- **Language:** sets the UI language. Requires a restart.

#### 5.5.4. CAPE-OPEN

- **Remove solid phases...:** This is for ChemSep compatibility. If enabled, DWSIM will hide the solid phase in Material Streams from CAPE-OPEN Unit Operations.

#### 5.5.5. Compound Constant Properties

- **Ignore compound constant properties...:** If enabled, this will prevent DWSIM from using compound constant data from the loaded simulation files and use the data from the compound databases themselves.

#### 5.5.6. DWSIM/Octave Bridge Settings

- **Octave Binaries Path:** Set the path where the GNU Octave binaries are located. This is only required if you're running DWSIM on Windows.
- **Octave Process Timeout:** Set the timeout for the Octave processes, in minutes.

### 5.5.7. DWSIM/Python Bridge Settings

- **Python Binaries Path:** Set the path where the GNU Octave binaries are located. This is only required if you're running DWSIM on Windows.
- **Python Process Timeout:** Set the timeout for the Octave processes, in minutes.

## Part III.

# Cross-Platform User Interface (CPU)

## 6. Main Interface

When DWSIM is opened, you'll see a launcher menu, where you can create a new simulation or open a saved one, among other functions.

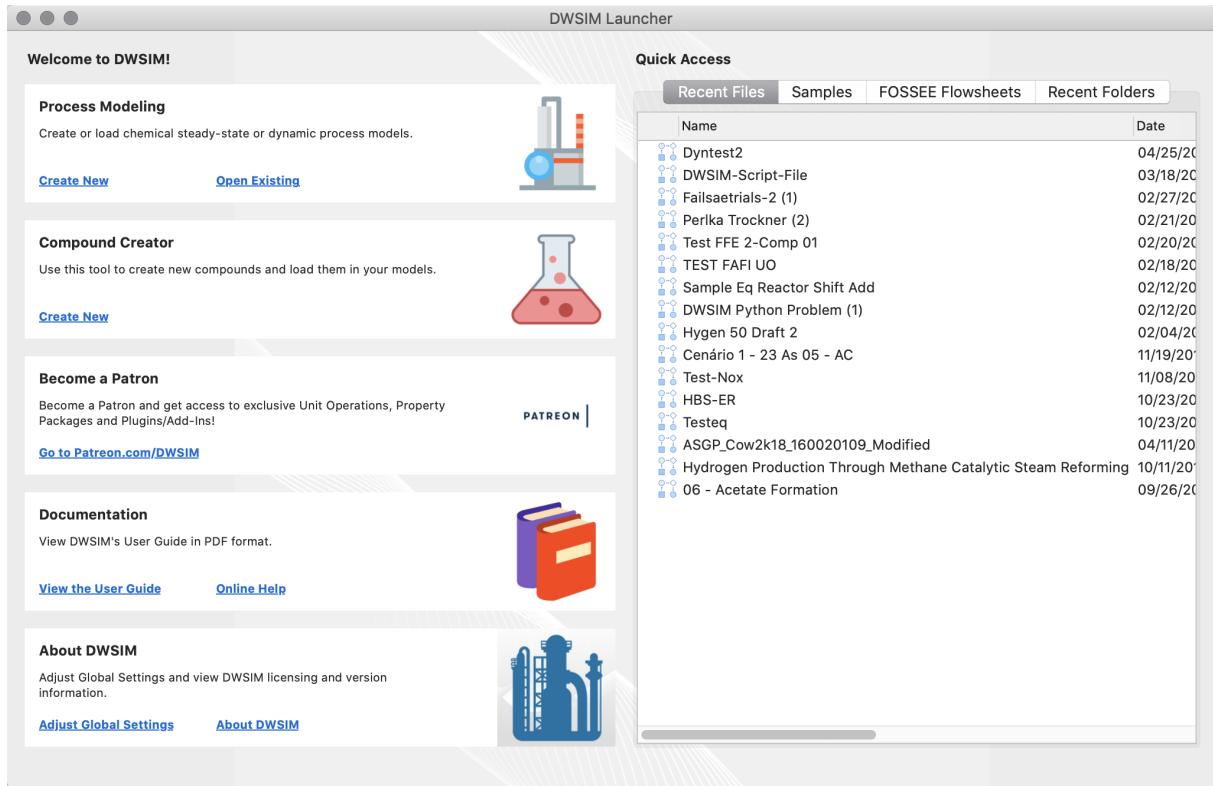


Figure 54: DWSIM Launcher.

If you create a new simulation, you'll get a mostly blank screen with a menu bar at the top and a Log Panel on the bottom part. The **Simulation Setup Wizard** will then appear.

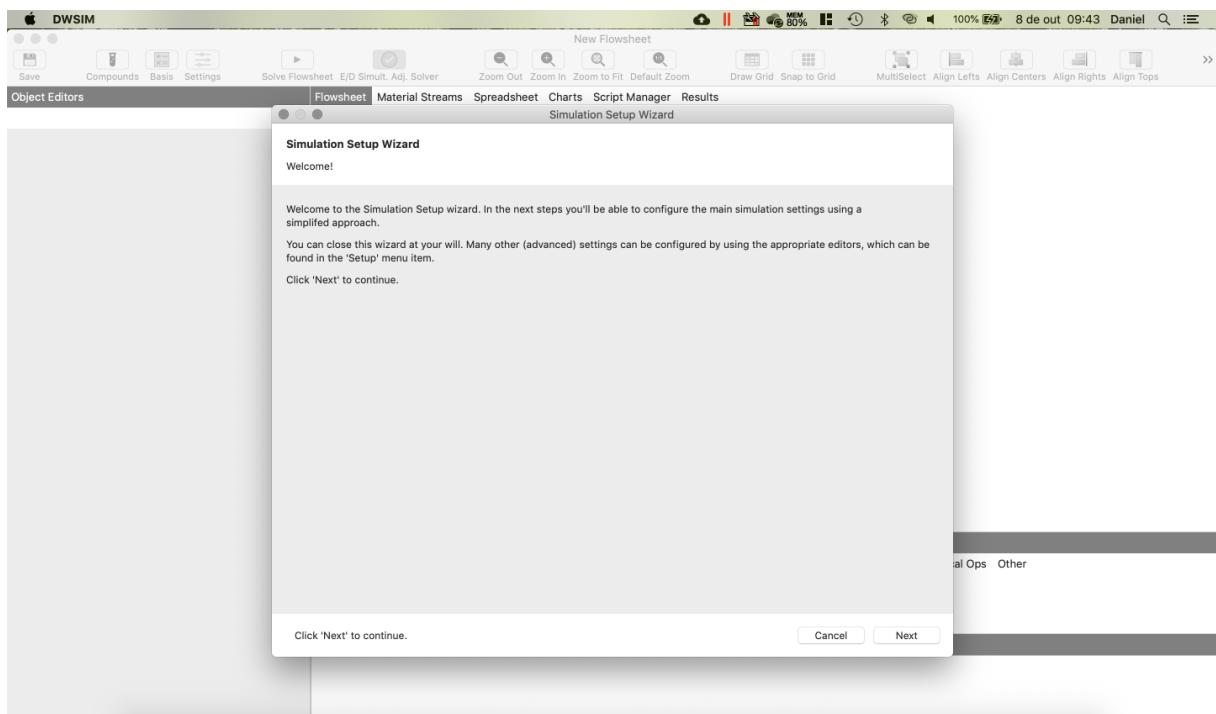


Figure 55: DWSIM Main Flowsheet Window with Simulation Setup Wizard.

## 7. Configuring a Simulation

In order to run a simulation/flowsheet, you need to add some Compounds, setup a Property Package, add Objects to the Flowsheet and connect them to each other following the process flow.

### 7.1. Components/Compounds

There are two essential information required by DWSIM in order to correctly start a simulation. The first refers to the available components (or compounds).

To add a compound to the simulation, select it on the list. To remove an added compound, just deselect it.

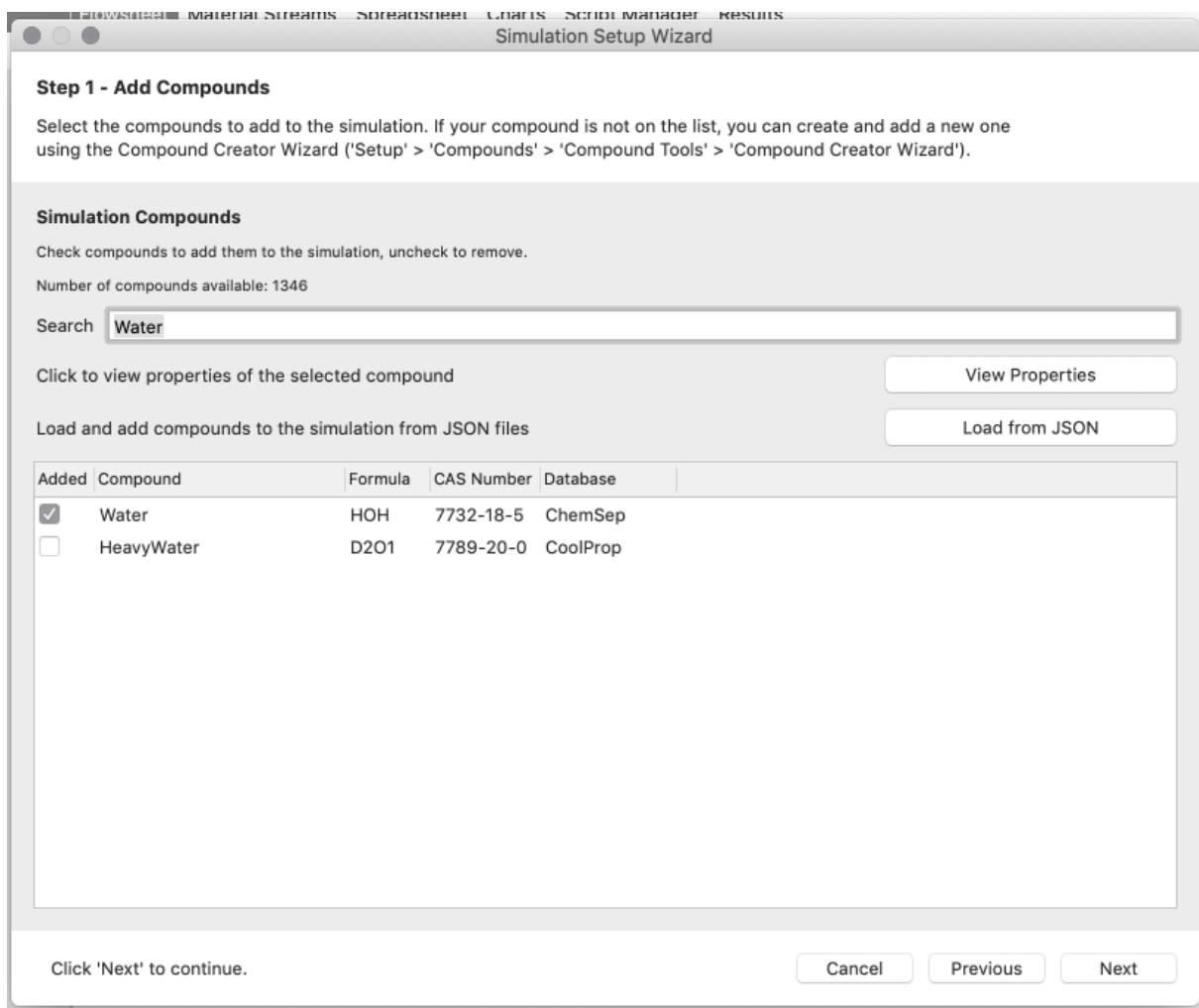


Figure 56: Selecting a Compound with the Simulation Setup Wizard.

## 7.2. Property Packages

The Property Package consists in a set of methods and models for the calculation of physical and chemical properties of material streams in the simulation. It is composed of a thermodynamic model - an equation of state or a hybrid model - and methods for property calculation, like the surface tension of the liquid phase.

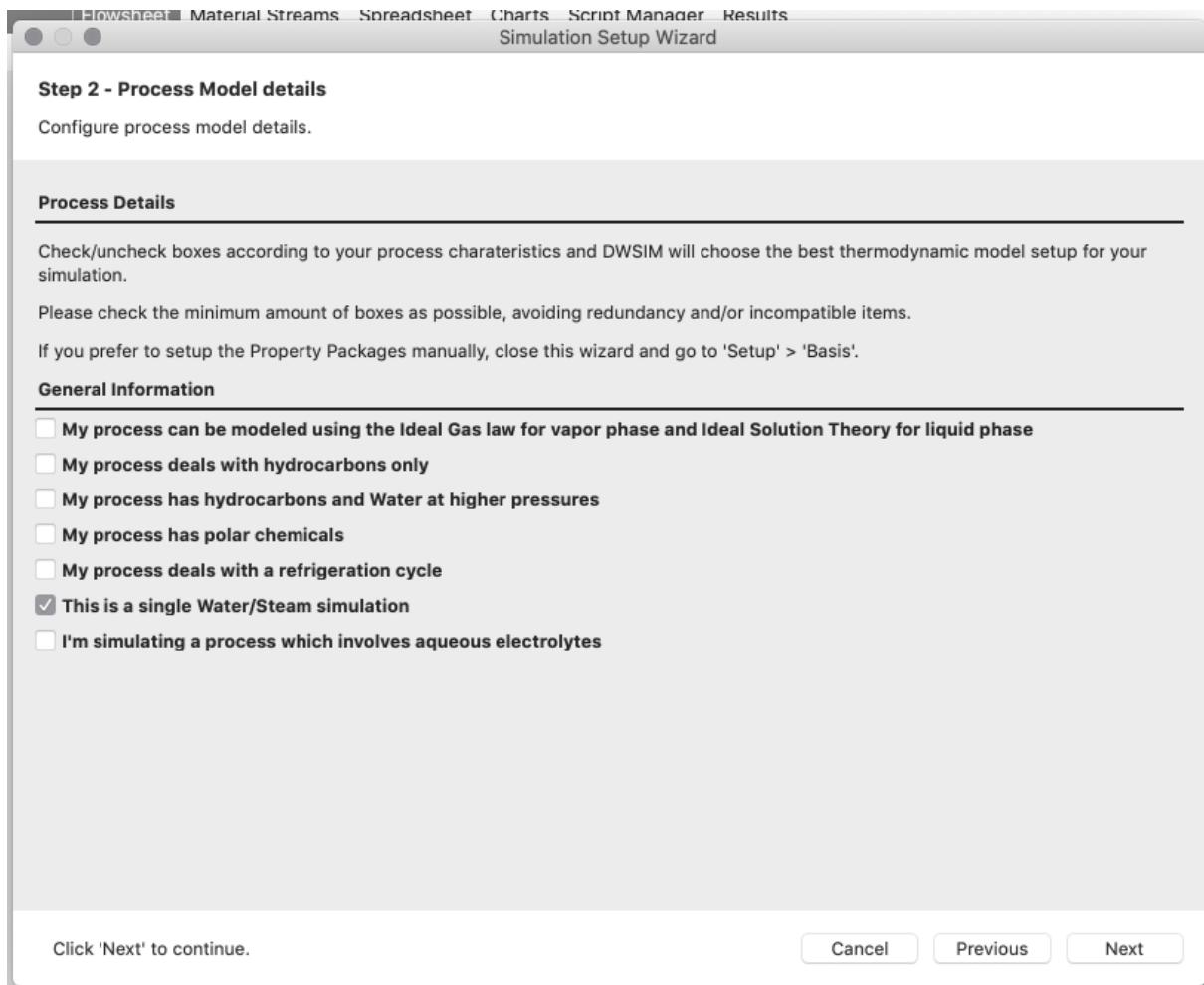


Figure 57: Selecting a Property Package with the Simulation Setup Wizard.

If the selected property package has any editable property, the "Configure" button becomes clickable and the user can click on it to show the property package configuration window.

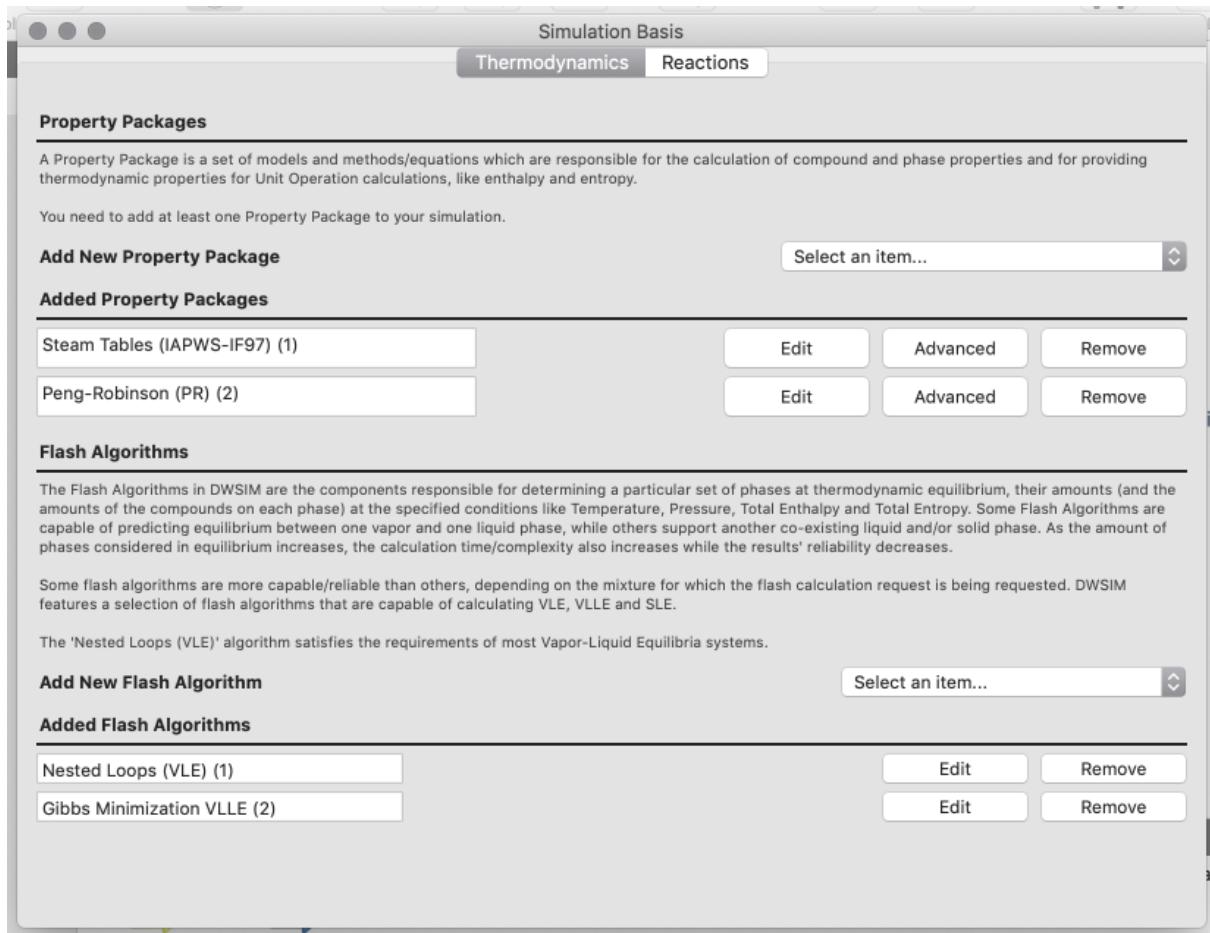


Figure 58: Viewing Property Packages on the Simulation Basis panel.

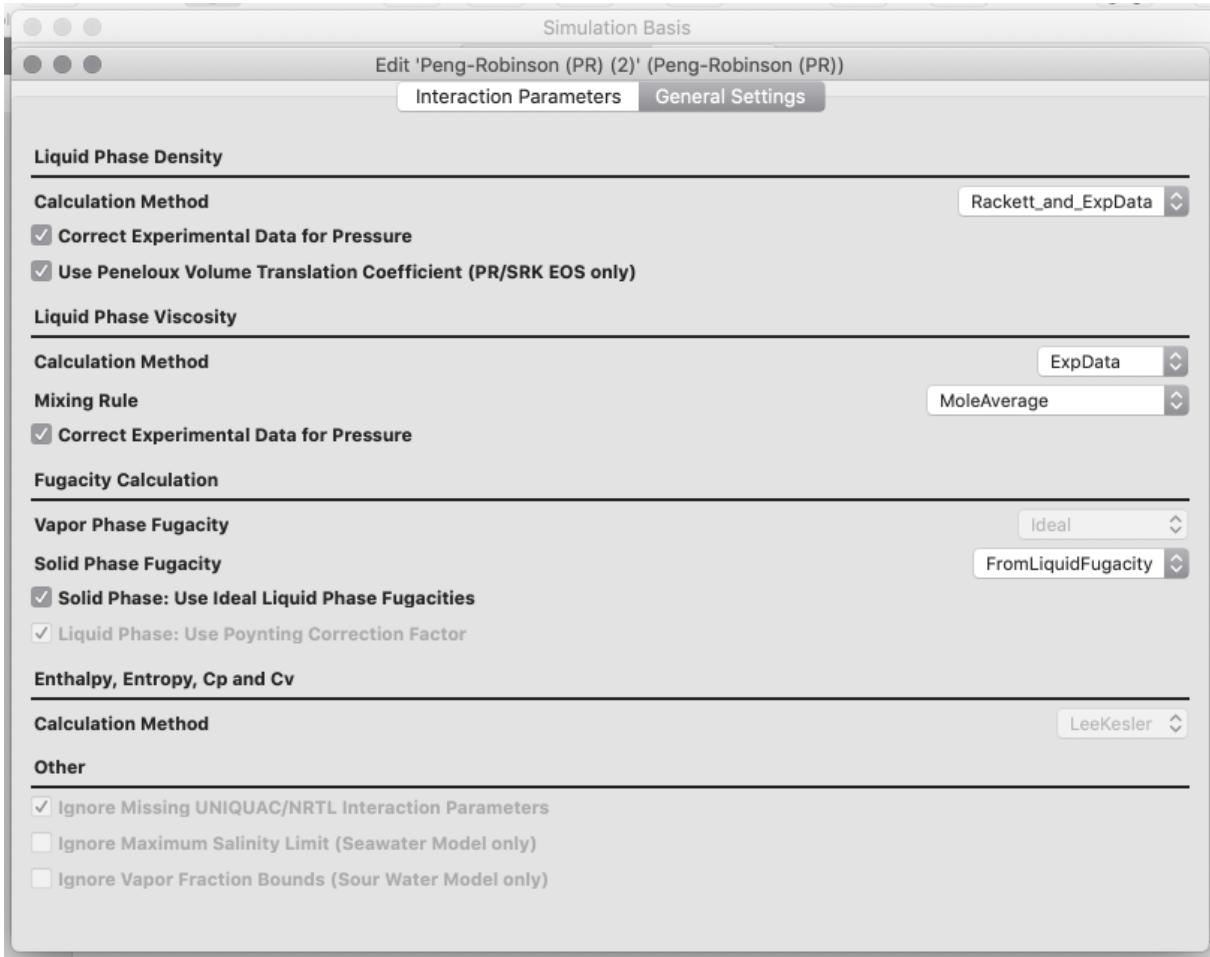


Figure 59: Editing Property Package Settings.

### 7.3. Flash Algorithms

The Flash Algorithms in DWSIM are the components responsible for determining a particular set of phases at thermodynamic equilibrium, their amounts (and the amounts of the compounds on each phase) at the specified conditions like Temperature, Pressure, Total Enthalpy and Total Entropy. Some Flash Algorithms are capable of predicting equilibrium between one vapor and one liquid phase, while others support another co-existing liquid and/or solid phase. As the amount of phases considered in equilibrium increases, the calculation time/complexity also increases while the results' reliability decreases. Some flash algorithms are more capable/reliable than others, depending on the mixture for which the flash calculation request is being requested. DWSIM features a selection of flash algorithms that are capable of calculating VLE, VLLE and SLE. They are:

1. **Nested Loops (VLE):** recommended for the vast majority of VLE systems;
2. **Nested Loops (VLLE):** recommended for systems where the liquid phase may be unstable (will split in two liquid phases with different compositions);
3. **Inside-Out (VLE/VLLE):** recommended for petroleum simulations with many pseudocomponents. The 3-phase option must be used when a second liquid phase is expected (i.e. free water);
4. **Gibbs Minimization (VLE/VLLE):** recommended for difficult chemical systems;

5. **Nested Loops for Eutectic Solid Systems (SLE-E):** Calculates Solid-Liquid Equilibria for eutectic systems considering the solid phase as being ideal.
6. **Nested Loops for Solid Solution Systems (SLE-SS):** Calculates Solid-Liquid Equilibria for solid solution systems considering the solid phase as being ideal.
7. **Nested Loops (3-phase immiscible VLLE):** For systems with an immiscible second liquid phase (VLLE). The first compound selected in the key compound list will be the immiscible one.

#### Multiple Property Packages and Flash Algorithms

DWSIM allows multiple Property Packages and Flash Algorithms to be added to a single simulation (Figure 58), which can be associated with each unit operation and material stream on an individual basis.

## 7.4. Systems of Units

Three basic units systems are present in DWSIM: **SI System** (selected by default), **CGS System** and **English (Imperial) System**. The simulation's units system can be viewed/modified in the "Units System" section of the "Simulation Settings" panel.



Figure 60: Viewing the Systems of Units on the Flowsheet Settings Panel.

You can also create a custom system of units. It is worth remembering that the units systems can also be modified at any time during the simulation - the changes are applied immediately.

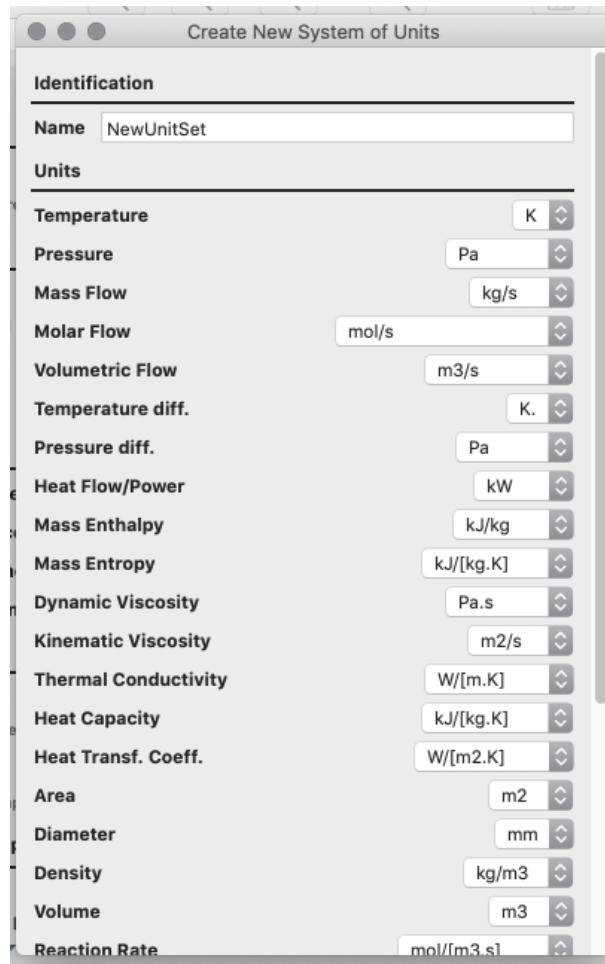


Figure 61: Creating a new System of Units.

## 8. Process Modeling (Flowsheeting)

### 8.1. Inserting Flowsheet Objects

To add an object to the flowsheet, go to 'Object' > 'Add New Simulation Object', or drag the icons from the **Object Palette** to the PFD.

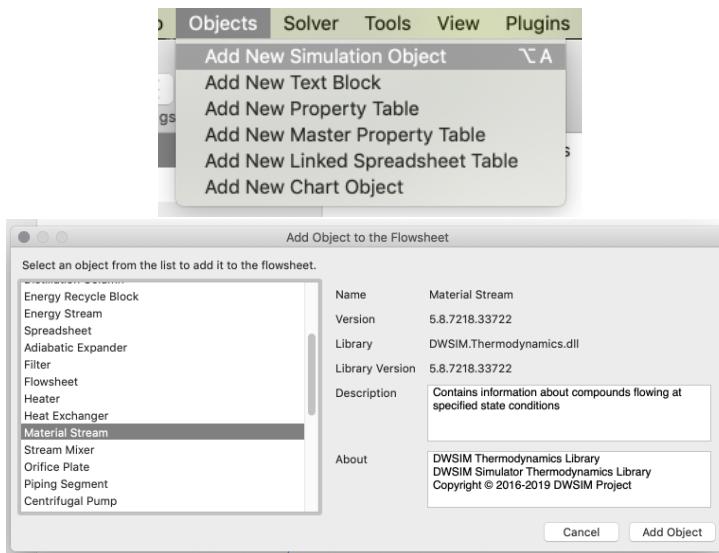


Figure 62: Add New Simulation Object.

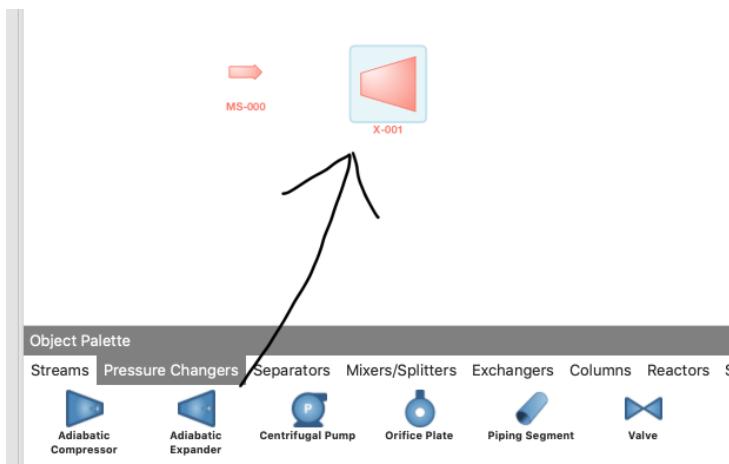


Figure 63: Dragging Objects from the Object Palette to the Flowsheet PFD.

The elements of a simulation (objects) which can be added to the flowsheet are:

- **Material Stream:** used to represent matter which enters and leaves the limits of the simulation and passes through the unit operations. The user should define their conditions and composition in order for DWSIM to calculate their properties accordingly;
- **Energy Stream:** used to represent energy which enters and leaves the limits of the simulation and passes through the unit operations;
- **Mixer:** used to mix up to three material streams into one, while executing all the mass and energy balances;
- **Splitter:** mass balance unit operation - divides a material stream into two or three other streams;
- **Valve:** works like a fixed pressure drop for the process, where the outlet material stream properties are calculated beginning from the principle that the expansion is an isenthalpic process;

- **Pipe:** simulates a fluid flow process (mono or two-phase). The pipe implementation in DWSIM provides the user with various configuration options, including heat transfer to environment or even to the soil in buried pipes. Two correlations for pressure drop calculations are available: Beggs and Brill and Lockhart and martinelli. Both reduces to Darcy equation in the case of single-phase flow;
- **Pump:** used to provide energy to a liquid stream in the form of pressure. The process is isenthalpic, and the non-idealities are considered according to the pump efficiency, which is defined by the user;
- **Separator Vessel:** used to separate the vapor and liquid phases of a stream into two other distinct streams;
- **Compressor:** used to provide energy to a vapor stream in the form of pressure. The ideal process is isentropic (constant entropy) and the non-idealities are considered according to the compressor efficiency, which is defined by the user;
- **Expander:** the expander is used to extract energy from a high-pressure vapor stream. The ideal process is isentropic (constant entropy) and the non-idealities are considered according to the expander efficiency, which is defined by the user;
- **Heater:** simulates a stream heating process;
- **Cooler:** simulates a stream cooling process;
- **Conversion Reactor:** simulates a reactor where conversion reactions occur;
- **Equilibrium Reactor:** simulates a reactor where equilibrium reactions occur;
- **PFR:** simulates a Plug Flow Reactor (PFR);
- **CSTR:** simulates a Continuous-Stirred Tank Reactor (CSTR);
- **Shortcut Column:** simulates a simple distillation column with approximate results using shorcut calculations;
- **Distillation Column:** simulates a distillation column using rigorous thermodynamic models;
- **Absorption Column:** simulates an absorption column using rigorous thermodynamic models;
- **Heat Exchanger:** simulates a countercurrent heat exchanger using rigorous thermodynamic models.
- **Component Separator:** model to simulate a generic process for component separation.
- **Solids Separator:** model to simulate a generic process for solid compound separation.

Additionally, the following logical operations are available in DWSIM:

- **Adjust:** used to make a variable to be equal to a user-defined value by changing the value of other (independent) variable;
- **Recycle:** used to mix downstream material with upstream material in a flowsheet.

## 8.2. Connecting/Disconnecting objects

The material and energy streams represent mass and energy flowing between unit operations. You can connect/disconnect streams to/from Unit Operations or Logical Blocks by selecting the object and working with the Combo Boxes on the "Connections" panel within the Object Editor.

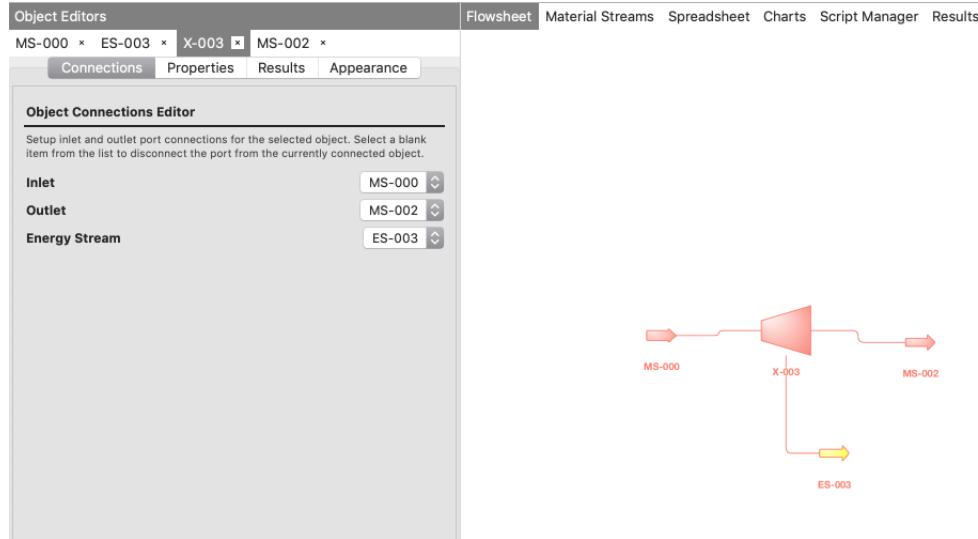


Figure 64: Editing the Connections of an Object.

## 8.3. Process data management

### 8.3.1. Entering process data

The objects' process data (temperature, pressure, flow, composition and/or other parameters) can be entered in the property editor window, accessible through the 'Edit Properties' context menu item.

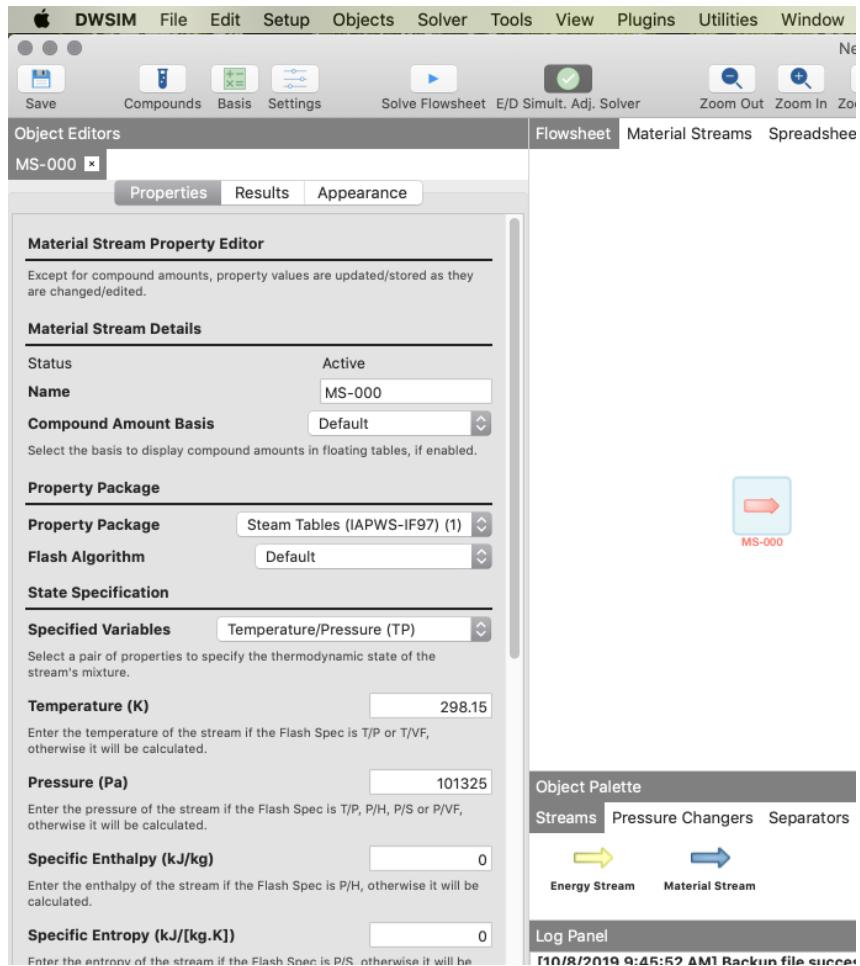


Figure 65: Entering process data.

## 8.4. Running a Simulation

DWSIM is a sequential modular process simulator, that is, all calculations are made in a per-module basis, according to the connections between the objects. The calculator checks if an object has all of its properties defined and, if yes, passes the data for the downstream object and calculates it, repeating the process in a loop until it reaches an object that doesn't have any of its downstream connections attached to any object. This way, the entire flowsheet can be calculated as many times as necessary without having to "tell" DWSIM which object must be calculated. In fact, this is done indirectly if the user define all the properties and make all connections between objects correctly.

To solve the flowsheet, press F5 or click on 'Solver' > 'Solve Flowsheet'. As DWSIM's solver does its job, messages are shown in the log panel. These messages tell the user if the object was calculated successfully or if there was an error while calculating it.

If the calculation finishes without errors, you can proceed to viewing the results.

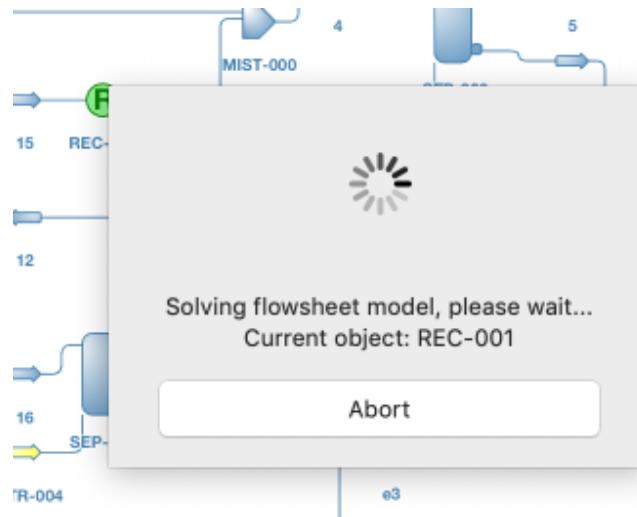


Figure 66: The Flowsheet Solver doing its work.

## 8.5. Viewing Results

Simulation Results can be viewed either in full reports in PDF/ODS/ODT format, or individually through the Results tab, by selecting the object on the list, or by going to the Results tab on the Object Editor, if it has already been calculated.

**Results**

Select Object / View Results Report

Last successful flowsheet calculation: 10/8/2019 9:49:03 AM

Calculation results for Material Stream MS-000		
Compounds:	{ Water, }	
Specification:	Temperature and Pressure	
Temperature:	298.15 K	
Pressure:	101325 Pa	
Property Package:	Steam Tables (IAPWS-IF97)	
Stream Temperature	298.15	K
Stream Pressure	101325	Pa
Stream Enthalpy	0.104929	kJ/kmol
Stream Entropy	0.000367231	kJ/[kmol.K]
Vapor Phase Molar Fraction	0	
Liquid Phase 1 Molar Fraction	1	
Liquid Phase 2 Molar Fraction	0	
Solid Phase Molar Fraction	0	
[Liquid Phase] Mass Flow	1	kg/s
[Liquid Phase] Molar Flow	55.5084	mol/s
[Liquid Phase] Volumetric Flow	0.00100296	m <sup>3</sup> /s
[Liquid Phase] Phase Mole Fraction	1	
[Liquid Phase] Phase Mass Fraction	1	
[Liquid Phase] Water Mole Frac	1	
[Liquid Phase] Water Mass Frac	1	
[Liquid Phase] Water Mole Flow	55.5084	mol/s
[Liquid Phase] Water Mass Flow	1	kg/s
[Liquid Phase] Molecular Weight	18.0153	kg/kmol
[Liquid Phase] Compressibility Facto...	0.000738579	
[Liquid Phase] Isothermal Compressibl...	-9.86877	1/Pa
[Liquid Phase] Bulk Modulus	-0.10133	Pa
[Liquid Phase] Joule Thomson Coeffic...	-1.12703E-10	K/Pa
[Liquid Phase] Speed of Sound	NaN	m/s
[Liquid Phase] Volume	0.0180686	m <sup>3</sup> /kmol
[Liquid Phase] Density	997.048	kg/m <sup>3</sup>

**Log Panel**

- [10/8/2019 9:48:52 AM] Backup file successfully saved to '/Users/Daniel/Documents/DWSIM Application Data/Backup/10\_8\_2019\_9\_42\_52\_AM.dwxmz'.
- [10/8/2019 9:49:02 AM] The flowsheet is being calculated, please wait...
- [10/8/2019 9:49:02 AM] The flowsheet was calculated successfully.
- [10/8/2019 9:49:02 AM] Last run execution time: 0:00:00.239462

Figure 67: The Results Panel.

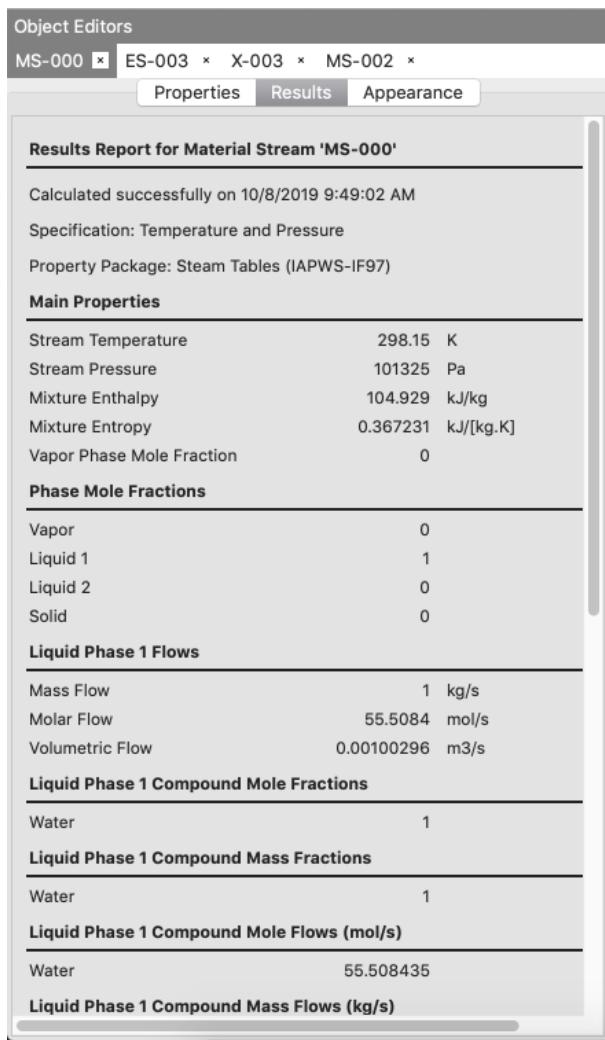


Figure 68: Viewing results from a single object.

## 8.6. Flowsheet Utilities

DWSIM includes some utilities which provides the user with more information about the process being simulated.

- True Critical Point - utility to calculate the true Critical Point of a mixture (PR/SRK EOS only)
- Phase Envelope - Material Stream phase equilibria envelope calculation
- Binary Envelope - special envelopes for binary mixtures
- Pure Compound Properties Viewer

Utilities calculate their properties for a single stream only. In the majority of cases, this object must be calculated in order to be available for selection in the utility window.

### 8.6.1. Pure Component Properties

The Pure Component Properties utility is used to view and edit pure component constants, view molecular properties and general temperature dependent properties like ideal gas Cp, vapor pressure, liquid

viscosity and vaporization enthalpy.

Compound Properties: Water	
	Constant Molecular Liquid Vapor Solid
<b>Constant Properties</b>	
Database	ChemSep
ID	1921
CAS Number	7732-18-5
Molecular Weight (kg/kmol)	18.0153
Critical Temperature (C)	373.99
Critical Pressure (kPa)	22064
Critical Volume (m3/kmol)	0.05595
Critical Compressibility	0.229
Acentric Factor	0.344
Ideal Gas Enthalpy of Formation at 25 °C (kJ/kg)	-13422.7
Ideal Gas Gibbs Energy of Formation at 25 °C (kJ/kg)	-12688.7
Normal Boiling Point (C)	100
Temperature of Fusion (C)	0
Enthalpy of Fusion @ Tf (kJ/mol)	6.00174
Chao-Seader Acentric Factor	0.328
Chao-Seader Solubility Parameter	0.0114199
Chao-Seader Liquid Molar Volume (mL/mol)	18.0674
Rackett Compressibility	0.2338
Peng-Robinson Volume Translation Coefficient	0
SRK Volume Translation Coefficient	0
UNIQUAC R	0.92
UNIQUAC Q	1.4
[Electrolyte] Charge	
[Electrolyte] Hydration Number	0
[Electrolyte] Positive Ion	
[Electrolyte] Negative Ion	
[Electrolyte] Solid Density Temperature (C)	-273.15
[Electrolyte] Solid Density @ Ts (kg/m3)	0
[Electrolyte] Standard State Gibbs Energy of Formation at 298 K (kJ/kg)	0
[Electrolyte] Standard State Enthalpy of Formation at 298 K (kJ/kg)	0

Figure 69: Pure Compound Property Viewer.

### 8.6.2. Phase Envelope

The Phase Envelope utility allows the visualization of the existing relations between thermodynamic properties of a mixture of components in a material stream. The following phase envelopes can be generated in DWSIM: Pressure-Temperature, Pressure-Enthalpy, Pressure-Entropy, Pressure-Volume, Temperature-Pressure, Temperature-Enthalpy, Temperature-Entropy, Temperature-Volume, Volume-Pressure, Volume-Temperature, Volume-Enthalpy and Volume-Entropy.

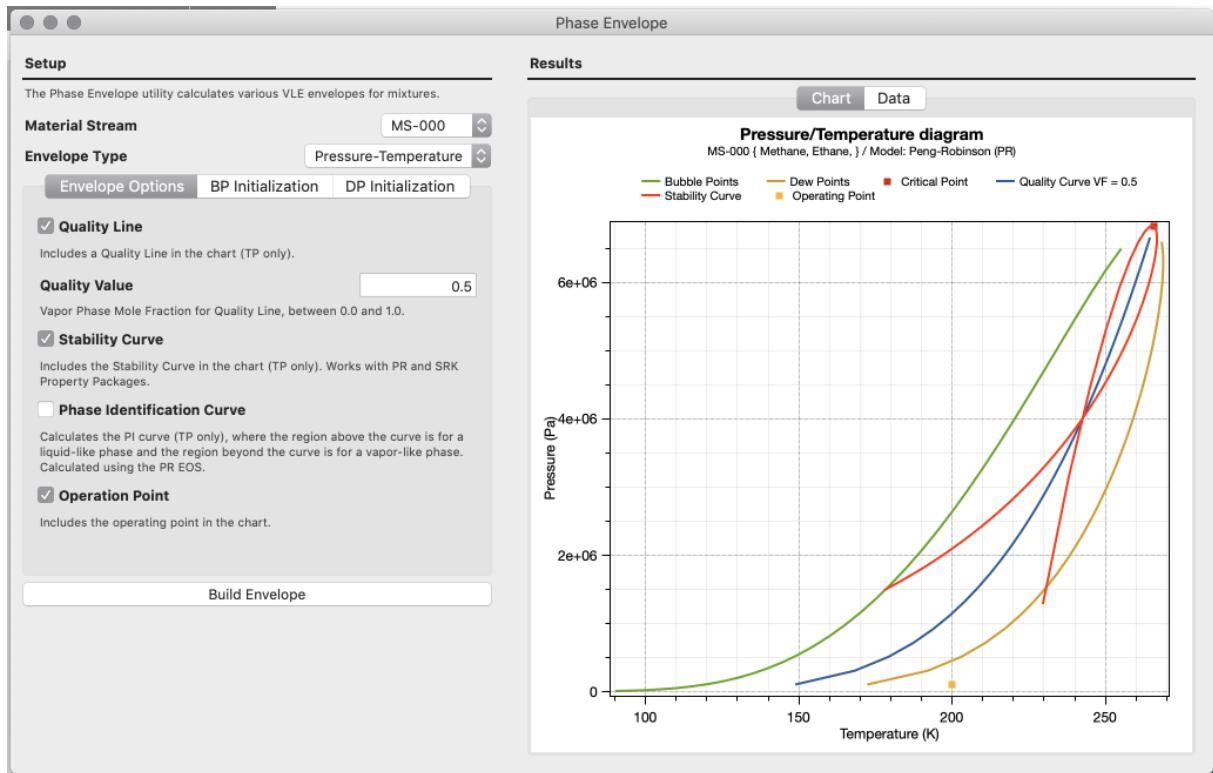


Figure 70: Phase Envelope Utility.

### 8.6.3. Binary Envelope

The Binary Envelope utility is a specialized phase envelope builder for viewing specific two-component diagrams (T-x-y, P-x-y, etc.). For the T-x-y diagram type, different equilibrium lines can be calculated, depending on Property Package and Flash Algorithm selections.

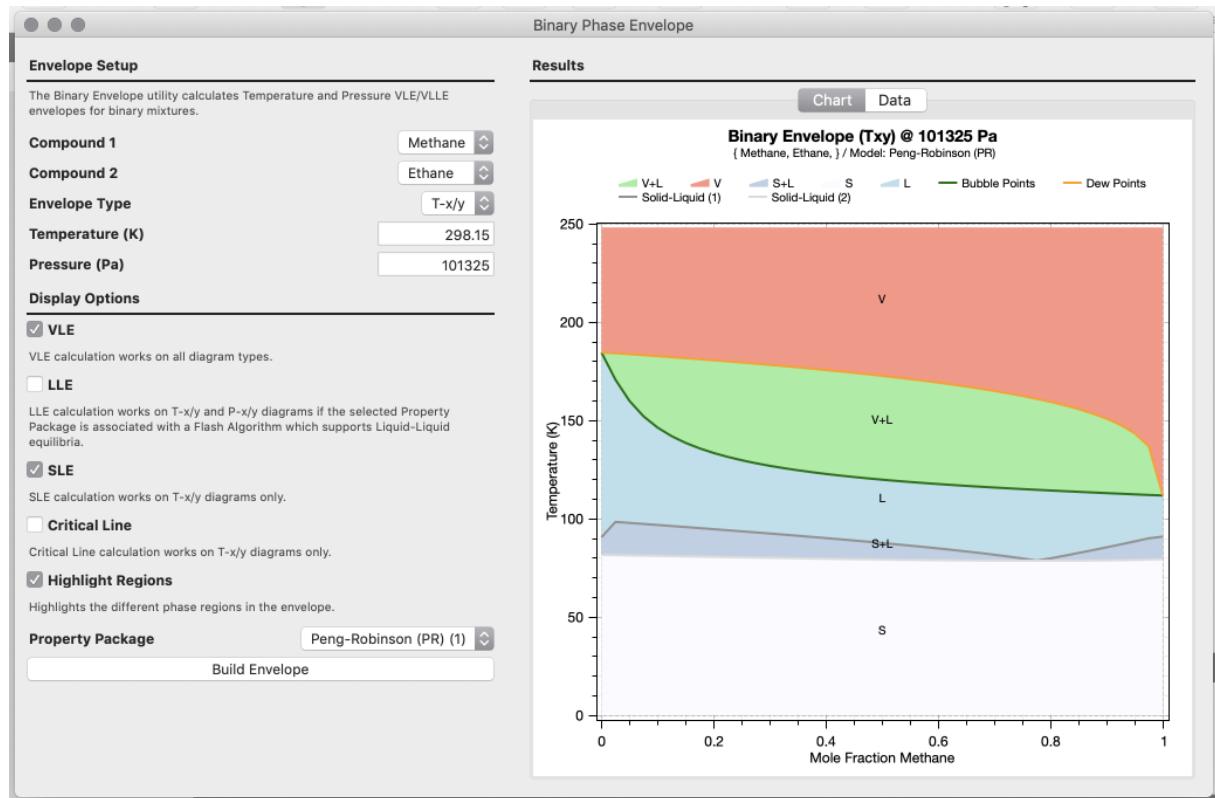


Figure 71: Binary Envelope Utility.

# Part IV.

# Unit Operation Models

## 9. Streams

### 9.1. Material Stream

The Material Stream is used to represent matter which enters and leaves the limits of the simulation, passing through the unit operations. Input Parameters Specification: One of the following flash types may be selected:

- Temperature and Pressure
- Pressure and Enthalpy
- Pressure and Entropy
- Pressure and Vapor Fraction
- Temperature and Vapor Fraction

The above selection dictates which variables should be defined in order to enable DWSIM calculate the others:

- Temperature: stream temperature
- Pressure: stream pressure Specific
- Enthalpy: stream's specific enthalpy Specific
- Entropy: stream's specific entropy
- Molar Fraction (Vapor Phase): the vapor phase mole fraction of the stream

**9.1.0.1. Composition** The stream composition can be entered on basis of: Mole/Mass fraction/flow Standard liquid volumetric fraction Molality or molarity The material stream composition can only be edited if the stream is not connected to any unit operation upstream, which means that it doesn't work as an output of any operation. If that is the case, the stream configures itself as "read-only" and the user will not be able to edit any of its properties directly. These are written into the stream as a result of calculation from its connected unit operation.

For the mole/mass fraction basis options, the user must enter the composition in such a way that the amounts sum up to 1. Regardless of the input basis option, new values are saved only when the "Apply / Commit Changes" button is pressed.

Molarity and Molality input composition options only make sense in the context of an electrolyte simulation. In this case, the solute amounts should be entered in moles and the solvent amount in L (molarity) or kg (molality).

**9.1.0.2. Flow** One of the three types of flow (mass, molar or volumetric) must be given. The other two are calculated automatically in the case when temperature and pressure are already defined. If composition is defined by either mole or mass flow of each component, the total flow of the stream is calculated automatically. If a definition of composition is utilised, you will have to define flowrate of the stream on stream level outside of composition dialog.

### 9.1.1. Calculation Method

When the four properties described above are defined, the material stream is calculated and its properties are shown in the same window. The calculation sequence for the material stream is the following:

1. A flash calculation is done to know the component distribution between phases
2. Properties of each phase are calculated individually
3. Finally, the mixture properties are calculated

In the first step, a TP Flash is done by default, but it can be replaced by a PH flash if the user defines this option in the simulation configuration window. When in "read-only" mode, the stream properties are calculated according to parameters given by the upstream unit operation (in the majority of cases, a TP flash is done as well).

### 9.1.2. Output parameters

The following items are calculated:

- Component distribution between phases. The compositions can be displayed either as molar or mass fraction by selecting the correspondent option at Composition basis drop down field.
- Phase properties: specific enthalpy, specific entropy, molecular weight, density, volumetric flow rate @ T and P, phase molar and mass fraction, compressibility factor, constant-pressure heat capacity ( $C_p$ ),  $C_p/C_v$ , thermal conductivity, surface tension (liquid phase only), kinematic viscosity, dynamic viscosity.
- Mixture properties: specific enthalpy, specific entropy, molecular weight, density, thermal conductivity.

If the flash algorithm setting "Calculate bubble and dew points at stream conditions" is activated, these are also shown in the "Mixture" properties section.

## 9.2. Energy Stream

The energy stream is used to represent energy entering and leaving the limits of the simulation, used by the unit operations, either to represent loss, demand or power generation. As we are dealing with steady-state simulations, one defines the energy stream in terms of power (energy by unit of time) and not energy itself.

The energy either is written by user to be used for energy consuming unit operations or from its connected unit operation as a result of its calculations.

**9.2.0.1. Input Parameters** Energy: Energy by unit of time (power) which is represented by the stream;

**9.2.0.2. Output Parameters** There are no output parameters for this object.

## 10. Unit Operations

### 10.1. Mixer

The Mixer is used to mix up to six material streams into one, while executing all the mass and energy balances.

#### ***Input Parameters***

- Downstream pressure: defines how the downstream pressure must be calculated from the pressure of the streams connected to the mixer inlets.

***Calculation Method*** The mixer does the mass balance in the equipment and determines the mass flow and the composition of the outlet stream. Pressure is calculated according to the parameter defined by the user. Temperature is calculated by doing a PH Flash in the outlet stream, with the enthalpy calculated from the inlet streams (energy balance).

***Output Parameters*** There are no output parameters for this object.

### 10.1.1. Splitter

The splitter is a mass balance unit operation - divides a material stream into two or three other streams.

#### ***Input Parameters***

- Stream fractions: this property defines the mass flow fraction to be passed to each outlet stream in the splitter. Each fraction must have a value between 0 and 1 and the total sum must not be bigger than 1.

***Output Parameters*** There are no output parameters for this object.

### 10.2. Separator Vessel

The separator vessel (also known as *flash drum*) is used to separate liquid phases from vapor in a mixed material stream.

#### ***Input Parameters***

- Override separation temperature and pressure: these properties define if the flash calculation will be done with the temperature and pressure of the inlet stream or the ones defined by the user.

**Calculation Method** The separator vessel just divides the inlet stream phases into two or three distinct streams. If the user defines values for the separation temperature and/or pressure, a TP Flash is done in the new conditions before the distribution of phases through the outlet streams.

**Output Parameters** There are no output parameters for this object.

### 10.3. Tank

In the current version of DWSIM, the Tank works like a fixed pressure drop for the process.

#### **Input Parameters**

- Pressure drop: pressure difference between the outlet and inlet streams.

**Calculation Method** The outlet stream pressure is calculated from the inlet pressure and the pressure drop given. The temperature remains the same. A TP Flash is done to calculate the properties of the outlet stream.

**Output Parameters** There are no output parameters for this object.

### 10.4. Pipe Segment

The Pipe Segment unit operation can be used to simulate fluid flow process in a pipe. Two of the most used correlations for the calculation of pressure drop are available in DWSIM. Temperature can be rigorously calculated considering the influence of the environment. With the help of the Recycle Logical Operation, the user can build large water distribution systems, as an example.

The pipe segment is divided in sections, which can be straight tubes, valves, curves, etc. Each section is subdivided in small sections for calculation purposes, as defined by the user.

#### **Input Parameters**

- Hydraulic profile: clicking on the ellipsis button opens the pipe hydraulic profile editor. In the hydraulic profile editor, the user adds sections, define their type and in how many increments it will be divided during the calculations, the pipe material, length, elevation and internal and external diameters. Each change can be saved by clicking in the "Apply" button.
- Pressure drop correlation: select the model to be used for the pressure drop calculation in the pipe segment.
- Thermal profile: In the thermal profile editor it is possible to define how the temperature profile in the pipe should be calculated. The configurations in this window are valid for the **entire** pipe segment. Changes are saved automatically.

**Calculation Method** The pipe segment is calculated based on incremental mass and energy balances. The complete algorithm consists in three nested loops. The external loop iterates on the sections (increments), the middle loop iterates on the temperature and the internal loop calculates the pressure. The pressure and temperature are calculated as follows:

1. The inlet temperature and pressure are used to estimate the increment outlet pressure and temperature.
2. Fluid properties are calculated based in a arithmetic mean of inlet and outlet conditions.
3. The calculated properties and the inlet pressure are used to calculate the pressure drop. With it, the outlet pressure is calculated.
4. The calculated and estimated pressure are compared, and if their difference exceeds the tolerance, a new outlet pressure is estimated, and the steps 2 and 3 are repeated.
5. Once the internal loop has converged, the outlet temperature is calculated. If the global heat transfer coefficient ( $U$ ) was given, the outlet temperature is calculated from the following equation:

$$Q = UA\Delta T_{ml} \quad (10.1)$$

where:  $Q$  = heat transferred,  $A$  = heat transfer area (external surface) and  $\Delta T_{ml}$  = logarithmic mean temperature difference.

6. The calculated temperature is compared to the estimated one, and if their difference exceeds the specified tolerance, a new temperature is estimated and new properties are calculated (return to step 2).
7. When both pressure and temperature converges, the results are passed to the next increment, where calculation restarts.

### Output Parameters

- Delta-T: temperature variation in the pipe segment.
- Delta-P: pressure variation in the pipe segment.
- Heat exchanged: amount of heat exchanged with the environment, or lost by friction in the pipe walls.
- Results (table): results are show section by section in a table.
- Results (graph): a graph shows the temperature, pressure, liquid holdup, velocity and heat exchanged profiles.

## 10.5. Valve

The Valve works like a fixed pressure drop for the process, where the outlet material stream properties are calculated beginning from the principle that the expansion is an isenthalpic process.

### **Input Parameters**

- Pressure drop: pressure difference between the outlet and inlet streams.

**Calculation Method** The outlet stream pressure is calculated from the inlet pressure and the pressure drop. The outlet stream temperature is found by doing a PH Flash. This way, in the majority of cases, the outlet temperature will be less than or equal to the inlet one.

### **Output Parameters**

- Delta-T: temperature drop observed in the valve expansion process.

## **10.6. Pump**

The pump is used to provide energy to a liquid stream in the form of pressure. The process is isenthalpic, and the non-idealities are considered according to the pump efficiency, which is defined by the user.

### **Input Parameters**

- Delta-P: pressure rise in the pump.
- Efficiency: pump adiabatic efficiency;
- Ignore vapor in the inlet stream: defines if the calculator should ignore any vapor in the inlet stream;
- Use the provided Delta-P: defines if the pressure of the outlet stream will be calculated by the user-defined Delta-P or the energy stream connected to the pump.

**Calculation Method** The calculation method for the pump is different for the two cases (when the provided delta-p or the potency of the energy stream is used). In the first method, we have the following sequence:

- Outlet stream enthalpy:

$$H_2 = H_1 + \frac{\Delta P}{\rho}, \quad (10.2)$$

- Pump discharge pressure:

$$P_2 = P_1 + \Delta P \quad (10.3)$$

- Pump required power:

$$Pot = \frac{W (H_2 - H_1)}{\eta}, \quad (10.4)$$

where:

<i>Pot</i>	pump power
<i>W</i>	mass flow
<i>H</i> <sub>2</sub>	outlet stream specific enthalpy
<i>H</i> <sub>1</sub>	inlet stream specific enthalpy
$\eta$	pump efficiency

→ Outlet temperature: PH Flash (with P<sub>2</sub> and H<sub>2</sub>).

In the second case (calculated outlet pressure), we have the following sequence:

→ Outlet stream enthalpy:

$$H_2 = H_1 + \frac{Pot \eta}{W}, \quad (10.5)$$

→  $\Delta P$ :

$$\Delta P = \rho(H_2 - H_1), \quad (10.6)$$

→ Discharge pressure:

$$P_2 = P_1 + \Delta P \quad (10.7)$$

→ Outlet temperature: PH Flash.

### Outlet Parameters

→ Delta-T: temperature variation in the pumping process.

→ Power required: power required by the pump.

## 10.7. Compressor/Expander

The compressor/expander is used to provide or generate energy to/from a vapor phase stream. The ideal process is isentropic (constant entropy) and the non-idealities are considered according to the compressor efficiency and thermodynamic path (adiabatic or polytropic), which are defined by the user.

### Input Parameters

→ Delta-P: pressure change in the equipment.

→ Efficiency: adiabatic/polytropic efficiency;

→ Ignore liquid in the inlet stream: defines if the calculator should ignore any liquid in the inlet stream;

→ Thermodynamic path: select the thermodynamic path according to the experimental/field data available.

**Calculation Method** Isentropic (Adiabatic) or Polytropic power is calculated from:

$$P = \frac{H_{2s} - H_1}{\eta} W \quad (10.8)$$

for compressor, and

$$P = (H_{2s} - H_1) \times W \times \eta \quad (10.9)$$

for expander, where:

$H_{2s}$       Outlet Enthalpy for Isentropic Process

$H_1$       Inlet Enthalpy

$W$       Mass Flow

$\eta$       Adiabatic or Polytropic Efficiency

Isentropic (Adiabatic) and Polytropic Coefficients are calculated from:

$$n_i = \frac{\ln(P_2/P_1)}{\ln(\rho_{2i}/\rho_1)} \quad (10.10)$$

$$n_p = \frac{\ln(P_2/P_1)}{\ln(\rho_2/\rho_1)} \quad (10.11)$$

where:

$\rho_{2i}$       Outlet Gas Density calculated with Inlet Gas Entropy

Adiabatic and Polytropic Heads are calculated from:

$$H = \frac{P \times W \times g}{\eta} \quad (10.12)$$

where:

$H$       Adiabatic or Polytropic Head

$P$       Adiabatic or Polytropic Power

$W$       Mass Flow

$\eta$       Adiabatic or Polytropic Efficiency

$g$       Gravitational Constant (9.8 m/s<sup>2</sup>)

## 10.8. Heater

The heater simulates a stream heating process.

***Input Parameters***

- Pressure drop: pressure drop in the heater.
- Heat added: amount of heat added in the heater.
- Efficiency: heater efficiency.
- Use heat provided: defines if the heat added is determined by the value informed by the user or by the energy stream connected to the heater.

**Calculation Method** The outlet stream temperature is calculated by doing a PH Flash, were the outlet stream enthalpy is calculated by a energy balance in the heater.

***Output Parameters***

- Delta-T: temperature rise observed in the heating process.

**10.9. Cooler**

The cooler simulates a stream cooling process.

***Input Parameters***

- Pressure drop: pressure drop in the cooler.
- Heat removed: amount of heat removed by the cooler.
- Efficiency: cooler efficiency.

**Calculation Method** The outlet stream temperature is calculated by doing a PH Flash, were the outlet stream enthalpy is calculated by a energy balance in the cooler.

***Output Parameters***

- Delta-T: temperature drop observed in the cooling process.

**10.10. Shortcut Column**

The shortcut column is used to calculate the minimum reflux and distribution of products in a distillation column by the method of Fenske-Underwood-Gilliland [1]. The column should have a single feed stage, two products (top and bottom), condenser (partial or total) and reboiler. The results are the minimum reflux, thermal loads and temperature of the condenser and reboiler for a fixed reflux ratio, in addition to determining the optimum feed stage and the minimum number of stages.

**Input Parameters**

- Connections: feed, product, top, bottom and heat loads (condenser / reboiler).
- Type of condenser: partial or total.
- Reflux Ratio: ratio between the flow of liquid that returns from the condenser to the column and the one that leaves the condenser as the top product.
- Light Key: component used as a reference so that the lighter ones are present only in the top product.
- Heavy Key: component used as a reference so that the heavier ones are present only in the product of fund.
- Condenser pressure: pressure of the condenser.
- Reboiler pressure: pressure of the reboiler.

**Output Parameters**

- Minimum reflux: reflux ratio of minimum to ensure the separation specified.
- Minimum number of stages: the minimum number of training which ensures the separation specified.
- Optimal feed stage: the feed stage that minimizes the thermal load of the reboiler.
- Liquid / Vapor flows: internal flows in sections of rectification and stripping of the column.
- Thermal loads: thermal loads of condenser and reboiler.

**10.11. Rigorous Column (Distillation / Absorption)**

The rigorous column is an unit operation that represents the fractionating towers, where components in a mixture are separated in various equilibrium stages. It is called *rigorous* because of the thermodynamic models used in the solution of the mass and energy balances throughout the column.

In DWSIM, four types of rigorous columns are available: **Distillation Column**, **Absorption Column**, **Refluxed Absorber** and **Reboiled Absorber**. All of them share the same basic interface, with the following characteristics:

- Supports multiple feed streams
- Supports multiple side draws
- Supports energy streams representing heat exchangers on each stage
- Definition of pressure and efficiency by stage

**Solving methods** In the current version of DWSIM, three methods are available to solve the mass and energy balances in a column. They are:

- Bubble-Point method of Wang and Henke (distillation column only) [2]
- *Sum-Rates* method of Burningham and Otto (absorption column only)[2]
- Russell's *Inside-Out* method (all column types)[2]
- Simultaneous Correction method of Naphtali and Sandholm (all column types)[2]

**Configuration options** The following table describes the configuration options for each solving method.

Method	Parameter	Options	Default Value	Description
BP	Stop at iteration number	0 to 1000 (-1 to disable)	-1 (disabled)	You can use this option to generate initial estimates for the IO and SC methods.
IO/SC	Step for numerical derivatives	1E-3 to 1E-12	1E-5	Controls the value of change in variables in order to calculate the numerical derivatives.
IO/SC	Maximum variable change factor	>1	10	Limits the changes in variables so the maximum variable change will be (newvalue/oldvalue) = factor.
IO	Adjust Sb scaling factor	True/False	False	Use the Sb scaling factor to adjust the initial material balance estimation. Helps to stabilize the algorithm when poor initial estimates are used.
IO	Calculate Kb by weighted average	True/False	False	Calculates the Kb (reference K) factor for each stage by using the weighted average method as suggested by Russell. If this option is set to false, the highest K-value is used for the relative volatility calculation for each component on each stage.
IO	Use identity matrix as the first jacobian inverse	True/False	False	If True, tells DWSIM to use an identity matrix as the first estimate of the Jacobian inverse, which by its turn is used to update each independent variable. If set to False, the first Jacobian is calculated using numerical derivatives.
IO	Use damping factor	True/False	True	Set this option to True to use a damping factor, which helps to minimize the error on each inner loop iteration.
SC/O	Use Newton's method to update variables	True/False	False	The variables are normally updated by Broyden's method, but setting this option to True helps to stabilize the algorithm on hard-to-converge situations.
SC/O	Store and reuse jacobian matrix	True/False	True	This option is useful when the column must be solved multiple times without significant changes in variables, i. e. when using recycled streams as column feeds.
SC	Damping factor	0 to 2	0.5	Set the damping factor which limits the changes to variables as predicted by the Newton (or Broyden) step.
SC	Maximum allowed stage temperature change	> 0	10 K	Sets the maximum allowable stage temperature change between iterations. Changes higher than this value will not be considered.

Table 1: Column solving methods' configuration parameters.

**Results** After the solution of mass and energy balances for the entire column, the output streams are calculated and the following results are shown in the properties window of the column:

- Heat exchanged in the condenser (where applicable)
- Heat exchanged in the reboiler (where applicable)
- Temperature profiles and flow of liquid / vapor entering at each stage
- Distribution profiles of components on each stage

## 10.12. Heat Exchanger

DWSIM has a model for the countercurrent, two-stream heat exchanger which supports phase change and multiple phases in a stream.

**Input Parameters** The heat exchanger in DWSIM has five calculation modes:

1. Calculate hot fluid outlet temperature: you must provide the cold fluid outlet temperature and the exchange area to calculate the hot fluid temperature.
2. Calculate cold fluid outlet temperature: in this mode, DWSIM needs the hot fluid outlet temperature and the exchange area to calculate the cold fluid temperature.
3. Calculate both temperatures: in this mode, DWSIM needs the exchange area and the heat exchanged to calculate both temperatures.
4. Calculate area: in this mode you must provide the HTC and both temperatures to calculate the exchange area.
5. Rate a Shell and Tube exchanger: in this mode you must provide the exchanger geometry and DWSIM will calculate output temperatures, pressure drop on the shell and tubes, overall HTC, LMTD, and exchange area. This calculation mode uses a simplified version of Tinker's method [3] for Shell and Tube exchanger calculations.

You can provide the pressure drop for both fluids in the exchanger for modes 1 to 4 only.

**Calculation Mode** The heat exchanger in DWSIM is calculated using the simple convection heat equation:

$$Q = UA\Delta T_{ml}, \quad (10.13)$$

where:  $Q$  = heat transferred,  $A$  = heat transfer area (external surface) and  $\Delta T_{ml}$  = Logarithmic Mean Temperature Difference (LMTD). We also remember that:

$$Q = m\Delta H, \quad (10.14)$$

where:  $Q$  = heat transferred from/to the fluid and  $\Delta H$  = outlet-inlet enthalpy difference.

The calculation procedure depends on the mode selected:

1. Calculate hot fluid outlet temperature: HTC (Heat Transfer Coefficient), hot fluid outlet temperature, heat load and LMTD.
2. Calculate cold fluid outlet temperature: HTC, cold fluid outlet temperature, heat load and LMTD.
3. Calculate both temperatures: HTC, cold and hot fluid outlet temperatures and LMTD.
4. Calculate area: exchange area and LMTD.
5. Rate Shell and Tube exchanger: exchanger geometry information.

**Results** The results given by the heat exchanger after calculation depends on the mode selected:

1. Calculate hot fluid outlet temperature: overall HTC, hot fluid outlet temperature, heat load and LMTD.
2. Calculate cold fluid outlet temperature: overall HTC, cold fluid outlet temperature, heat load and LMTD.
3. Calculate both temperatures: overall HTC, cold and hot fluid outlet temperatures and LMTD.
4. Calculate area: exchange area and LMTD.
5. Rate Shell and Tube exchanger: area, LMTD, LMTD correction factor (F), overall HTC, hot fluid outlet temperature, cold fluid outlet temperature, hot fluid pressure drop (shell/tubes only), cold fluid pressure drop (shell/tubes only).

### 10.13. Component Separator

The Component Separator is a mass balance unit operation. The components are separated between two streams, specified as fractions or absolute flow rates. The energy balance is then calculated after the separation.

#### **Input Parameters**

- Specified stream: sets the stream to which the separation specifications will be applied. "0" corresponds to the Outlet stream 1 (overhead) and "1" corresponds to the Outlet stream 2 (bottoms).

#### **Results**

- Energy imbalance: Difference between enthalpy of outlet and inlet streams. in some cases it can be interpreted as the energy necessary to do the separation.

## 10.14. Orifice Plate

This model corresponds to the ISO 5167 specification for Orifice Plates, and can be used to measure flow rates if used in conjunction with the Adjust logical operation (change flow rates until specified pressure drop matches the calculated one).

### **Input Parameters**

- Pressure tappings: select the option which corresponds to the arrangement of the tappings for pressure reading.
- Orifice diameter: inner diameter of the plate.
- Beta (d/D): ratio between plate's inner and outer diameters.
- Correction factor: multiplier for the mass flow rate used in the calculation of the pressure drop across the orifice. Default is 1.

### **Results**

- Orifice pressure drop: Pressure drop across the orifice. This is the value that is read through the tappings.
- Overall pressure drop: Pressure drop after recovery. This values should always be less or equal to the orifice pressure drop.
- Delta T: temperature drop across the orifice, considering that the process is an adiabatic expansion.

## 10.15. Custom Unit Operation

This Unit Operation lets you run scripts as the main calculation routine (inside the Calculate() procedure which is called by the calculator for all flowsheet simulation objects). There are up to six streams available for the UO, three as input and three as output.

Supported Languages are IronPython, IronRuby, VBScript and JScript. You can use some predefined reference variables inside your script, defined as shortcuts to the most common objects:

<b>ims1</b>	Input Material Stream in slot 1 (MaterialStream class instance)
<b>ims2</b>	Input Material Stream in slot 2 (MaterialStream class instance)
<b>ims3</b>	Input Material Stream in slot 3 (MaterialStream class instance)
<b>oms1</b>	Output Material Stream in slot 1 (MaterialStream class instance)
<b>oms2</b>	Output Material Stream in slot 2 (MaterialStream class instance)
<b>oms3</b>	Output Material Stream in slot 3 (MaterialStream class instance)
<b>Me</b>	Reference variable to the Custom UO object instance (CustomUO UnitOperation class)
<b>Flowsheet</b>	Reference variable to the active flowsheet object (FormChild class)

**Solver** Flowsheet solver class instance, used to send commands to the calculator (COMSolver class)

**Spreadsheet** Reference variable to the Spreadsheet object.

## 10.16. Solids Separator

The solids separator is used to separate solids from a liquid phase in a mixed material stream.

### **Input Parameters**

- Solids Separation Efficiency: defines the amount of solids in the liquid stream. 100% efficiency means no solids in the liquid stream.
- Liquids Separation Efficiency: defines the amount of liquid in the solids stream. 100% efficiency means no liquid in the solids stream.

**Calculation Method** The solids separator does a mass balance and splits the inlet stream phases into two distinct streams.

### 10.16.1. Continuous Cake Filter

In a continuous filter, the feed, filtrate and cake move at steady constant rates. It is evident that the process consists of several steps in series - cake formation, washing, drying and discharging - and that each step involves progressive and continual change in conditions. The pressure drop across the filter during cake formation is, however, held constant.

**Calculation Method** For a continuous cake filter, the equation that relates the filter characteristics with the rate of solids production is [4]

$$\frac{\dot{m}_c}{A_T} = \frac{\left[2c\alpha\Delta P f n / \mu + (nR_m)^2\right]^{0,5} - nR_m}{\alpha}, \quad (10.15)$$

where:

$\dot{m}_c$	rate of solids production, $kg/s$
$A_T$	total filter area, $m^2$
$\Delta P$	total pressure drop, $Pa$
$f$	fraction of filter area available for cake formation
$c$	solids concentration in the solids stream
$\alpha$	specific cake resistance, $m/kg$
$R_m$	filter medium resistance, $m^{-1}$
$n$	drum speed, $s^{-1}$

### **Input and Output Parameters**

- Filter Medium Resistance: filter medium flow resistance;
- Specific Cake Resistance: specific cake flow resistance;
- Cycle Time: filter cycle time.
- Cake Relative Humidity: filter cake moisture in % wet basis;
- Filter Calculation Mode: Design or Simulation. If **Design** is selected, DWSIM will calculate the filter area given the total pressure drop. If **Simulation** is selected, it will do the opposite;
- Total Filter Area: filter area measured perpendicularly to the direction of flow;
- Pressure Drop: total pressure drop across the filter (cake + medium).

## **10.17. Excel Unit Operation**

The Excel (Spreadsheet) Unit Operation is used to enable user defined models of unit operations which are calculated by Excel spreadsheet calculator.

Each instance of that unit operation model is associated with a separate calculation file. This file is preconfigured in a special way to receive data from DWSIM for executing the calculations and hold results to be transferred back. The results are fetched from DWSIM to be fed back into the exit streams of that unit operation. You also may transfer user defined parameters from DWSIM to Excel and read other user defined parameters as results back to DWSIM. Up to four streams may be connected both to inlet and outlet of that unit. The energy stream receives the calculated energy from enthalpy balance.

**10.17.0.1. Calculation Parameters** Calculation parameters are defined inside the Excel definition file as input parameters.

**Editor:** This field displays the Excel file being associated with that unit. Click on file name to open Unit editor. Results Heat added After calculating the Excel user model DWSIM calculates the enthalpy balance of input and output streams. Other results are parameters calculated by the user model as output parameters. Annotation The selected parameter may contain an annotation which is defined inside the Excel file for.

**Search Button:** to search Excel file to be associated with Unit Operation

**Create New Button:** to create new definition file from template. Edit Opens definition file with Excel for editing. After editing you just save the Excel file from within Excel.

**Excel definition file:** the excel definition file has a preconfigured structure which should not be altered. Otherwise DWSIM may not be able to transfer information forth and back in the right way!

**10.17.0.2. Input Tab** DWSIM is writing informations of all connected input streams into the blue area. From line 12 downward all components and their molar flows are listed. The red area contains the parameters which are required for calculation. These parameters are displayed in DWSIM as "Calculation Parameters" inside the property tab of ExcelUO. You may list as many properties as you want here. DWSIM starts to read the list below the heading downwards until it finds an empty cell. Each parameter features a name, a value, a unit and an annotation.

**10.17.0.3. Output Tab** The output tab has the same structure as the input tab. Molare flows of each component of every output stream are to be written into their fields by the user defined calculation procedures. You also have to calculate temperature and pressure of all streams leaving the unit. The enthalpy of each stream is calculated by DWSIM automatically after finishing Excel calculations.

**10.17.0.4. Results** DWSIM is calculating the enthalpy balance of the unit from enthalpy of outlet streams minus enthalpy of inlet streams. The result of this calculation is written to the energy stream. After finishing the calculations in Excel, DWSIM checks the mass balance of that unit. If mass balance is not ok DWSIM will issue an error message.

## 10.18. Flowsheet Unit Operation

The Flowsheet Unit Operation allows you to run a XML simulation file as a block inside another flowsheet. This can be useful if you have a large simulation and want to split it in several, smaller blocks which can be run as independent simulations, making it easier to mantain, make modifications and fix errors in the smaller blocks.

Mass transfer between flowsheets is done in a per-compound basis, that is, any compound that doesn't match in both simulations will have its flow data erased in the inner flowsheet and will be ignored in the outer one. The mass is transferred to and from the inner flowsheet by matching inlet and outlet

Material Streams connected to the Flowsheet UO with streams in the inner flowsheet. Besides mass/mole flow information, temperature, pressure and enthalpy are also written and read to and from the inner flowsheet.

DWSIM uses the Property Package models defined in the inner flowsheet to do the mass and energy balances. Only settings like Parallel CPU and Parallel GPU calculations affect the way that DWSIM does its calculations inside the block, since it will use the parameters defined when the inner flowsheet was last saved.

You can select Parameters/Properties from objects in the inner flowsheet to expose them to the outer flowsheet, allowing usage of these parameters in Optimization and Sensitivity Analysis cases, Script blocks and for displaying in the outer flowsheet as well.

**10.18.0.1. Connections** Ten inlet and ten outlet Material Stream ports are available for connecting with Material Streams from the inner flowsheet.

After connecting streams to the ports, you must open the Control Panel to map the connected streams to streams in the inner flowsheet.

**10.18.0.2. Calculation Parameters** **Simulation file:** selects the XML simulation file to use as the inner flowsheet.

**Control Panel:** opens the Control Panel to initialize the flowsheet, define stream mapping, expose input and output parameters from the inner flowsheet and define the mass transfer mode.

**Initialize on load:** if true, initializes the inner flowsheet during the opening of the main flowsheet.

**Update process data on saving:** when saving the main simulation file, DWSIM will update the process data from the inner flowsheet in the selected XML file. Only object process data calculated by the solver will be updated. Other settings will remain unchanged.

**Redirect calculator messages:** show calculation details from the inner flowsheet in the main flowsheet's log window.

**View flowsheet:** shows the inner flowsheet PFD.

**10.18.0.3. Linked Input Parameters** This section will display the parameters that you've selected in the control panel to use as input parameters in the outer flowsheet. They can be changed anytime and will trigger the calculation of the flowsheet block.

**10.18.0.4. Linked Output Parameters** This section will display the parameters that you've selected in the control panel to use as output parameters in the outer flowsheet. They are read-only and will be updated only after the flowsheet block is calculated successfully.

**10.18.0.5. Results** **Mass balance error:** shows the mass balance error in %. It can be useful to detect orphan streams in the inner flowsheet, that is, streams that work as inlet or outlet streams in the inner flowsheet but aren't connected to any stream in the main flowsheet, as this may lead to large mass balance errors.

**Control Panel:** Use the Control Panel to initialize the flowsheet. Only after initialization you'll be able to make the connection and expose parameters from the inner flowsheet. You can also use the "Initialize/Reload" button to reload the simulation file after you've done changes in the simulation by opening it in another DWSIM window.

**10.18.0.6. Viewing the inner flowsheet** By clicking on the "View Flowsheet" button in the property grid you'll be able to view the inner flowsheet, check object properties and the overall layout. You can't change anything here, so any attempt to do so will result in an error.

## 10.19. Reactors

The reactors in DWSIM are specialized modules that solve a particular system of reactions of the same type in sequence or in parallel. There are five different types of reactors available:

- Continuous Stirred Tank Reactor (CSTR)
- Plug Flow Reactor (PFR)
- Gibbs Minimization Reactor
- Equilibrium Reactor
- Conversion Reactor

To run a simulation of a reactor, the user needs to define the chemical reactions which will take place in the reactor. This is done through the **Reactions Manager**, accessible through the **Simulation Settings** panel (Classic UI) or on the **Basis Panel** (Cross-Platform UI).

Reactions can be of Equilibrium, Conversion, Kinetic or Heterogeneous Catalytic types. One or more reactions can be combined to define a Reaction Set. The reactors then "see" the reactions through the reaction sets.

Equilibrium reactions are defined by an equilibrium constant (K). The source of information for the equilibrium constant can be a direct gibbs energy calculation, an expression defined by the user or a constant value. Equilibrium reactions can be used in Equilibrium and Gibbs reactors.

Conversion reactions are defined by the amount of a base compound which is consumed in the reaction. This amount can be a fixed value or a function of the system temperature. Conversion reactions are supported by the Conversion reactor.

Kinetic reactions are reactions defined by a kinetic expression. These reactions are supported by the PFR and CSTR reactors.

Heterogeneous Catalytic reactions in DWSIM must obey the Langmuir–Hinshelwood mechanism, where compounds react over a solid catalyst surface. In this model, reaction rates are a function of catalyst amount (i.e. mol/kg cat.s). These reactions are supported by the PFR and CSTR reactors.

**10.19.0.1. Input Parameters** All reactors share the same basic interface. Properties that need to be set include Input and Output Streams Energy Streams that represent the heat exchanged with the environment Reaction Set to be used Operation Mode (isothermal or adiabatic) Pressure Drop through the reactor. For the PFR and CSTR, it is also necessary to define: Volume of the reaction medium Catalyst loading, particle diameter and void fraction, for Heterogeneous Catalytic reaction simulations only For the Gibbs Reactor, you must select the calculation mode: Direct Minimization: The equilibrium composition is found in such a way that the final gibbs energy is at its minimum. Reaction Extents: Like in the Equilibrium Reactor, the equilibrium composition is found according to the reaction information in the reaction set.

**10.19.0.2. Output Parameters** The output of the reactor calculations includes conversions of the components involved in the reactions, the temperature change and the heat exchanged in the reactor, among other information.

For the PFR, the concentration profiles of reactants and products is shown along the longitudinal axis of the reactor (assuming that the concentration does not vary radially).

**10.19.0.3. Calculation Method** The **Conversion Reactor** is solved by simple energy and mass balances, calculating the reaction heat by considering the variation in the amount of the base component. The **PFR** and **CSTR** are solved by a numerical method for systems of ordinary differential equations (ODEs). **Equilibrium** and **Gibbs** Reactors are solved by using the procedure described by Michelsen in [5].

## 10.20. Logical Operations

### 10.20.1. Recycle

The Recycle operation is composed by a block in the flowsheet which does convergence verifications in systems where downstream material connects somewhere upstream in the diagram. With this tool it is possible to build complex flowsheets, with many recycles, and solve them in an efficient way by using the acceleration methods present in this logical operation.

There are two acceleration methods available: Wegstein and Dominant Eigenvalue. The Wegstein method must be used when there isn't a significant interaction between convergent variables, in the con-

trary the other method can be used. The successive substitution method is slow, but convergence is guaranteed.

The Wegstein method requires some parameters which can be edited by the user. The dominant eigenvalue does not require any additional parameter. The user can define convergence parameters for temperature, pressure and mass flow in the recycle, that is, the minimum acceptable values for the difference in these values between the inlet and outlet streams, which, rigorously, must be identical. The smaller these values are, the more time is used by the calculator in order to converge the recycle.

As a result, the actual error values are shown, together with the necessary convergence iteration steps.

#### 10.20.2. Energy Recycle

The Energy Recycle logical operation works the same way as the normal Recycle, except that it is aimed at Energy Streams. Here, instead of defining convergence parameters for temperature, pressure and flow rate, you'll define the convergence error for the energy only.

#### 10.20.3. Adjust

The Adjust is a logical operation which changes the value of a variable in a stream in order to attain a specification which can be a user-defined value or the value of other variable, in other stream. The adjust operation is very useful when there is a specification which cannot be accomplished directly, imposing the necessity of doing a trial and error calculation. If this is the case, the Adjust does everything automatically.

The user selects the controlled (specified) variable and the manipulated one. Then he defines the parameters:

- Adjust value (or offset): desired value for the variable or the value to be added or subtracted from the referenced variable.
- Maximum iterations: maximum number of iterations to be executed by the adjust;

In the Adjust Control Panel, the user controls the operation convergence. Two convergence methods are available.

#### 10.20.4. Specification

The Specification Logical Operation is used to make a variable assume a value which is function of other variable, from other object. In opposition to the Adjust Op, the Spec is calculated automatically if the source variable is modified during the flowsheet calculation. In order for the calculator do not make any repeated calculation in the target object, one can define the "Calculate target object?" as *False*.

In the Specification Control panel, the user can define the relation between the variables by writing a function as complex as required. The user can click on the "Evaluate Expression" to check if the expression is valid.

## Part V.

# Dynamic Modeling and Simulation

### 11. Introduction

The key difference between the steady-state models and dynamic process models is the ability to take into account variation over time. However, as can be seen below the difference is not simply the addition of a time dimension; dynamic modeling often brings a whole different approach that results in dynamic models being a much truer-to-life representation of the process in many other respects.

While steady-state analysis is mainly used for process flowsheet design, usually to determine mass and energy balances and approximate equipment sizes, or perhaps stream properties, the ability of dynamic models to model transient behavior opens up a whole new world of application. Typical applications of dynamic models are as follows:

- analysis of transient behavior, including performance during start-up, shutdown, and load change;
- regulatory (i.e., PID) control scheme analysis and design;
- design of optimal operating procedures – for example, to optimize transition between product grades;
- design of batch processes;
- design of inherently dynamic continuous processes – for example, pressure swing adsorption;
- fitting data from nonsteady-state operations – for example, dynamic experiments, which contain much more information than steady-state experiments, or estimation of process parameters from transient plant data;
- safety analysis – for example, the determination of peak pressures on compressor trip;
- inventory accounting and reconciliation of plant data;
- online or offline parameter re-estimation to determine key operating parameters such as fouling or deactivation constants;
- online soft-sensing;
- operator training.

A dynamic simulation requires increased calculation time and is mathematically more complex than a steady-state one. It can be seen as a multiply repeated steady-state simulation (based on a fixed time step) with constantly changing parameters.

The dynamic model shares the same physical property packages as the steady-state model, simulating the behaviour of the chemical system in a similar manner. On the other hand, the dynamic model uses a different set of conservation equations which account for changes occurring over time.

The equations for material, energy, and composition balances include an additional accumulation (volume) term which is differentiated with respect to time. Numerical integration is used to determine the process behaviour at sequential time steps. The smaller the step, the more closely the calculated solution matches the analytic solution. However, this gain in rigour is offset by the additional calculation time required to simulate the same amount of elapsed real time. A reasonable compromise is achieved by using the largest possible step size, while maintaining an acceptable degree of accuracy without becoming unstable.

In DWSIM, dynamic modeling is handled by the **Dynamics Manager**. Dynamic modeling components include **Events** and **Event Sets**, **Cause-and-Effect Matrices**, **Integrators**, **Monitored Variables** and **Schedules**.

## 12. Dynamic Simulation Structure and Configuration

### 12.1. Dynamic Model Setup

A dynamic model in DWSIM can be configured by starting from a solved flowsheet in steady-state mode. Additionally, each Boundary Material Stream must have a Pressure or Flow specification. In dynamic mode, a Valve Unit Operation connected to these streams will determine the Pressure-Flow relationships for the rest of the flowsheet as the remaining Unit Operation blocks get solved at each time step (Figure 72).

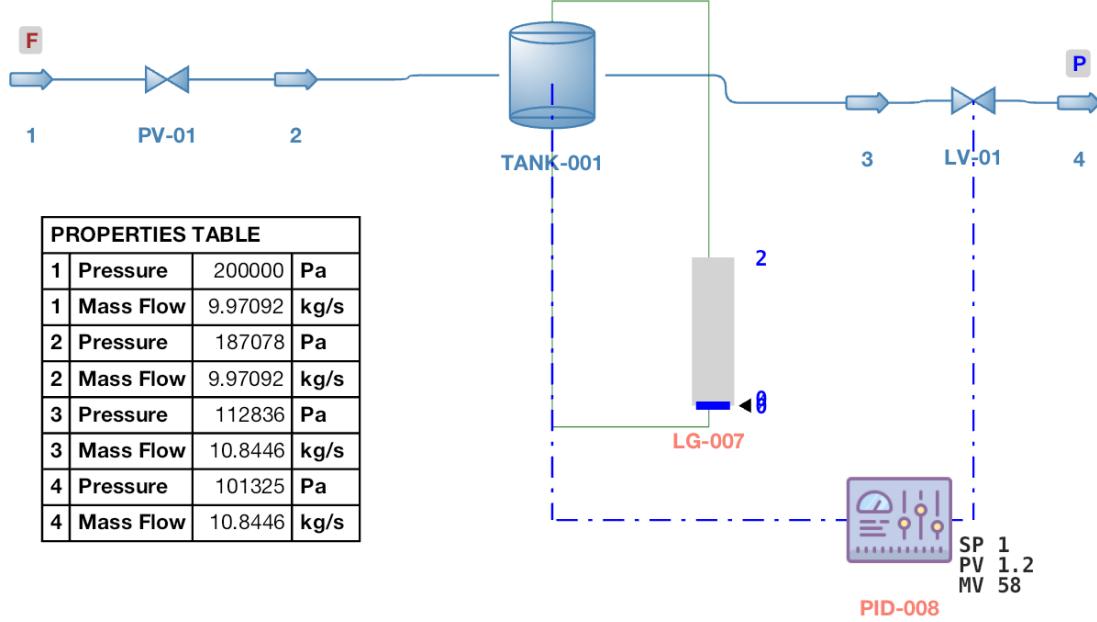


Figure 72: Example of a valid flowsheet for dynamic mode. Both boundary streams are connected to valves. The inlet stream has a Flow specification while the outlet stream is Pressure-specified.

## 12.2. Event Sets

Dynamic Events are property changes that occur in a defined time step. A group of Dynamic Events is called an Event Set.

## 12.3. Cause-and-Effect Matrices

Cause-and-Effect Matrices consist in a set of property changes that occur after an alarm is activated during the dynamic simulation. Each alarm activation triggers a property change defined by the user.

## 12.4. Integrators

Integrators are the components responsible to run the flowsheet sequentially in a dynamic simulation. They can be configured to run during a specified duration in fixed intervals, triggering the mass and balance solvers according to the user preference.

### 12.4.1. Monitored Variables

Each Integrator can have a set of monitored variables, that is, a set of object properties which will have their values stored after each integration step. Monitored variables can have their values exported for later visualization in the Spreadsheet.

## 12.5. Schedules

The Schedule is a set of definitions which, together, can be considered as a *case study* in the context of a dynamic simulation. A Schedule must have the following items defined in order to run:

- Associated Integrator (Required)
- Event Set (Optional)
- Cause-and-Effect Matrix (Optional)
- Initial Flowsheet State (Optional)

A Schedule can be run from the **Integrator Controls** window. After running a Schedule, the results (values of monitored variables at each step) can be copied to a new spreadsheet.

# 13. Dynamic Simulation Flowsheet Blocks

## 13.1. Analog Gauge

Displays an analog gauge on the flowsheet. The gauge is associated to a property of an object on the flowsheet.

## 13.2. Digital Gauge

Displays a digital gauge on the flowsheet. The gauge is associated to a property of an object on the flowsheet.

### 13.3. Level Gauge

Displays a level gauge on the flowsheet. The gauge is associated to a property of an object on the flowsheet.

### 13.4. PID Controller

Introductory text taken from Wikipedia: [https://en.wikipedia.org/wiki/PID\\_controller](https://en.wikipedia.org/wiki/PID_controller)

#### 13.4.1. Introduction

A proportional–integral–derivative controller (PID controller or three-term controller) is a control loop mechanism employing feedback that is widely used in industrial control systems and a variety of other applications requiring continuously modulated control. A PID controller continuously calculates an error value  $e(t)$  as the difference between a desired setpoint (SP) and a measured process variable (PV) and applies a correction based on proportional, integral, and derivative terms (denoted P, I, and D respectively), hence the name.

In practical terms, it automatically applies accurate and responsive correction to a control function. An everyday example is the cruise control on a car, where ascending a hill would lower speed if only constant engine power were applied. The controller's PID algorithm restores the measured speed to the desired speed with minimal delay and overshoot by increasing the power output of the engine.

#### 13.4.2. Fundamental Operation

The distinguishing feature of the PID controller is the ability to use the three control terms of proportional, integral and derivative influence on the controller output to apply accurate and optimal control. The block diagram on (73) shows the principles of how these terms are generated and applied. It shows a PID controller, which continuously calculates an error value  $e(t)$  as the difference between a desired setpoint  $SP = r(t)$  and a measured process variable  $PV = y(t)$ , and applies a correction based on proportional, integral, and derivative terms. The controller attempts to minimize the error over time by adjustment of a control variable  $u(t)$ , such as the opening of a control valve, to a new value determined by a weighted sum of the control terms.

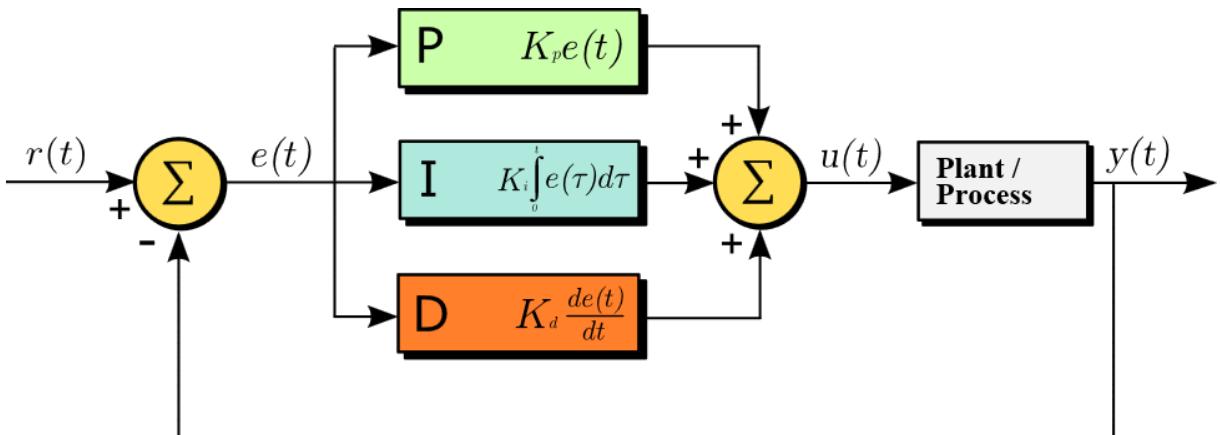


Figure 73: A block diagram of a PID controller in a feedback loop.  $r(t)$  is the desired process value or setpoint (SP), and  $y(t)$  is the measured process value (PV). (source: wikipedia)

In this model:

- Term **P** is proportional to the current value of the SP – PV error  $e(t)$ . For example, if the error is large and positive, the control output will be proportionately large and positive, taking into account the gain factor "K". Using proportional control alone will result in an error between the setpoint and the actual process value, because it requires an error to generate the proportional response. If there is no error, there is no corrective response.
- Term **I** accounts for past values of the SP – PV error and integrates them over time to produce the I term. For example, if there is a residual SP – PV error after the application of proportional control, the integral term seeks to eliminate the residual error by adding a control effect due to the historic cumulative value of the error. When the error is eliminated, the integral term will cease to grow. This will result in the proportional effect diminishing as the error decreases, but this is compensated for by the growing integral effect.
- Term **D** is a best estimate of the future trend of the SP – PV error, based on its current rate of change. It is sometimes called "anticipatory control", as it is effectively seeking to reduce the effect of the SP – PV error by exerting a control influence generated by the rate of error change. The more rapid the change, the greater the controlling or dampening effect.

**Tuning** The balance of these effects is achieved by loop tuning to produce the optimal control function.

The tuning constants are shown as "K" and must be derived for each control application, as they depend on the response characteristics of the complete loop external to the controller. These are dependent on the behaviour of the measuring sensor, the final control element (such as a control valve), any control signal delays and the process itself. Approximate values of constants can usually be initially entered knowing the type of application, but they are normally refined, or tuned, by "bumping" the process in practice by introducing a setpoint change and observing the system response.

**Control action** The mathematical model and practical loop above both use a "direct" control action for all the terms, which means an increasing positive error results in an increasing positive control output for the summed terms to apply correction. However, the output is called "reverse" acting if it is necessary to apply negative corrective action. For instance, if the valve in the flow loop was 100–0% valve opening for 0–100% control output – meaning that the controller action has to be reversed. Some process control schemes and final control elements require this reverse action. An example would be a valve for cooling water, where the fail-safe mode, in the case of loss of signal, would be 100% opening of the valve; therefore 0% controller output needs to cause 100% valve opening.

**Integral wind-up** Integral wind-up, also known as integrator wind-up or reset wind-up, refers to the situation in a PID feedback controller where a large change in setpoint occurs (say a positive change) and the integral term accumulates a significant error during the rise (wind-up), thus overshooting and continuing to increase as this accumulated error is unwound (offset by errors in the other direction). The specific problem is the excess overshooting. This problem can be addressed by preventing the integral term from accumulating above or below pre-determined bounds (wind-up guard).

### 13.4.3. Mathematical form

The overall control function

$$u(t) = K_p e(t) + K_i \int_0^t e(t') dt' + K_d \frac{de(t)}{dt},$$

where  $K_p$ ,  $K_i$  and  $K_d$ , all non-negative, denote the coefficients for the proportional, integral, and derivative terms respectively (sometimes denoted P, I, and D).

### 13.4.4. Selective use of control terms

Although a PID controller has three control terms, some applications need only one or two terms to provide appropriate control. This is achieved by setting the unused parameters to zero and is called a PI, PD, P or I controller in the absence of the other control actions. PI controllers are fairly common in applications where derivative action would be sensitive to measurement noise, but the integral term is often needed for the system to reach its target value.

### 13.4.5. Overview of tuning methods

There are several methods for tuning a PID loop. The most effective methods generally involve the development of some form of process model, then choosing P, I, and D based on the dynamic model parameters. Manual tuning methods can be relatively time-consuming, particularly for systems with long loop times.

The choice of method will depend largely on whether or not the loop can be taken offline for tuning, and on the response time of the system. If the system can be taken offline, the best tuning method often involves subjecting the system to a step change in input, measuring the output as a function of time, and using this response to determine the control parameters.

**13.4.5.1. Manual tuning** If the system must remain online, one tuning method is to first set  $K_i$  and  $K_d$  values to zero. Increase the  $K_p$  until the output of the loop oscillates, then the  $K_p$  should be set to approximately half of that value for a "quarter amplitude decay" type response. Then increase  $K_i$  until any offset is corrected in sufficient time for the process. However, too much  $K_i$  will cause instability. Finally, increase  $K_d$ , if required, until the loop is acceptably quick to reach its reference after a load disturbance. However, too much  $K_d$  will cause excessive response and overshoot. A fast PID loop tuning usually overshoots slightly to reach the setpoint more quickly; however, some systems cannot accept overshoot, in which case an overdamped closed-loop system is required, which will require a  $K_p$  setting significantly less than half that of the  $K_p$  setting that was causing oscillation.

**Effects of *increasing* a parameter independently<sup>[22][23]</sup>**

Parameter	Rise time	Overshoot	Settling time	Steady-state error	Stability
$K_p$	Decrease	Increase	Small change	Decrease	Degrade
$K_i$	Decrease	Increase	Increase	Eliminate	Degrade
$K_d$	Minor change	Decrease	Decrease	No effect in theory	Improve if $K_d$ small

Figure 74: PID manual tuning summary.

**13.4.5.2. Ziegler–Nichols method** Another heuristic tuning method is formally known as the Ziegler–Nichols method, introduced by John G. Ziegler and Nathaniel B. Nichols in the 1940s. As in the method above, the  $K_i$  and  $K_d$  gains are first set to zero. The proportional gain is increased until it reaches the ultimate gain,  $K_u$ , at which the output of the loop starts to oscillate constantly.  $K_u$  and the oscillation period  $T_u$  are used to set the gains as follows:

**Ziegler–Nichols method**

Control Type	$K_p$	$K_i$	$K_d$
<b>P</b>	$0.50K_u$	—	—
<b>PI</b>	$0.45K_u$	$0.54K_u/T_u$	—
<b>PID</b>	$0.60K_u$	$1.2K_u/T_u$	$3K_uT_u/40$

Figure 75: Ziegler-Nichols tuning.

These gains apply to the ideal, parallel form of the PID controller. When applied to the standard PID form, only the integral and derivative time parameters  $T_i$  and  $T_d$  are dependent on the oscillation period  $T_u$ .

#### 13.4.6. DWSIM Implementation

The PID Controller model in DWSIM allows you to configure  $K_p$ ,  $K_i$  and  $K_d$ , set the action type (direct or reverse), set the Integral Wind-Up Guard value and enable/disable the controller. In Control Panel (Real-Time) mode, clicking over the PID Controller icon will display a small panel where you can enable or disable it, set it to AUTO or MANUAL operating mode and change the SP and/or MV values.

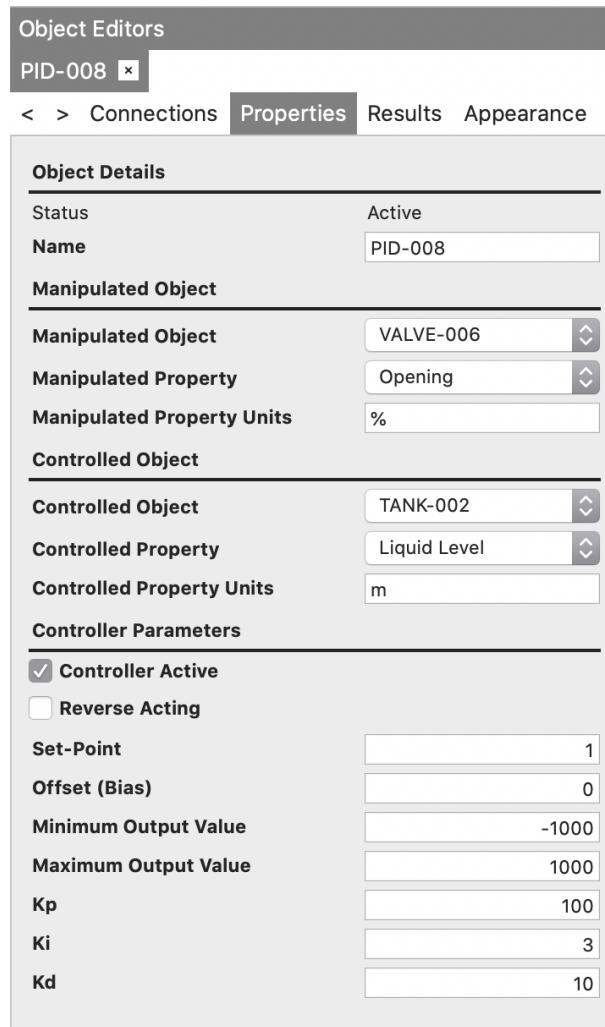


Figure 76: PID Controller Parameters.

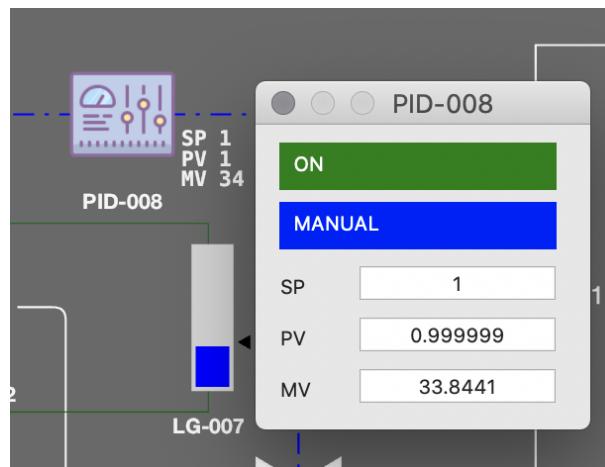


Figure 77: PID Editor in Control Panel (Real-Time) mode.

## 13.5. Input Box

The Input Box can be associated to a property of an object on the flowsheet. Clicking on the box will display an input field from where the user can update the property's value directly.

## 13.6. Switch

The Switch can be associated to a property of an object on the flowsheet. Changing the value of the switch will update the associated property between two predefined values.

# 14. Dynamic Simulation Tools

## 14.1. Control Panel Mode

Control Panel mode makes the Flowsheet read-only and must be used when a Schedule is running in real-time mode. In such scenario, variables can have their values changed through input boxes. You can also change some PID Controller parameters by clicking on their icons on the flowsheet.

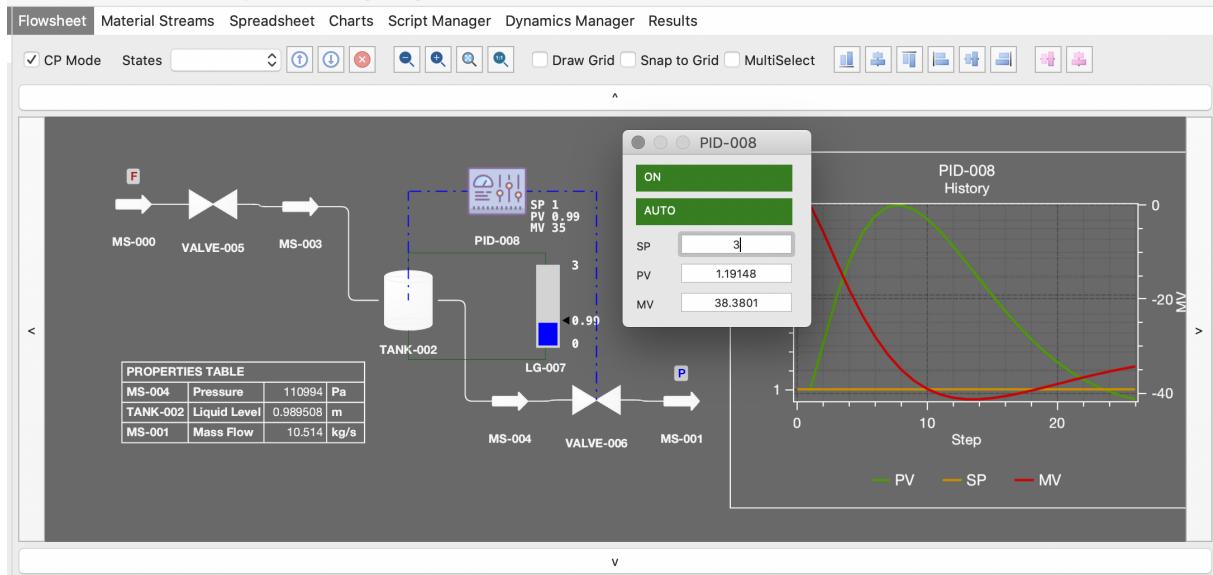


Figure 78: A Flowsheet running in Control Panel (real-time) dynamic mode.

## 14.2. PID Controller Tuning

The PID Controller Tuning tool can be used to tune the parameters ( $K_p$ ,  $K_i$ , and  $K_d$ ) of one or more PID Controllers. It uses a numerical optimizer (Simplex) to obtain the PID parameters which minimize the squared sum of errors ( $SP - PV$ ) for an entire schedule run.

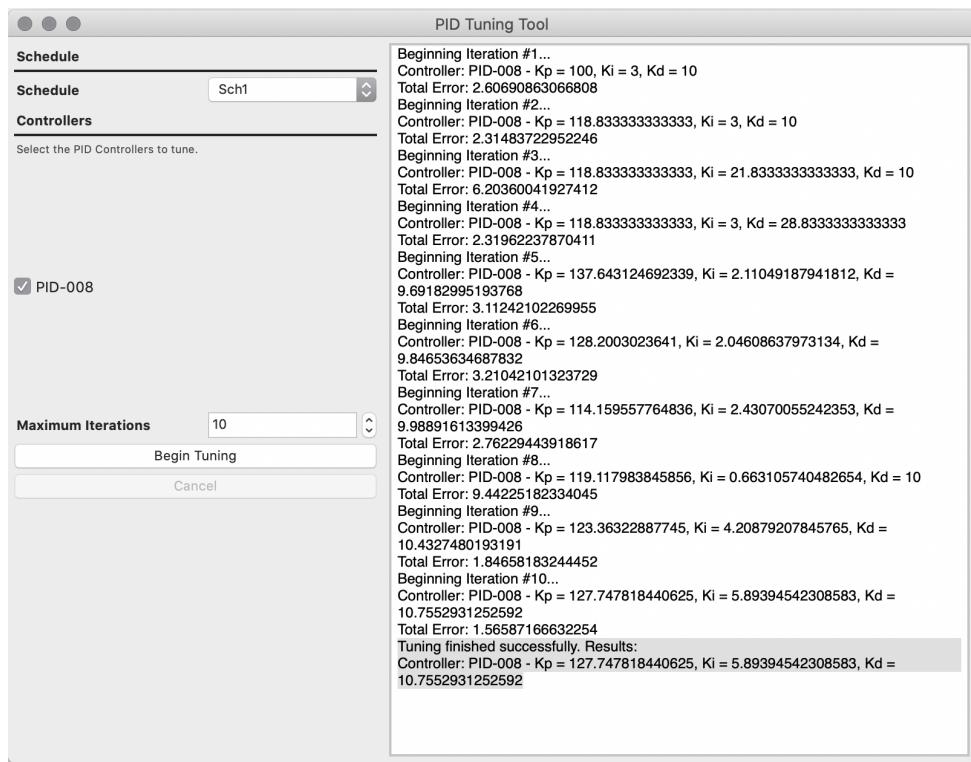


Figure 79: PID Controller Tuning Tool

## 15. Supported Unit Operations

As of v6.0, the following Unit Operations are supported in Dynamic Mode:

- Valve
- Gas-Liquid Separator
- Tank
- Compressor
- Expander
- Pump
- Heater
- Cooler
- Mixer
- Splitter
- Python Script Block
- CAPE-OPEN Block

- Heat Exchanger
- General Reactors (Conversion, Equilibrium and Gibbs)
- CSTR
- PFR
- Specification Logical Block

Unsupported Unit Operations will display an exclamation icon over their flowsheet objects when Dynamic Mode is enabled.



Figure 80: Unsupported Unit Operation in Dynamic Mode.

## Part VI.

# Technical Basis: Methods and Procedures

## 16. Introduction

The thermodynamic calculations are the basis of the simulations in DWSIM. It is important for a process simulator to cover a variety of systems, which can go from simple water handling processes to complex, more elaborated cases, such as simulations of processes in the petroleum/chemical industry.

DWSIM is able to model phase equilibria between solids, vapor and up to two liquid phases where possible. External CAPE-OPEN Property Packages may have different equilibrium capabilities.

The following sections describe the calculation methods used in DWSIM for the physical and chemical description of the elements of a simulation.

## 17. Thermodynamic Properties

### 17.1. Phase Equilibria Calculation

In a mixture which finds itself in a vapor-liquid equilibria state (VLE), the component fugacities are the same in all phases, that is [6]:

$$f_i^L = f_i^V \quad (17.1)$$

The fugacity of a component in a mixture depends on temperature, pressure and composition. in order to relate  $f_i^V$  with temperature, pressure and molar fraction, we define the fugacity coefficient,

$$\phi_i = \frac{f_i^V}{y_i P}, \quad (17.2)$$

which can be calculated from PVT data, commonly obtained from an equation of state. For a mixture of ideal gases,  $\phi_i = 1$ .

The fugacity of the  $i$  component in the liquid phase is related to the composition of that phase by the activity coefficient  $\gamma_i$ , which by itself is related to  $x_i$  and standard-state fugacity  $f_i^0$  by

$$\gamma_i = \frac{f_i^L}{x_i f_i^0}. \quad (17.3)$$

The standard state fugacity  $f_i^0$  is the fugacity of the  $i$ -th component in the system temperature, i.e. mixture, and in an arbitrary pressure and composition. in DWSIM, the standard-state fugacity of each component is considered to be equal to pure liquid  $i$  at the system temperature and pressure.

If an Equation of State is used to calculate equilibria, fugacity of the  $i$ -th component in the liquid phase is calculated by

$$\phi_i = \frac{f_i^L}{x_i P}, \quad (17.4)$$

with the fugacity coefficient  $\phi_i$  calculated by the EOS, just like it is for the same component in the vapor phase.

The fugacity coefficient of the  $i$ -th component either in the liquid or in the vapor phase is obtained from the same Equation of State through the following expressions

$$RT \ln \phi_i^L = \int_{V^L}^{\infty} \left[ \left( \frac{\partial P}{\partial n_i} \right)_{T, V, n_j} - \frac{RT}{V} \right] dV - RT \ln Z^L, \quad (17.5)$$

$$RT \ln \phi_i^V = \int_{V^V}^{\infty} \left[ \left( \frac{\partial P}{\partial n_i} \right)_{T, V, n_j} - \frac{RT}{V} \right] dV - RT \ln Z^V, \quad (17.6)$$

where the compressibility factor  $Z$  is given by

$$Z^L = \frac{PV^L}{RT} \quad (17.7)$$

$$Z^V = \frac{PV^V}{RT} \quad (17.8)$$

### 17.1.1. Fugacity Coefficient calculation models

**Peng-Robinson Equation of State** The Peng-Robinson equation [7] is an cubic Equation of State (characteristic related to the exponent of the molar volume) which relates temperature, pressure and molar volume of a pure component or a mixture of components at equilibrium. The cubic equations are, in fact, the simplest equations capable of representing the behavior of liquid and vapor phases simultaneously. The Peng-Robinson EOS is written in the following form

$$P = \frac{RT}{(V - b)} - \frac{a(T)}{V(V + b) + b(V - b)} \quad (17.9)$$

where

$P$	pressure
$R$	ideal gas universal constant
$v$	molar volume
$b$	parameter related to hard-sphere volume
$a$	parameter related to intermolecular forces

For pure substances, the  $a$  and  $b$  parameters are given by:

$$a(T) = [1 + (0.37464 + 1.54226\omega - 0.26992\omega^2)(1 - T_r^{(1/2)})]^2 0.45724(R^2T_c^2)/P_c \quad (17.10)$$

$$b = 0.07780(RT_c)/P_c \quad (17.11)$$

where

$\omega$	acentric factor
$T_c$	critical temperature
$P_c$	critical pressure
$T_r$	reduced temperature, $T/T_c$

For mixtures, equation 17.9 can be used, replacing  $a$  and  $b$  by mixture-representative values.  $a$  and  $b$  mixture values are normally given by the basic mixing rule,

$$a_m = \sum_i \sum_j x_i x_j \sqrt{(a_i a_j)} (1 - k_{ij}) \quad (17.12)$$

$$b_m = \sum_i x_i b_i \quad (17.13)$$

where

$x_{i,j}$  molar fraction of the  $i$  or  $j$  component in the phase (liquid or vapor)

$a_{i,j}$   $i$  or  $j$  component  $a$  constant

$b_{i,j}$   $i$  or  $j$  component  $b$  constant

$k_{ij}$  binary interaction parameter which characterizes the  $i-j$  pair



**The binary interaction parameters used by DWSIM are loaded from a databank (file) and can be modified in the Property Package configuration window.**

The fugacity coefficient obtained with the Peng-Robinson EOS is given by

$$\ln \frac{f_i}{x_i P} = \frac{b_i}{b_m} (Z - 1) - \ln (Z - B) - \frac{A}{2\sqrt{2B}} \left( \frac{\sum_k x_k a_{ki}}{a_m} - \frac{b_i}{b_m} \right) \ln \left( \frac{Z + 2,414B}{Z - 0,414B} \right), \quad (17.14)$$

where  $Z$  is the phase compressibility factor (liquid or vapor) and can be obtained from the equation 17.9,

$$Z^3 - (1 - B)Z^2 + (A - 3B^2 - 2B)Z - (AB - B^2 - 2B) = 0, \quad (17.15)$$

$$A = \frac{a_m P}{R^2 T^2} \quad (17.16)$$

$$B = \frac{b_m P}{RT} \quad (17.17)$$

$$Z = \frac{PV}{RT} \quad (17.18)$$

**Peng-Robinson (1978) Equation of State** The 1978 version [8] of the PR EOS introduces a modification to the  $a$  term:

If  $\omega_i \leq 0.491$ :

$$c_i = 0.37464 + 1.5422\omega_i - 0.26992\omega_i^2 \quad (17.19)$$

Else:

$$c_i = 0.379642 + 1.48503\omega_i - 0.164423\omega_i^2 + 0.016666\omega_i^3 \quad (17.20)$$

$$a_i = [1 + c_i(1 - T_r^{(1/2)})]^2 0.45724(R^2 T_c^2)/P_c \quad (17.21)$$

**Soave-Redlich-Kwong Equation of State** The Soave-Redlich-Kwong Equation [9] is also a cubic equation of state in volume,

$$P = \frac{RT}{(V - b)} - \frac{a(T)}{V(V + b)}, \quad (17.22)$$

The  $a$  and  $b$  parameters are given by:

$$a(T) = [1 + (0.48 + 1.574\omega - 0.176\omega^2)(1 - T_r^{(1/2)})]^2 0.42747(R^2T_c^2)/P_c \quad (17.23)$$

$$b = 0.08664(RT_c)/P_c \quad (17.24)$$

The equations 17.12 and 17.13 are used to calculate mixture parameters. Fugacity is calculated by

$$\ln \frac{f_i}{x_i P} = \frac{b_i}{b_m} (Z - 1) - \ln(Z - B) - \frac{A}{B} \left( \frac{\sum_k x_k a_{ki}}{a_m} - \frac{b_i}{b_m} \right) \ln \left( \frac{Z + B}{Z} \right) \quad (17.25)$$

The phase compressibility factor  $Z$  is obtained from the equation 17.22,

$$Z^3 - Z^2 + (A - B - B^2)Z - AB = 0, \quad (17.26)$$

$$A = \frac{a_m P}{R^2 T^2} \quad (17.27)$$

$$B = \frac{b_m P}{R T} \quad (17.28)$$

$$Z = \frac{PV}{RT} \quad (17.29)$$

The equations 17.15 and 17.26, in low temperature and pressure conditions, can provide three roots for  $Z$ . In this case, if liquid properties are being calculated, the smallest root is used. If the phase is vapor, the largest root is used. The remaining root has no physical meaning; at high temperatures and pressures (conditions above the pseudocritical point), the equations 17.15 and 17.26 provides only one real root.

**Peng-Robinson with Volume Translation** Volume translation solves the main problem with two-constant EOS's, poor liquid volumetric predictions. A simple correction term is applied to the EOS-calculated molar volume,

$$v = v^{EOS} - c, \quad (17.30)$$

where  $v$  =corrected molar volume,  $v^{EOS}$  =EOS-calculated volume, and  $c$  =component-specific constant. The shift in volume is actually equivalent to adding a third constant to the EOS but is special because equilibrium conditions are unaltered.

It is also shown that multicomponent VLE is unaltered by introducing the volume-shift term  $c$  as a mole-fraction average,

$$v_L = v_L^{EOS} - \sum x_i c_i \quad (17.31)$$

Volume translation can be applied to any two-constant cubic equation, thereby eliminating the volumetric deficiency suffered by all two-constant equations [10].

### Peng-Robinson-Stryjek-Vera

#### PRSV1

A modification to the attraction term in the Peng-Robinson equation of state published by Stryjek and Vera in 1986 (PRSV) significantly improved the model's accuracy by introducing an adjustable pure component parameter and by modifying the polynomial fit of the acentric factor.

The modification is:

$$\kappa = \kappa_0 + \kappa_1 (1 + T_r^{0.5}) (0.7 - T_r) \quad (17.32)$$

$$\kappa_0 = 0.378893 + 1.4897153 \omega - 0.17131848 \omega^2 + 0.0196554 \omega^3 \quad (17.33)$$

where  $\kappa_1$  is an adjustable pure component parameter. Stryjek and Vera published pure component parameters for many compounds of industrial interest in their original journal article.

#### PRSV2

A subsequent modification published in 1986 (PRSV2) [11] further improved the model's accuracy by introducing two additional pure component parameters to the previous attraction term modification.

The modification is:

$$\kappa = \kappa_0 + [\kappa_1 + \kappa_2 (\kappa_3 - T_r) (1 - T_r^{0.5})] (1 + T_r^{0.5}) (0.7 - T_r) \quad (17.34)$$

$$\kappa_0 = 0.378893 + 1.4897153 \omega - 0.17131848 \omega^2 + 0.0196554 \omega^3 \quad (17.35)$$

where  $\kappa_1$ ,  $\kappa_2$ , and  $\kappa_3$  are adjustable pure component parameters.

PRSV2 is particularly advantageous for VLE calculations. While PRSV1 does offer an advantage over the Peng-Robinson model for describing thermodynamic behavior, it is still not accurate enough, in general, for phase equilibrium calculations. The highly non-linear behavior of phase-equilibrium calculation methods tends to amplify what would otherwise be acceptably small errors. It is therefore recommended that PRSV2 be used for equilibrium calculations when applying these models to a design. However, once the equilibrium state has been determined, the phase specific thermodynamic values at equilibrium may be determined by one of several simpler models with a reasonable degree of accuracy.

#### 17.1.2. Chao-Seader and Grayson-Streed models

Chao-Seader ([12]) and Grayson-Streed ([13]) are older, semi-empirical models. The Grayson-Streed correlation is an extension of the Chao-Seader method with special applicability to hydrogen. In DWSIM, only the equilibrium values produced by these correlations are used in the calculations. The Lee-Kesler method is used to determine the enthalpy and entropy of liquid and vapor phases.

**Chao Seader** Use this method for heavy hydrocarbons, where the pressure is less than 10 342 kPa (1 500 psia) and the temperature is between the range -17.78 °C and 260 °C.

**Grayson Streed** Recommended for simulating heavy hydrocarbon systems with a high hydrogen content.

### 17.1.3. Calculation models for the liquid phase activity coefficient

The activity coefficient  $\gamma$  is a factor used in thermodynamics to account for deviations from ideal behaviour in a mixture of chemical substances. In an ideal mixture, the interactions between each pair of chemical species are the same (or more formally, the enthalpy of mixing is zero) and, as a result, properties of the mixtures can be expressed directly in terms of simple concentrations or partial pressures of the substances present. Deviations from ideality are accommodated by modifying the concentration by an activity coefficient. . The activity coefficient is defined as

$$\gamma_i = \left[ \frac{\partial(nG^E/RT)}{\partial n_i} \right]_{P,T,n_j \neq i} \quad (17.36)$$

where  $G^E$  represents the excess Gibbs energy of the liquid solution, which is a measure of how far the solution is from ideal behavior. For an ideal solution,  $\gamma_i = 1$ . Expressions for  $G^E/RT$  provide values for the activity coefficients.

**UNIQUAC and UNIFAC models** The UNIQUAC equation considers  $g \equiv G^E/RT$  formed by two additive parts, one combinatorial term  $g^C$  to take into account the size of the molecules, and one residual term  $g^R$ , which take into account the interactions between molecules:

$$g \equiv g^C + g^R \quad (17.37)$$

The  $g^C$  function contains only pure species parameters, while the  $g^R$  function incorporates two binary parameters for each pair of molecules. For a multicomponent system,

$$g^C = \sum_i x_i \ln \phi_i/x_i + 5 \sum_i q_i x_i \ln \theta_i/\phi_i \quad (17.38)$$

and

$$g^R = - \sum_i q_i x_i \ln \left( \sum_j \theta_j \tau_j i \right) \quad (17.39)$$

where

$$\phi_i \equiv (x_i r_i) / \left( \sum_j x_j r_j \right) \quad (17.40)$$

and

$$\theta_i \equiv (x_i q_i) / \left( \sum_j x_j q_j \right) \quad (17.41)$$

The  $i$  subscript indicates the species, and  $j$  is an index that represents all the species,  $i$  included. All sums are over all the species. Note that  $\tau_{ij} \neq \tau_{ji}$ . When  $i = j$ ,  $\tau_{ii} = \tau_{jj} = 1$ . In these equations,  $r_i$  (a relative molecular volume) and  $q_i$  (a relative molecular surface area) are pure species parameters. The influence of temperature in  $g$  enters by means of the  $\tau_{ij}$  parameters, which are temperature-dependent:

$$\tau_{ij} = \exp(u_{ij} - u_{jj})/RT \quad (17.42)$$

This way, the UNIQUAC parameters are values of  $(u_{ij} - u_{jj})$ .

An expression for  $\gamma_i$  is found through the application of the following relation:

$$\ln \gamma_i = [\partial(nG^E/RT)/\partial n_i]_{(P,T,n_j \neq i)} \quad (17.43)$$

The result is represented by the following equations:

$$\ln \gamma_i = \ln \gamma_i^C + \ln \gamma_i^R \quad (17.44)$$

$$\ln \gamma_i^C = 1 - J_i + \ln J_i - 5q_i(1 - J_i/L_i + \ln J_i/L_i) \quad (17.45)$$

$$\ln \gamma_i^R = q_i(1 - \ln s_i - \sum_j \theta_j \tau_{ij}/s_j) \quad (17.46)$$

where

$$J_i = r_i / (\sum_j r_j x_j) \quad (17.47)$$

$$L = q_i / (\sum_j q_j x_j) \quad (17.48)$$

$$s_i = \sum_l \theta_l \tau_{li} \quad (17.49)$$

Again the  $i$  subscript identify the species,  $j$  and  $l$  are indexes which represent all the species, including i. all sums are over all the species, and  $\tau_{ij} = 1$  for  $i = j$ . The parameters values  $(u_{ij} - u_{jj})$  are found by regression of binary VLE/LLE data.

The UNIFAC method for the estimation of activity coefficients depends on the concept of that a liquid mixture can be considered a solution of its own molecules. These structural units are called subgroups. The greatest advantage of this method is that a relatively small number of subgroups can be combined to form a very large number of molecules.

The activity coefficients do not only depend on the subgroup properties, but also on the interactions between these groups. Similar subgroups are related to a main group, like "CH<sub>2</sub>", "OH", "ACH" etc.; the identification of the main groups are only descriptive. All the subgroups that belongs to the same main group are considered identical with respect to the interaction between groups. Consequently, the parameters which characterize the interactions between the groups are identified by pairs of the main groups.

The UNIFAC method is based on the UNIQUAC equation, where the activity coefficients are given by the equation 17.43. When applied to a solution of groups, the equations 17.45 and 17.46 are written in the form:

$$\ln \gamma_i^C = 1 - J_i + \ln J_i - 5q_i(1 - J_i/L_i + \ln J_i/L_i) \quad (17.50)$$

$$\ln \gamma_i^R = q_i(1 - \sum_k (\theta_k \beta_{ik}/s_k) - e_{ki} \ln \beta_{ik}/s_k) \quad (17.51)$$

The parameters  $J_i$  e  $L_i$  are still given by eqs. 17.61 and ???. Furthermore, the following definitions apply:

$$r_i = \sum_k \nu_k^{(i)} R_k \quad (17.52)$$

$$q_i = \sum_k \nu_k^{(i)} Q_k \quad (17.53)$$

$$e_{ki} = (\nu_k^{(i)} Q_k)/q_i \quad (17.54)$$

$$\beta_{ik} = \sum_m e_{mk} \tau_{mk} \quad (17.55)$$

$$\theta_k = (\sum_i x_i q_i e_{ki}) / (\sum_i x_j q_j) \quad (17.56)$$

$$s_k = \sum_m \theta_m \tau_{mk} \quad (17.57)$$

$$s_i = \sum_l \theta_l \tau_{li} \quad (17.58)$$

$$\tau_{mk} = \exp(-a_{mk})/T \quad (17.59)$$

The  $i$  subscript identify the species, and  $j$  is an index that goes through all the species. The  $k$  subscript identify the subgroups, and  $m$  is an index that goes through all the subgroups. The parameter  $\nu_k^{(i)}$  is the number of the  $k$  subgroup in a molecule of the  $i$  species. The subgroup parameter values  $R_k$  and  $Q_k$  and the interaction parameters  $-a_{mk}$  are obtained in the literature.

**Modified UNIFAC (Dortmund) model** The UNIFAC model, despite being widely used in various applications, has some limitations which are, in some way, inherent to the model. Some of these limitations are:

1. UNIFAC is unable to distinguish between some types of isomers.
2. The  $\gamma - \phi$  approach limits the use of UNIFAC for applications under the pressure range of 10-15 atm.
3. The temperature is limited within the range of approximately 275-425 K.

4. Non-condensable gases and supercritical components are not included.
5. Proximity effects are not taken into account.
6. The parameters of liquid-liquid equilibrium are different from those of vapor-liquid equilibrium.
7. Polymers are not included.
8. Electrolytes are not included.

Some of these limitations can be overcome. The insensitivity of some types of isomers can be eliminated through a careful choice of the groups used to represent the molecules. The fact that the parameters for the liquid-liquid equilibrium are different from those for the vapor-liquid equilibrium seems not to have a theoretical solution at this time. One solution is to use both data from both equilibria to determine the parameters as a modified UNIFAC model. The limitations on the pressure and temperature can be overcome if the UNIFAC model is used with equations of state, which carry with them the dependencies of pressure and temperature.

These limitations of the original UNIFAC model have led several authors to propose changes in both combinatorial and the residual parts. To modify the combinatorial part, the basis is the suggestion given by Kikic et al. (1980) in the sense that the Staverman-Guggenheim correction on the original term of Flory-Huggins is very small and can, in most cases, be neglected. As a result, this correction was empirically removed from the UNIFAC model. Among these modifications, the proposed by Gmehling and coworkers [Weidlich and Gmehling, 1986; Weidlich and Gmehling, 1987; Gmehling et al., 1993], known as the model UNIFAC-Dortmund, is one of the most promising. In this model, the combinatorial part of the original UNIFAC is replaced by:

$$\ln \gamma_i^C = 1 - J_i + \ln J_i - 5q_i(1 - J_i/L_i + \ln J_i/L_i) \quad (17.60)$$

$$J_i = r_i^{3/4} / (\sum_j r_j^{3/4} x_j) \quad (17.61)$$

where the remaining quantities is defined the same way as in the original UNIFAC. Thus, the correction in-Staverman Guggenheim is empirically taken from the template. It is important to note that the in the UNIFAC-Dortmund model, the quantities  $R_k$  and  $Q_k$  are no longer calculated on the volume and surface area of Van der Waals forces, as proposed by Bondi (1968), but are additional adjustable parameters of the model.

The residual part is still given by the solution for groups, just as in the original UNIFAC, but now the parameters of group interaction are considered temperature dependent, according to:

$$\tau_{mk} = \exp(-a_{mk}^{(0)} + a_{mk}^{(1)}T + a_{mk}^{(2)}T^2)/T \quad (17.62)$$

These parameters must be estimated from experimental phase equilibrium data. Gmehling et al. (1993) presented an array of parameters for 45 major groups, adjusted using data from the vapor-liquid equilibrium, excess enthalpies, activity coefficients at infinite dilution and liquid-liquid equilibrium. enthalpy and entropy of liquid and vapor.

**Modified UNIFAC (NIST) model** This model [14] is similar to the Modified UNIFAC (Dortmund), with new modified UNIFAC parameters reported for 89 main groups and 984 group–group interactions using critically evaluated phase equilibrium data including vapor–liquid equilibrium (VLE), liquid–liquid equilibrium (LLE), solid–liquid equilibrium (SLE), excess enthalpy (HE), infinite dilution activity coefficient (AINF) and excess heat capacity (CPE) data. A new algorithmic framework for quality assessment of phase equilibrium data was applied for qualifying the consistency of data and screening out possible erroneous data. Substantial improvement over previous versions of UNIFAC is observed due to inclusion of experimental data from recent publications and proper weighting based on a quality assessment procedure. The systems requiring further verification of phase equilibrium data were identified where insufficient number of experimental data points is available or where existing data are conflicting.

**NRTL model** Wilson (1964) presented a model relating  $g^E$  to the molar fraction, based mainly on molecular considerations, using the concept of local composition. Basically, the concept of local composition states that the composition of the system in the vicinity of a given molecule is not equal to the overall composition of the system, because of intermolecular forces.

Wilson's equation provides a good representation of the Gibbs' excess free energy for a variety of mixtures, and is particularly useful in solutions of polar compounds or with a tendency to association in apolar solvents, where Van Laar's equation or Margules' one are not sufficient. Wilson's equation has the advantage of being easily extended to multicomponent solutions but has two disadvantages: first, the less important, is that the equations are not applicable to systems where the logarithms of activity coefficients, when plotted as a function of  $x$ , show a maximum or a minimum. However, these systems are not common. The second, a little more serious, is that the model of Wilson is not able to predict limited miscibility, that is, it is not useful for LLE calculations.

Renon and Prausnitz [15] developed the *NRTL* equation (*Non-Random, Two-Liquid*) based on the concept of local composition but, unlike Wilson's model, the NRTL model is applicable to systems of partial miscibility. The model equation is:

$$\ln \gamma_i = \frac{\sum_{j=1}^n \tau_{ji} x_j G_{ji}}{\sum_{k=1}^n x_k G_{ki}} + \sum_{j=1}^n \frac{x_j G_{ij}}{\sum_{k=1}^n x_k G_{kj}} \left( \tau_{ij} - \frac{\sum_{m=1}^n \tau_{mj} x_m G_{mj}}{\sum_{k=1}^n x_k G_{kj}} \right), \quad (17.63)$$

$$G_{ij} = \exp(-\tau_{ij} \alpha_{ij}), \quad (17.64)$$

$$\tau_{ij} = a_{ij}/RT, \quad (17.65)$$

where

$\gamma_i$	Activity coefficient of component $i$
$x_i$	Molar fraction of component $i$
$a_{ij}$	Interaction parameter between $i-j$ ( $a_{ij} \neq a_{ji}$ ) (cal/mol)
$T$	Temperature (K)

$\alpha_{ij}$  non-randomness parameter for the  $i-j$  pair ( $\alpha_{ij} = \alpha_{ji}$ )

The significance of  $G_{ij}$  is similar to  $\Lambda_{ij}$  from Wilson's equation, that is, they are characteristic energy parameters of the  $ij$  interaction. The parameter is related to the non-randomness of the mixture, i.e. that the components in the mixture are not randomly distributed but follow a pattern dictated by the local composition. When it is zero, the mixture is completely random, and the equation is reduced to the two-suffix Margules equation.

For ideal or moderately ideal systems, the NRTL model does not offer much advantage over Van Laar and three-suffix Margules, but for strongly non-ideal systems, this equation can provide a good representation of experimental data, although good quality data is necessary to estimate the three required parameters.

#### 17.1.4. Models for Aqueous Electrolyte Systems

**17.1.4.1. Revised LIQUAC model (LIQUAC\*)** In electrolyte systems, different properties, such as mean activity coefficients, osmotic coefficients, boiling point elevations, freezing point depressions and salt solubilities can be calculated using the new electrolyte models like LIQUAC and LIFAC.

Common usage scenarios:

- desalination processes
- crystallization processes
- waste water treatment

In the LIQUAC\* model [16], the activity coefficient is calculated by three different terms:

$$\ln \gamma_i = \ln \gamma_i^{LR} + \ln \gamma_i^{MR} + \ln \gamma_i^{SR} \quad (17.66)$$

These three terms, the long range term (LR), the middle range term (MR) and the short range term (SR), consider the different kinds of interactions in electrolyte solutions. The long range term is taken into account by the Debye–Hückel theory as modified by Fowler and Guggenheim to consider different solvents and solvent mixtures. This term takes into account direct charge effects like attraction and repulsion between ions and the formation of a solvate shell in solution and is calculated differently for ions and solvents. The middle range term was developed from the semiempirical Pitzer model and takes into account the indirect charge effects such as interactions between dipoles–dipoles and dipoles–indirect dipoles. The short range term was developed from the corresponding local composition model and takes into account direct neighborhood effects of the compounds in solution. For the calculation of the short range term the part consists of a combinatorial (C) and a residual (R) part. While the combinatorial part takes into account the entropic interactions, i.e. the size and the form of the molecules the residual part considers the enthalpic interactions.

**Extended UNIQUAC [17]** Sander et al. presented in 1986 an extension of the UNIQUAC model by adding a Debye–Hückel term allowing this Extended UNIQUAC model to be used for electrolyte solutions. The model has since been modified and it has proven itself applicable for calculations of vapor–liquid–liquid–solid equilibria and of thermal properties in aqueous solutions containing electrolytes and

non-electrolytes. The model is shown in its current form here as it is presented by Thomsen (1997). The extended UNIQUAC model consists of three terms: a combinatorial or entropic term, a residual or enthalpic term and an electrostatic term

$$G^{ex} = G_{Combinatorial}^{ex} + G_{Residual}^{ex} + G_{Extended\ Debye-Hückel}^{ex} \quad (17.67)$$

The combinatorial and the residual terms are identical to the terms used in the traditional UNIQUAC equation. The electrostatic term corresponds to the extended Debye-Hückel law. The combinatorial, entropic term is independent of temperature and only depends on the relative sizes of the species:

$$\frac{G_{Combinatorial}^{ex}}{RT} = \sum_i x_i \ln \left( \frac{\phi_i}{x_i} \right) - \frac{z}{2} \sum_i q_i x_i \ln \left( \frac{\phi_i}{\theta_i} \right) \quad (17.68)$$

The two model parameters  $r_i$  and  $q_i$  are the volume and surface area parameters for component  $i$ . In the classical application of the UNIQUAC model, these parameters are calculated from the properties of non electrolyte molecules. In the Extended UNIQUAC application to multi component electrolyte solutions, this approach gave unsatisfactory results. The volume and surface area parameters were instead considered to be adjustable parameters. The values of these two parameters are determined by fitting to experimental data. Especially thermal property data such as heat of dilution and heat capacity data are efficient for determining the value of the surface area parameter  $q$ , because the UNIQUAC contribution to the excess enthalpy and excess heat capacity is proportional to the parameter  $q$ . The residual, enthalpic term is dependent on temperature through the parameter  $\psi_{ji}$ :

$$\frac{G_{Residual}^{ex}}{RT} = - \sum_i \left[ x_i q_i \ln \left( \sum_j \theta_j \psi_{ji} \right) \right] \quad (17.69)$$

the parameter  $\psi_{ji}$  is given by:

$$\psi_{ji} = \exp \left( - \frac{u_{ji} - u_{ii}}{T} \right) \quad (17.70)$$

$u_{ji}$  and  $u_{ii}$  are interaction energy parameters. The interaction energy parameters are considered symmetrical and temperature dependent in this model

$$u_{ji} = u_{ij} = u_{ij}^0 + u_{ij}^T (T - 298.15) \quad (17.71)$$

The values of the interaction energy parameters and are determined by fitting to experimental data.

The combinatorial and the residual terms of the UNIQUAC excess Gibbs energy function are based on the rational, symmetrical activity coefficient convention. The Debye-Hückel electrostatic term however is expressed in terms of the rational, symmetrical convention for water, and the rational, unsymmetrical convention for ions.

The electrostatic contributions to the water activity coefficients and the ionic activity coefficients are obtained by partial molar differentiation of the extended Debye-Hückel law excess Gibbs energy term. The term used for water is

$$\ln \gamma_w^{DH} = \frac{2}{3} M_w A I^{3/2} \sigma \left( b I^{1/2} \right) \quad (17.72)$$

$$\sigma(x) = \frac{3}{x^3} \left\{ 1 + x - \frac{1}{1+x} - 2 \ln(1+x) \right\} \quad (17.73)$$

In this expression,  $b = 1.5 \text{ (kg/mol)}^{1/2}$ . The term used for ions is:

$$\ln(\gamma_i^{*DH}) = -Z_i^2 \frac{A\sqrt{T}}{1+b\sqrt{T}} \quad (17.74)$$

Based on table values of the density of pure water, and the relative permittivity of water,  $\varepsilon_r$ , the Debye-Hückel parameter  $A$  can be approximated in the temperature range  $273.15 \text{ K} < T < 383.15 \text{ K}$  by:

$$A = [1.131 + 1.335E - 3(T - 273.15) + 1.164E - 5(T - 273.15)^2] \quad (17.75)$$

The activity coefficient for water is calculated in the Extended UNIQUAC model by summation of the three terms:

$$\ln \gamma_w = \ln \gamma_w^C + \ln \gamma_w^R + \ln \gamma_w^{DH} \quad (17.76)$$

The activity coefficient for ion  $i$  is obtained as the rational, unsymmetrical activity coefficient according to the definition of rational unsymmetrical activity coefficients by adding the three contributions:

$$\ln \gamma_i^* = \ln \frac{\gamma_i^C}{\gamma_i^{C\infty}} + \ln \frac{\gamma_i^R}{\gamma_i^{R\infty}} + \ln \gamma_i^{*DH} \quad (17.77)$$

The rational, unsymmetrical activity coefficient for ions calculated with the Extended UNIQUAC model can be converted to a molal activity coefficient. This is relevant for comparison with experimental data.

The temperature dependency of the activity coefficients in the Extended UNIQUAC model is built into the model equations as outlined above. The temperature dependency of the equilibrium constants used in the Extended UNIQUAC model is calculated from the temperature dependency of the Gibbs energies of formation of the species Parameters for water and for the following ions can be found in[17]  $\text{H}^+$ ,  $\text{Na}^+$ ,  $\text{K}^+$ ,  $\text{NH}_4^+$ ,  $\text{Cl}^-$ ,  $\text{SO}_4^{2-}$ ,  $\text{HSO}_4^-$ ,  $\text{NO}_3^-$ ,  $\text{OH}^-$ ,  $\text{CO}_3^{2-}$ ,  $\text{HCO}_3^-$ ,  $\text{S}_2\text{O}_8^{2-}$ .

A significant advantage of the Extended UNIQUAC model compared to models like the Bromley model or the Pitzer model is that temperature dependence is built into the model. This enables the model to also describe thermodynamic properties that are temperature derivatives of the excess Gibbs function, such as heat of mixing and heat capacity.

## 17.2. Enthalpy, Entropy and Heat Capacities



*$H^{id}$  values are calculated from the ideal gas heat capacity. For mixtures, a molar average is used.*

*The value calculated by the EOS is for the phase, independently of the number of components present in the mixture.*

**Peng-Robinson, Soave-Redlich-Kwong** For the cubic equations of state, enthalpy, entropy and heat capacities are calculated by the *departure functions*, which relates the phase properties in the conditions

of the mixture with the same mixture property in the ideal gas state. This way, the following departure functions are defined [18],

$$\frac{H - H^{id}}{RT} = X; \frac{S - S^{id}}{R} = Y \quad (17.78)$$

values for  $X$  and  $Y$  are calculated by the PR and SRK EOS, according to the table 2:

Table 2: Enthalpy/Entropy calculation with an EOS

	$\frac{H - H^{id}}{RT}$	$\frac{S - S^{id}}{R}$
PR	$Z - 1 - \frac{1}{2^{1.5} b RT} [a - T \frac{da}{dT}] \times \ln(Z - B) - \ln \frac{P}{P^0} - \frac{A}{2^{1.5} b RT} [\frac{T}{a} \frac{da}{dT}] \times \ln \left[ \frac{V+2.414b}{V+0.414b} \right]$	$\times \ln \left[ \frac{V+2.414b}{V+0.414b} \right]$
SRK	$Z - 1 - \frac{1}{b RT} [a - T \frac{da}{dT}] \times \ln(Z - B) - \ln \frac{P}{P^0} - \frac{A}{B} [\frac{T}{a} \frac{da}{dT}] \times \ln \left[ 1 + \frac{b}{V} \right]$	$\times \ln \left[ 1 + \frac{B}{Z} \right]$



In DWSIM,  $P_o = 1$  atm.

Heat capacities are obtained directly from the EOS, by using the following thermodynamic relations:

$$C_p - C_p^{id} = T \int_{\infty}^V \left( \frac{\partial^2 P}{\partial T^2} \right) dV - \frac{T(\partial P/\partial T)_V^2}{(\partial P/\partial V)_T} - R \quad (17.79)$$

$$C_p - C_v = -T \frac{\left( \frac{\partial P}{\partial T} \right)_V^2}{\left( \frac{\partial P}{\partial V} \right)_T} \quad (17.80)$$

**Lee-Kesler** Enthalpies, entropies and heat capacities are calculated by the Lee-Kesler model [19] through the following equations:

$$\frac{H - H^{id}}{RT_c} = T_r \left( Z - 1 - \frac{b_2 + 2b_3/T_r + 3b_4/T_r^2}{T_r V_r} - \frac{c_2 - 3c_3/T_r^2}{2T_r V_r^2} + \frac{d_2}{5T_r V_r^2} + 3E \right) \quad (17.81)$$

$$\frac{S - S^{id}}{R} + \ln \left( \frac{P}{P_0} \right) = \ln Z - \frac{b_2 + b_3/T_r^2 + 2b_4/T_r^3}{V_r} - \frac{c_1 - 2c_3/T_r^3}{2V_r^2} + \frac{d_1}{5V_r^5} + 2E \quad (17.82)$$

$$\frac{C_v - C_v^{id}}{R} = \frac{2(b_3 + 3b_4/T_r)}{T_r^2 V_r} - \frac{3c_3}{T_r^3 V_r^2} - 6E \quad (17.83)$$

$$\frac{C_p - C_p^{id}}{R} = \frac{C_v - C_v^{id}}{R} - 1 - T_r \frac{\left(\frac{\partial P_r}{\partial T_r}\right)_{V_r}^2}{\left(\frac{\partial P_r}{\partial V_r}\right)_{T_r}} \quad (17.84)$$

$$E = \frac{c_4}{2T_r^3\gamma} \left[ \beta + 1 - \left( \beta + 1 + \frac{\gamma}{V_r^2} \right) \exp \left( -\frac{\gamma}{V_r^2} \right) \right] \quad (17.85)$$



**An iterative method is required to calculate  $V_r$ . The user should always watch the values generated by DWSIM in order to detect any issues in the compressibility factors generated by the Lee-Kesler model.**

$$Z = \frac{P_r V_r}{T_r} = 1 + \frac{B}{V_r} + \frac{C}{V_r^2} + \frac{D}{V_r^5} + \frac{c_4}{T_r^3 V_r^2} \left( \beta + \frac{\gamma}{V_r^2} \right) \exp \left( -\frac{\gamma}{V_r^2} \right) \quad (17.86)$$

$$B = b_1 - b_2/T_r - b_3/T_r^2 - b_4/T_r^3 \quad (17.87)$$

$$C = c_1 - c_2/T_r + c_3/T_r^3 \quad (17.88)$$

$$D = d_1 + d_2/T_r \quad (17.89)$$

Each property must be calculated based in two fluids apart from the main one, one simple and other for reference. For example, for the compressibility factor,

$$Z = Z^{(0)} + \frac{\omega}{\omega^{(r)}} \left( Z^{(r)} - Z^{(0)} \right), \quad (17.90)$$

where the (0) superscript refers to the simple fluid while the (r) superscript refers to the reference fluid. This way, property calculation by the Lee-Kesler model should follow the sequence below (enthalpy calculation example):

1.  $V_r$  and  $Z^{(0)}$  are calculated for the simple fluid at the fluid  $T_r$  and  $P_r$ , using the equation 17.81, and with the constants for the simple fluid, as shown in the table 4,  $(H - H^0)/RT_c$  is calculated. This term is  $[(H - H^0)/RT_c]^{(0)}$ . in this calculation,  $Z$  in the equation 17.81 is  $Z^{(0)}$ .
2. The step 1 is repeated, using the same  $T_r$  and  $P_r$ , but using the constants for the reference fluid as shown in table 4. With these values, the equation 17.81 allows the calculation of  $[(H - H^0)/RT_c]^{(r)}$ . In this step,  $Z$  in the equation 17.81 is  $Z^{(r)}$ .
3. Finally, one determines the residual enthalpy for the fluid of interest by

$$[(H - H^0)/RT_c] = [(H - H^0)/RT_c]^{(0)} + \frac{\omega}{\omega^{(r)}} \left( [(H - H^0)/RT_c]^{(r)} - [(H - H^0)/RT_c]^{(0)} \right), \quad (17.91)$$

where  $\omega^{(r)} = 0,3978$ .

Table 4: Constants for the Lee-Kesler model

Constant	Simple Fluid	Reference Fluid
$b_1$	0.1181193	0.2026579
$b_2$	0.265728	0.331511
$b_3$	0.154790	0.027655
$b_4$	0.030323	0.203488
$c_1$	0.0236744	0.0313385
$c_2$	0.0186984	0.0503618
$c_3$	0.0	0.016901
$c_4$	0.042724	0.041577
$d_1 \times 10^4$	0155488	0.48736
$d_2 \times 10^4$	0.623689	0.0740336
$\beta$	0.65392	1.226
$\gamma$	0.060167	0.03754

### 17.3. Speed of Sound

The speed of sound in a given phase is calculated by the following equations:

$$c = \sqrt{\frac{K}{\rho}}, \quad (17.92)$$

where:

$c$  Speed of sound (m/s)

$K$  Bulk Modulus (Pa)

$\rho$  Phase Density ( $\text{kg/m}^3$ )

### 17.4. Joule-Thomson Coefficient

In thermodynamics, the Joule–Thomson effect (also known as the Joule–Kelvin effect, Kelvin–Joule effect, or Joule–Thomson expansion) describes the temperature change of a real gas or liquid when it is forced through a valve or porous plug while kept insulated so that no heat is exchanged with the environment. This procedure is called a throttling process or Joule–Thomson process. At room temperature, all gases except hydrogen, helium and neon cool upon expansion by the Joule–Thomson process. The rate of change of temperature with respect to pressure in a Joule–Thomson process is the Joule–Thomson coefficient.

The Joule–Thomson coefficient for a given phase is calculated by the following definition:

$$\mu = \left( \frac{\partial T}{\partial P} \right)_H, \quad (17.93)$$

The JT coefficient is calculated rigorously by the PR and SRK equations of state, while the Goldzberg correlation is used for all other models,

$$\mu = \frac{0.0048823 T_{pc} (18/T_{pr}^2 - 1)}{P_{pc} C_p \gamma}, \quad (17.94)$$

for gases, and

$$\mu = -\frac{1}{\rho C_p}, \quad (17.95)$$

for liquids.

## 18. Transport Properties

### 18.1. Density

**Liquid Phase** Liquid phase density is calculated with the Rackett equation for non-EOS models when experimental data is not available [18],

$$V_s = \frac{RT_C}{P_C} Z_{RA}^{[1+(1-T_r)^{2/7}]}, \quad (18.1)$$

where:

$V_s$	Saturated molar volume ( $\text{m}^3/\text{mol}$ )
$T_c$	Critical temperature (K)
$P_c$	Critical pressure (Pa)
$T_r$	Reduced temperature
$Z_{RA}$	Rackett constant of the component (or the mixture)
$R$	Ideal Gas constant (8,314 J/[mol.K])



If  $T \geq T_{cm}$ , the Rackett method does not provide a value for  $V_s$  and, in this case, DWSIM uses the EOS-generated compressibility factor to calculate the density of the liquid phase.

For mixtures, the equation 18.1 becomes

$$V_s = R \left( \sum \frac{x_i T_{ci}}{P_{ci}} \right) Z_{RA}^{[1+(1-T_r)^{2/7}]}, \quad (18.2)$$

with  $T_r = T/T_{cm}$ , and

$$T_{cm} = \sum \sum \phi_i \phi_j T_{ci_j}, \quad (18.3)$$

$$\phi_i = \frac{x_i V_{c_i}}{\sum x_i V_{c_i}}, \quad (18.4)$$

$$T_{c_{ij}} = \left[ \frac{8 (V_{c_i} V_{c_j})^{1/2}}{\left( V_{c_i}^{1/3} + V_{c_j}^{1/3} \right)^3} \right] (T_{c_i} T_{c_j})^{1/2}, \quad (18.5)$$

where:

$x_i$  Molar fraction

$V_{c_i}$  Critical volume ( $\text{m}^3/\text{mol}$ )

If  $Z_{RA}$  isn't available, it is calculated from the component acentric factor,

$$Z_{RA} = 0.2956 - 0.08775\omega, \quad (18.6)$$

If the component (or mixture) isn't saturated, a correction is applied in order to account for the effect of pressure in the volume,

$$V = V_s \left[ 1 - (0.0861488 + 0.0344483\omega) \ln \frac{\beta + P}{\beta + P_{vp}} \right], \quad (18.7)$$

with

$$\begin{aligned} \frac{\beta}{P} = & -1 - 9.070217 (1 - T_r)^{1/3} + 62.45326 (1 - T_r)^{2/3} - 135.1102 (1 - T_r) + \\ & + \exp(4.79594 + 0.250047\omega + 1.14188\omega^2) (1 - T_r)^{4/3}, \end{aligned} \quad (18.8)$$

where:

$V$  Compressed liquid volume ( $\text{m}^3/\text{mol}$ )

$P$  Pressure (Pa)

$P_{vp}$  Vapor pressure / Bubble point pressure (Pa)

Finally, density is calculated from the molar volume by the following relation:

$$\rho = \frac{MM}{1000V}, \quad (18.9)$$

where:

$\rho$  Density ( $\text{kg}/\text{m}^3$ )

$V$  Specific volume of the fluid ( $\text{m}^3/\text{mol}$ )

$MM$  Liquid phase molecular volume ( $\text{kg}/\text{kmol}$ )



**For the Ideal Gas Property Package, the compressibility factor is considered to be equal to 1.**

**Vapor Phase** Vapor phase density is calculated from the compressibility factor generated by the EOS model, according with the following equation:

$$\rho = \frac{MM P}{1000 Z R T}, \quad (18.10)$$

where:

$\rho$  Density (kg/m<sup>3</sup>)

$MM$  Molecular weight of the vapor phase (kg/kmol)

$P$  Pressure (Pa)

$Z$  Vapor phase compressibility factor

$R$  Ideal Gas constant (8,314 J/[mol.K])

$T$  Temperature (K)

For ideal gases, the same equation is used, with  $Z = 1$ .

**Mixture** If there are two phases at system temperature and pressure, the density of the mixture is calculated by the following expression:

$$\rho_m = f_l \rho_l + f_v \rho_v, \quad (18.11)$$

where:

$\rho_{m,l,v}$  Density of the mixture / liquid phase / vapor phase (kg/m<sup>3</sup>)

$f_{l,v}$  Volumetric fraction of the liquid phase / vapor phase (kg/kmol)

## 18.2. Viscosity

**Liquid Phase** When experimental data is not available, liquid phase viscosity is calculated from

$$\eta_L = \exp \left( \sum_i x_i \ln \eta_i \right), \quad (18.12)$$

where  $\eta_i$  is the viscosity of each component in the phase, which depends on the temperature and is calculated from experimental data. Dependence of viscosity with the temperature is described in the equation

$$\eta = \exp(A + B/T + C \ln T + DT^E), \quad (18.13)$$

where  $A, B, C, D$  and  $E$  are experimental coefficients (or generated by DWSIM in the case of pseudocomponents or hypotheticals).

**Vapor Phase** Vapor phase viscosity is calculated in two steps. First, when experimental data is not available, the temperature dependence is given by the *Lucas* equation [18],

$$\eta\xi = [0, .807T_r^{0.618} - 0.357 \exp(-0.449T_r) + 0.34 \exp(-4.058T_r) + 0.018] \quad (18.14)$$

$$\xi = 0,176 \left( \frac{T_c}{MM^3 P_c^4} \right)^{1/6}, \quad (18.15)$$

where

$\eta$  Viscosity ( $\mu P$ )

$T_c, P_c$  Component (or mixture) critical properties

$T_r$  Reduced temperature,  $T/T_c$

$MM$  Molecular weight (kg/kmol)

In the second step, the experimental or calculated viscosity with the Lucas method is corrected to take into account the effect of pressure, by the *Jossi-Stiel-Thodos* method [18],

$$\begin{aligned} \left[ (\eta - \eta_0) \left( \frac{T_c}{MM^3 P_c^4} \right)^{1/6} + 1 \right]^{1/4} &= 1.023 + 0.23364\rho_r + \\ &+ 0.58533\rho_r^2 - 0.40758\rho_r^3 + 0.093324\rho_r^4, \end{aligned} \quad (18.16)$$

where

$\eta, \eta_0$  Corrected viscosity / Lucas method calculated viscosity ( $\mu P$ )

$T_c, P_c$  Component critical properties

$\rho_r$  Reduced density,  $\rho/\rho_c = V/V_c$

$MM$  Molecular weight (kg/kmol)

If the vapor phase contains more than a component, the viscosity is calculated by the same procedure, but with the required properties calculated by a molar average.

### 18.3. Surface Tension

When experimental data is not available, the liquid phase surface tension is calculated by doing a molar average of the individual component tensions, which are calculated with the *Brock-Bird* equation [18],

$$\frac{\sigma}{P_c^{2/3} T_c^{1/3}} = (0.132\alpha_c - 0.279) (1 - T_r)^{11/9} \quad (18.17)$$

$$\alpha_c = 0.9076 \left[ 1 + \frac{T_{br} \ln(P_c/1.01325)}{1 - T_{br}} \right], \quad (18.18)$$

where

$\sigma$	Surface tension (N/m)
$T_c$	Critical temperature (K)
$P_c$	Critical pressure (Pa)
$T_{br}$	Reduced normal boiling point, $T_b/T_c$

### 18.4. Isothermal Compressibility

Isothermal compressibility of a given phase is calculated following the thermodynamic definition:

$$\beta = -\frac{1}{V} \frac{\partial V}{\partial P} \quad (18.19)$$

The above expression is calculated rigorously by the PR and SRK equations of state. For the other models, a numerical derivative approximation is used.

### 18.5. Bulk Modulus

The Bulk Modulus of a phase is defined as the inverse of the isothermal compressibility:

$$K = \frac{1}{\beta} \quad (18.20)$$

## 19. Thermal Properties

### 19.1. Thermal Conductivity

**Liquid Phase** When experimental data is not available, the contribution of each component for the thermal conductivity of the liquid phase is calculated by the *Latini* method [18],

$$\lambda_i = \frac{A(1 - T_r)^{0.38}}{T_r^{1/6}} \quad (19.1)$$

$$A = \frac{A^* T_b^{0.38}}{M M^\beta T_c^\gamma}, \quad (19.2)$$

where  $A^*$ ,  $\alpha$ ,  $\beta$  and  $\gamma$  depend on the nature of the liquid (Saturated Hydrocarbon, Aromatic, Water, etc). The liquid phase thermal conductivity is calculated from the individual values by the *Li* method [18],

$$\lambda_L = \sum \sum \phi_i \phi_j \lambda_{ij} \quad (19.3)$$

$$\lambda_{ij} = 2(\lambda_i^{-1} + \lambda_j^{-1})^{-1} \quad (19.4)$$

$$\phi_i = \frac{x_i V_{c_i}}{\sum x_i V_{c_i}}, \quad (19.5)$$

where

$\lambda_L$  liquid phase thermal conductivity (W/[m.K])

**Vapor Phase** When experimental data is not available, vapor phase thermal conductivity is calculated by the *Ely and Hanley* method [18],

$$\lambda_V = \lambda^* + \frac{1000\eta^*}{MM} 1.32 \left( C_v - \frac{3R}{2} \right), \quad (19.6)$$

where

$\lambda_V$  vapor phase thermal conductivity (W/[m.K])

$C_v$  constant volume heat capacity (J/[mol.K])

$\lambda^*$  and  $\eta^*$  are defined by:

$$\lambda^* = \lambda_0 H \quad (19.7)$$

$$H = \left( \frac{16.04E - 3}{MM/1000} \right)^{1/2} f^{1/2}/h^{2/3} \quad (19.8)$$

$$\lambda_0 = 1944\eta_0 \quad (19.9)$$

$$f = \frac{T_0\theta}{190.4} \quad (19.10)$$

$$h = \frac{V_c}{99.2}\phi \quad (19.11)$$

$$\theta = 1 + (\omega - 0.011)(0.56553 - 0.86276 \ln T^+ - 0.69852/T^+) \quad (19.12)$$

$$\phi = [1 + (\omega - 0.011)(0.38650 - 1.1617 \ln T^+)] 0.288/Z_c \quad (19.13)$$

If  $T_r \leq 2$ ,  $T^+ = T_r$ . If  $T_r > 2$ ,  $T^+ = 2$ .

$$h = \frac{V_c}{99.2}\phi \quad (19.14)$$

$$\eta^* = \eta_0 H \frac{MM/1000}{16.04E - 3} \quad (19.15)$$

$$\eta_0 = 10^{-7} \sum_{n=1}^9 C_n T_0^{(n-4)/3} \quad (19.16)$$

$$T_0 = T/f \quad (19.17)$$

## 20. Aqueous Solution Properties

### 20.1. Mean salt activity coefficient

The mean salt activity coefficient is calculated from the activity coefficients of the ions,

$$\ln(\gamma_{\pm}^{*,m}) = \frac{\nu_+}{\nu} \ln(\gamma_+^{*,m}) + \frac{\nu_-}{\nu} \ln(\gamma_-^{*,m}) \quad (20.1)$$

In this equation  $\nu_+$  and  $\nu_-$  are the stoichiometric coefficients of the cations and anions of the salt, while  $\nu$  stands for the sum of these stoichiometric coefficients. With the mean salt activity coefficient the real behavior of a salt can be calculated and it can, e.g. be used for the calculation of electromotoric forces EMF.

### 20.2. Osmotic coefficient

The osmotic coefficient represents the reality of the solvent in electrolyte systems. It is calculated by the logarithmic ratio of the activity and mole fraction of the solvent:

$$\Phi = -\frac{\ln a_i}{M_s \sum_{ion} m_{ion}} \quad (20.2)$$

### 20.3. Freezing point depression

The Schröder and van Laar equation is used:

$$\frac{\ln a_i}{(1 - (T_{m,i}/T))} = \frac{\Delta_m h_i}{RT_{m,i}} \quad (20.3)$$

On the right hand side of the equation a constant factor is achieved, while on the left hand side the activity depends on temperature and composition. For a given composition the freezing point of the system can be calculated iteratively by varying the system temperature. The best way to do this is by starting at the freezing point of the pure solvent. This equation also allows calculating the freezing point of mixed solvent electrolyte systems.

## 21. Specialized Models / Property Packages

### 21.1. IAPWS-IF97 Steam Tables

Water is used as cooling medium or heat transfer fluid and it plays an important role for air-condition. For conservation or for reaching desired properties, water must be removed from substances (drying). In other cases water must be added (humidification). Also, many chemical reactions take place in hydrous solutions. That's why a good deal of work has been spent on the investigation and measurement of water properties over the years. Thermodynamic, transport and other properties of water are known better than of any other substance. Accurate data are especially needed for the design of equipment in steam power plants (boilers, turbines, condensers). In this field it's also important that all parties involved, e.g., companies bidding for equipment in a new steam power plant, base their calculations on the same property data values because small differences may produce appreciable differences.

A standard for the thermodynamic properties of water over a wide range of temperature and pressure was developed in the 1960's, the 1967 IFC Formulation for Industrial Use (IFC-67). Since 1967 IFC-67 has been used for "official" calculations such as performance guarantee calculations of power cycles.

In 1997, IFC-67 has been replaced by a new formulation, the IAPWS Industrial Formulation 1997 for the Thermodynamic Properties of Water and Steam or **IAPWS-IF97** for short. IAPWS-IF97 was developed in an international research project coordinated by the International Association for the Properties of Water and Steam (IAPWS). The formulation is described in a paper by W. Wagner et al., "The IAPWS Industrial Formulation 1997 for the Thermodynamic Properties of Water and Steam," ASME J. Eng. Gas Turbines and Power, Vol. 122 (2000), pp. 150-182 and several steam table books, among others ASME Steam Tables and Properties of Water and Steam by W. Wagner, Springer 1998.

The IAPWS-IF97 divides the thermodynamic surface into five regions:

- Region 1 for the liquid state from low to high pressures,
- Region 2 for the vapor and ideal gas state,
- Region 3 for the thermodynamic state around the critical point,
- Region 4 for the saturation curve (vapor-liquid equilibrium),
- Region 5 for high temperatures above 1073.15 K (800 °C) and pressures up to 10 MPa (100 bar).

For regions 1, 2, 3 and 5 the authors of IAPWS-IF97 have developed fundamental equations of very high accuracy. Regions 1, 2 and 5 are covered by fundamental equations for the Gibbs free energy  $g(T,p)$ , region 3 by a fundamental equation for the Helmholtz free energy  $f(T,v)$ . All thermodynamic properties can then be calculated from these fundamental equations by using the appropriate thermodynamic relations. For region 4 a saturation-pressure equation has been developed.

In chemical engineering applications mainly regions 1, 2, 4, and to some extent also region 3 are of interest. The range of validity of these regions, the equations for calculating the thermodynamic properties, and references are summarized in Attachment 1. The equations of the high-temperature region 5 should be looked up in the references. For regions 1 and 2 the thermodynamic properties are given as a function of temperature and pressure, for region 3 as a function of temperature and density. For other independent variables an iterative calculation is usually required. So-called backward equations are provided in IAPWS-IF97 which allow direct calculation of properties as a function of some other sets of variables (see references).

Accuracy of the equations and consistency along the region boundaries are more than sufficient for engineering applications.

More information about the IAPWS-IF97 Steam Tables formulation can be found at <http://www.thermo.ruhr-uni-bochum.de/en/prof-w-wagner/software/iapws-if97.html?id=172>.

## 21.2. IAPWS-08 Seawater

The IAPWS-08 Seawater Property Package is based on the **Seawater-Ice-Air (SIA)** library. The Seawater-Ice-Air (SIA) library contains the **TEOS-10** subroutines for evaluating a wide range of thermodynamic properties of pure water (using IAPWS-95), seawater (using IAPWS-08 for the saline part), ice Ih (using IAPWS-06) and for moist air (using Feistel et al. (2010a), IAPWS (2010)).

**TEOS-10** is based on a Gibbs function formulation from which all thermodynamic properties of seawater (density, enthalpy, entropy sound speed, etc.) can be derived in a thermodynamically consistent manner. TEOS-10 was adopted by the Intergovernmental Oceanographic Commission at its 25th Assembly in June 2009 to replace EOS-80 as the official description of seawater and ice properties in marine science.

A significant change compared with past practice is that TEOS-10 uses Absolute Salinity SA (mass fraction of salt in seawater) as opposed to Practical Salinity SP (which is essentially a measure of the conductivity of seawater) to describe the salt content of seawater. Ocean salinities now have units of g/kg.

Absolute Salinity (g/kg) is an SI unit of concentration. The thermodynamic properties of seawater, such as density and enthalpy, are now correctly expressed as functions of Absolute Salinity rather than being functions of the conductivity of seawater. Spatial variations of the composition of seawater mean that Absolute Salinity is not simply proportional to Practical Salinity; TEOS-10 contains procedures to correct for these effects.

More information about the SIA library can be found at <http://www.teos-10.org/software.htm>.

### 21.3. Black-Oil

When fluids flow from a petroleum reservoir to the surface, pressure and temperature decrease. This affects the gas/liquid equilibrium and the properties of the gas and liquid phases. The black-oil model enables estimation of these, from a minimum of input data.

The black-oil model employs 2 pseudo components:

1. Oil which is usually defined as the produced oil, at stock tank conditions.
2. Gas which then is defined as the produced gas at atmospheric standard conditions.

The basic modeling assumption is that the gas may dissolve in the liquid hydrocarbon phase, but no oil will dissolve in the gaseous phase. This implies that the composition of the gaseous phase is assumed the same at all pressure and temperatures.

The black-oil model assumption is reasonable for mixtures of heavy and light components, like many reservoir oils. The assumption gets worse for mixtures containing much of intermediate components (propane, butane), and is directly misleading for mixtures of light and intermediate components typically found in condensate reservoirs.

In DWSIM, a set of models calculates properties for a black oil fluid so it can be used in a process simulation. Black-oil fluids are defined in DWSIM through a minimum set of properties:

- Oil specific gravity (SGo) at standard conditions
- Gas specific gravity (SGg) at standard conditions
- Gas-to-oil ratio (GOR) at standard conditions
- Basic Sediments and Water (%)

Black oil fluids are defined and created through the **Compound Creator** tool. If multiple black-oil fluids are added to a simulation, a single fluid is calculated (based on averaged black-oil properties) and used to calculate stream equilibrium conditions and phase properties.

The Black-Oil Property Package is a simplified package for quick process calculations involving the black-oil fluids described above. All properties required by the unit operations are calculated based on the set of four basic properties (SGo, SGg, GOR and BSW), so the results of the calculations cannot be considered precise in any way. They can exhibit errors of several orders of magnitude when compared to real-world data.

For more accurate petroleum fluid simulations, use the petroleum characterization tools available in DWSIM together with an Equation of State model like Peng-Robinson or Soave-Redlich-Kwong.

## 21.4. CoolProp

CoolProp [20] is a C++ library that implements pure and pseudo-pure fluid equations of state and transport properties for 114 components.

The CoolProp library currently provides thermophysical data for 114 pure and pseudo-pure working fluids. The literature sources for the thermodynamic and transport properties of each fluid are summarized in a table in the Supporting Information available in the above reference.

For the CoolProp Property Package, DWSIM implements simple mixing rules based on mass fraction averages in order to calculate mixture enthalpy, entropy, heat capacities, density (and compressibility factor as a consequence). For equilibrium calculations, DWSIM requires values of fugacity coefficients at system's temperature and pressure. In the CoolProp Property Package, the vapor and liquid phases are considered to be ideal.

More information about CoolProp can be found at <http://www.coolprop.org>.

## 21.5. Sour Water

The Sour Water Property Package is based on the SWEQ model described in the USEPA Report EPA-600/2-80-067: "A new correlation of NH<sub>3</sub>, CO<sub>2</sub>, and H<sub>2</sub>S volatility data from aqueous sour water systems", by Wilson, Grant M., available online at <http://nepis.epa.gov/Exe/ZyPDF.cgi?Dockey=9101B309.PDF>.

In this model, chemical and physical equilibria of NH<sub>3</sub>, CO<sub>2</sub>, and H<sub>2</sub>S in sour water systems including the effects of release by caustic (NaOH) addition are considered. The original model is applicable for temperatures between 20 °C (68 °F) and 140 °C (285 °F), and pressures up to 50 psi. In DWSIM, use of the PR EOS to correct vapour phase non-idealities extends this range but, due to lack of experimental data, exact ranges cannot be specified.

The Sour Water Property Package supports calculation of liquid phase chemical equilibria between the following compounds:

- Water (H<sub>2</sub>O, ChemSep database)
- Ammonia (NH<sub>3</sub>, ChemSep database)
- Hydrogen sulfide (H<sub>2</sub>S, ChemSep database)
- Carbon dioxide (CO<sub>2</sub>, ChemSep database)
- Hydron (H<sup>+</sup>, Electrolytes database)
- Bicarbonate (HCO<sub>3</sub><sup>-</sup>, Electrolytes database)

- Carbonate (CO<sub>3</sub><sup>-2</sup>, Electrolytes database)
- Ammonium (NH<sub>4</sub><sup>+</sup>, Electrolytes database)
- Carbamate (H<sub>2</sub>NCOO<sup>-</sup>, Electrolytes database)
- Bisulfide (S-2, Electrolytes database)
- Sulfide (HS<sup>-</sup>, Electrolytes database)
- Hydroxide (OH<sup>-</sup>, Electrolytes database)
- Sodium Hydroxide (NaOH, Electrolytes database)
- Sodium (Na<sup>+</sup>, Electrolytes database)

The following reactions in the liquid (aqueous) phase are taken into account by the SWEQ model:

- CO<sub>2</sub> ionization, CO<sub>2</sub> + H<sub>2</sub>O  $\leftrightarrow$  H<sup>+</sup> + HCO<sub>3</sub><sup>-</sup>
- Carbonate production, HCO<sub>3</sub><sup>-</sup>  $\leftrightarrow$  CO<sub>3</sub><sup>-2</sup> + H<sup>+</sup>
- Ammonia ionization, H<sup>+</sup> + NH<sub>3</sub>  $\leftrightarrow$  NH<sub>4</sub><sup>+</sup>
- Carbamate production, HCO<sub>3</sub><sup>-</sup> + NH<sub>3</sub>  $\leftrightarrow$  H<sub>2</sub>NCOO<sup>-</sup> + H<sub>2</sub>O
- H<sub>2</sub>S ionization, H<sub>2</sub>S  $\leftrightarrow$  HS<sup>-</sup> + H<sup>+</sup>
- Sulfide production, HS<sup>-</sup>  $\leftrightarrow$  S-2 + H<sup>+</sup>
- Water self-ionization, H<sub>2</sub>O  $\leftrightarrow$  OH<sup>-</sup> + H<sup>+</sup>
- Sodium Hydroxide dissociation, NaOH  $\leftrightarrow$  OH<sup>-</sup> + Na<sup>+</sup>

## 22. Reactions

DWSIM includes support for chemical reactions through the Chemical Reactions Manager. Three types of reactions are available to the user:

**Conversion**, where you must specify the conversion (%) of the limiting reagent as a function of temperature

**Equilibrium**, where you must specify the equilibrium constant (K) as a function of temperature, a constant value or calculated from the Gibbs free energy of reaction ( $\Delta G / R$ ). The orders of reaction of the components are obtained from the stoichiometric coefficients.

**Kinetic**, where you should specify the frequency factor (A) and activation energy (E) for the direct reaction (optionally for the reverse reaction), including the orders of reaction (direct and inverse) of each component.

For each chemical reaction is necessary to specify the stoichiometric coefficients of the compounds and a base compound, which must be a reactant. This base component is used as reference for calculating the heat of reaction.

## 22.1. Conversion Reaction

In the conversion reaction it is assumed that the user has information regarding the conversion of one of the reactants as a function of temperature. By knowing the conversion and the stoichiometric coefficients, the quantities of the other components in the reaction can be calculated.

Considering the following chemical reaction:



where  $a$ ,  $b$  and  $c$  are the stoichiometric coefficients of reactants and product, respectively.  $A$  is the limiting reactant and  $B$  is in excess. The amount of each component at the end of the reaction can then be calculated from the following stoichiometric relationships:

$$N_A = N_{A_0} - N_{A_0}X_A \quad (22.2)$$

$$N_B = N_{B_0} - \frac{b}{a}N_{A_0}X_A \quad (22.3)$$

$$N_C = N_{C_0} + \frac{c}{a}(N_{A_0}X_A) \quad (22.4)$$

where  $N_{A,B,C}$  are the molar amounts of the components at the end of the reaction,  $N_{A_0,B_0,C_0}$  are the molar amount of the components at the start of the reaction and  $X_A$  is the conversion of the base-reactant  $A$ .

## 22.2. Equilibrium Reaction

In the equilibrium reaction, the quantity of each component at the equilibrium is related to equilibrium constant by the following relationship:

$$K = \prod_{j=1}^n (q_j)^{\nu_j}, \quad (22.5)$$

where  $K$  is the equilibrium constant,  $q$  is the basis of components (partial pressure in the vapor phase or activity in the liquid phase)  $\nu$  is the stoichiometric coefficient of component  $j$  and  $n$  is the number of components in the reaction.

The equilibrium constant can be obtained by three different means. One is to consider it a constant, another is considering it as a function of temperature, and finally calculate it automatically from the Gibbs free energy at the temperature of the reaction. The first two methods require user input.

### 22.2.1. Solution method

For each reaction that is occurring in parallel in the system, we can define  $\xi$  as the *reaction extent*, so that the molar amount of each component in the equilibrium is obtained by the following relationship:

$$N_j = N_{j_0} + \sum_i \nu_{ij} \xi_i, \quad (22.6)$$

where  $\xi_i$  is the coordinate of the reaction  $i$  and  $\nu_{ij}$  is the stoichiometric coefficient of the  $j$  component at reaction  $i$ . Defining the molar fraction of the component  $i$  as  $x_i = n_i/n_t$ , where  $n_t$  is the total number of mols, including inert, we have the following expression for each reaction  $i$ :

$$f_i(\xi) = \sum_i \ln(x_i) - \ln(K_i) = 0, \quad (22.7)$$

where the system of equations F can be easily solved by Newton-Raphson's method [5].

### 22.3. Kinetic Reaction

The kinetic reaction is defined by the parameters of the equation of Arrhenius (frequency factor and activation energy) for both the direct order and for the reverse order. Suppose we have the following kinetic reaction:



The reaction rate for the  $A$  component can be defined as

$$r_A = k[A][B] - k'[C][D] \quad (22.9)$$

where

$$k = A \exp(-E/RT) \quad (22.10)$$

$$k' = A' \exp(-E'/RT) \quad (22.11)$$

The kinetic reactions are used in Plug-Flow Reactors (PFRs) and in Continuous-Stirred Tank Reactors (CSTRs). In them, the relationship between molar concentration and the rate of reaction is given by

$$F_A = F_{A_0} + \int_0^V r_A dV, \quad (22.12)$$

where  $F_A$  is the molar flow of the  $A$  component and  $V$  is the reactor volume.

## 23. Property Estimation Methods

### 23.1. Petroleum Fractions

#### 23.1.1. Molecular weight

**Riazi and Al Sahhaf method [21]**

$$MM = \left[ \frac{1}{0.01964} (6.97996 - \ln(1080 - T_b)) \right]^{3/2}, \quad (23.1)$$

where

$MM$  Molecular weight (kg/kmol)

$T_b$  Boiling point at 1 atm (K)

If the specific gravity ( $SG$ ) is available, the molecular weight is calculated by

$$MM = 42.965[\exp(2.097 \times 10^{-4}T_b - 7.78712SG + 2.08476 \times 10^{-3}T_bSG)]T_b^{1.26007}SG^{4.98308} \quad (23.2)$$

**Winn [22]**

$$MM = 0.00005805PEMe^{2.3776}/d15^{0.9371}, \quad (23.3)$$

where

$PEMe$  Mean Boiling Point (K)

$d15$  Specific Gravity @ 60 °F

**Riazi[22]**

$$MM = 42.965 \exp(0.0002097PEMe - 7.78d15 + 0.00208476 \times PEMe \times d15) \times PEMe^{1.26007}d15^{4.98308} \quad (23.4)$$

**Lee-Kesler[22]**

$$t_1 = -12272.6 + 9486.4d15 + (8.3741 - 5.9917d15)PEMe \quad (23.5)$$

$$t_2 = (1 - 0.77084d15 - 0.02058d15^2) \times (0.7465 - 222.466/PEMe) \times 10^7/PEMe \quad (23.6)$$

$$t_3 = (1 - 0.80882d15 - 0.02226d15^2) \times (0.3228 - 17.335/PEMe) \times 10^{12}/PEMe^3 \quad (23.7)$$

$$MM = t_1 + t_2 + t_3 \quad (23.8)$$

**Farah**

$$MM = \exp(6.8117 + 1.3372A - 3.6283B) \quad (23.9)$$

$$MM = \exp(4.0397 + 0.1362A - 0.3406B - 0.9988d15 + 0.0039PEMe), \quad (23.10)$$

where

$A, B$  Walther-ASTM equation parameters for viscosity calculation

**23.1.2. Specific Gravity****Riazi e Al Sahhaf [21]**

$$SG = 1.07 - \exp(3.56073 - 2.93886MM^{0.1}), \quad (23.11)$$

where

$SG$  Specific Gravity

$MM$  Molecular weight (kg/kmol)

**23.1.3. Critical Properties****Lee-Kesler [21]**

$$T_c = 189.8 + 450.6SG + (0.4244 + 0.1174SG)T_b + (0.1441 - 1.0069SG)10^5/T_b \quad (23.12)$$

$$\begin{aligned} \ln P_c &= 5.689 - 0.0566/SG - (0.43639 + 4.1216/SG + 0.21343/SG^2) \times \\ &\quad \times 10^{-3}T_b + (0.47579 + 1.182/SG + 0.15302/SG^2) \times 10^{-6} \times T_b^2 - \\ &\quad -(2.4505 + 9.9099/SG^2) \times 10^{-10} \times T_b^3, \end{aligned} \quad (23.13)$$

where

$T_b$  NBP (K)

$T_c$  Critical temperature (K)

$P_c$  Critical pressure (bar)

**Farah**

$$T_c = 731.968 + 291.952A - 704.998B \quad (23.14)$$

$$T_c = 104.0061 + 38.75A - 41.6097B + 0.7831PEMe \quad (23.15)$$

$$T_c = 196.793 + 90.205A - 221.051B + 309.534d15 + 0.524PEMe \quad (23.16)$$

$$P_c = \exp(20.0056 - 9.8758 \ln(A) + 12.2326 \ln(B)) \quad (23.17)$$

$$P_c = \exp(11.2037 - 0.5484A + 1.9242B + 510.1272/PEMe) \quad (23.18)$$

$$P_c = \exp(28.7605 + 0.7158 \ln(A) - 0.2796 \ln(B) + 2.3129 \ln(d15) - 2.4027 \ln(PEMe) \times d15) \quad (23.19)$$

### Riazi-Daubert[22]

$$\begin{aligned} T_c = & 9.5233 \exp(-0.0009314PEMe - 0.544442d15 + 0.00064791 \times PEMe \times d15) \times \\ & \times PEMe^{0.81067} d15^{0.53691} \end{aligned} \quad (23.20)$$

$$\begin{aligned} P_c = & 31958000000 \exp(-0.008505PEMe - 4.8014d15 + 0.005749 \times PEMe \times d15) \times \\ & \times PEMe^{-0.4844} d15^{4.0846} \end{aligned} \quad (23.21)$$

### Riazi[22]

$$\begin{aligned} T_c = & 35.9413 \exp(-0.00069PEMe - 1.4442d15 + 0.000491 \times PEMe \times d15) \times \\ & \times PEMe^{0.7293} d15^{1.2771} \end{aligned} \quad (23.22)$$

#### 23.1.4. Acentric Factor

##### Lee-Kesler method [21]

$$\omega = \frac{-\ln \frac{P_c}{1.10325} - 5.92714 + 6.09648/T_{br} + 1.28862 \ln T_{br} - 0.169347T_{br}^6}{15.2518 - 15.6875/T_{br} - 13.472 \ln T_{br} + 0.43577T_{br}^6} \quad (23.23)$$

##### Korsten[22]

$$\omega = 0.5899 \times ((PEMV/T_c)^{1.3}) / (1 - (PEMV/T_c)^{1.3}) \times \log(P_c/101325) - 1 \quad (23.24)$$

#### 23.1.5. Vapor Pressure

##### Lee-Kesler method[21]

$$\begin{aligned} \ln P_r^{pv} = & 5.92714 - 6.09648/T_{br} - 1.28862 \ln T_{br} + 0.169347T_{br}^6 + \\ & + \omega(15.2518 - 15.6875/T_{br} - 13.4721 \ln T_{br} + 0.43577T_{br}^6), \end{aligned} \quad (23.25)$$

where

$P_r^{pv}$  Reduced vapor pressure,  $P^{pv}/P_c$

$T_{br}$  Reduced NBP,  $T_b/T_c$

$\omega$  Acentric factor

### 23.1.6. Viscosity

#### Letsou-Stiel [18]

$$\eta = \frac{\xi_0 + \xi_1}{\xi} \quad (23.26)$$

$$\xi_0 = 2.648 - 3.725T_r + 1.309T_r^2 \quad (23.27)$$

$$\xi_1 = 7.425 - 13.39T_r + 5.933T_r^2 \quad (23.28)$$

$$\xi = 176 \left( \frac{T_c}{MM^3 P_c^4} \right)^{1/6} \quad (23.29)$$

where

$\eta$  Viscosity (Pa.s)

$P_c$  Critical pressure (bar)

$T_r$  Reduced temperature,  $T/T_c$

$MM$  Molecular weight (kg/kmol)

#### Abbott[22]

$$t_1 = 4.39371 - 1.94733Kw + 0.12769Kw^2 + 0.00032629API^2 - 0.0118246KwAPI + (0.171617Kw^2 + 10.9943API + 0.0950663API^2 - 0.869218KwAPI) \quad (23.30)$$

$$\log v_{100} = \frac{t_1}{API + 50.3642 - 4.78231Kw}, \quad (23.31)$$

$$t_2 = -0.463634 - 0.166532API + 0.000513447API^2 - 0.00848995APIKw + (0.080325Kw + 1.24899API + 0.19768API^2) \quad (23.32)$$

$$\log v_{210} = \frac{t_2}{API + 26.786 - 2.6296Kw}, \quad (23.33)$$

where

$v_{100}$  Viscosity at 100 °F (cSt)

$v_{210}$  Viscosity at 210 °F (cSt)

$K_w$  Watson characterization factor

$API$  Oil API degree

## 23.2. Hypothetical Components

The majority of properties of the hypothetical components is calculated, when necessary, using the group contribution methods, with the UNIFAC structure of the hypo as the basis of calculation. The table 5 lists the properties and their calculation methods.

Table 5: Hypo calculation methods.

Property	Symbol	Method
Critical temperature	$T_c$	Joback [18]
Critical pressure	$P_c$	Joback [18]
Critical volume	$V_c$	Joback [18]
Normal boiling point	$T_b$	Joback [18]
Vapor pressure	$P^{pv}$	Lee-Kesler (Eq. 23.25)
Acentric factor	$\omega$	Lee-Kesler (Eq. 23.23)
Vaporization enthalpy	$\Delta H_{vap}$	Vetere [18]
Ideal gas heat capacity	$C_p^{gi}$	Harrison-Seaton [23]
Ideal gas enthalpy of formation	$\Delta H_f^{298}$	Marrero-Gani [24]

## 24. Other Properties

### 24.1. True Critical Point

The Gibbs criteria for the true critical point of a mixture of  $n$  components may be expressed of various forms, but the most convenient when using a pressure explicit cubic equation of state is

$$L = \begin{vmatrix} A_{11} & A_{12} & \dots & A_{1n} \\ A_{21} & A_{22} & & \\ \vdots & & & \\ A_{n1} & \dots & \dots & A_{nn} \end{vmatrix} = 0 \quad (24.1)$$

$$M = \begin{vmatrix} A_{11} & A_{12} & \dots & A_{1n} \\ A_{21} & A_{22} & & \\ \vdots & & & \\ A_{n-1,1} & \dots & \dots & A_{n-1,n} \\ \frac{\partial L}{\partial n_1} & \dots & \dots & \frac{\partial L}{\partial n_n} \end{vmatrix} = 0, \quad (24.2)$$

where

$$A_{12} = \left( \frac{\partial^2 A}{\partial n_1 \partial n_2} \right)_{T,V} \quad (24.3)$$

All the  $A$  terms in the equations 24.1 and 24.2 are the second derivatives of the total Helmholtz energy  $A$  with respect to mols and constant  $T$  and  $V$ . The determinants expressed by 24.1 and 24.2 are simultaneously solved for the critical volume and temperature. The critical pressure is then found by using the original EOS.

DWSIM utilizes the method described by Heidemann and Khalil [25] for the true critical point calculation using the *Peng-Robinson* and *Soave-Redlich-Kwong* equations of state.

## 24.2. Natural Gas Hydrates

The models for natural gas hydrates equilibrium calculations are mostly based in statistical thermodynamics to predict in which temperature and pressure conditions there will be formation or dissociation of hydrates. In these conditions,

$$f_w^i = f_w^H, \quad (24.4)$$

that is, the fugacity of water in hydrate is the same as in the water in any other phase present at equilibria.

The difference in the models present in DWSIM is mainly in the way that water fugacity in the hydrate phase is calculated. In the modified van der Waals and Platteeuw model, the isofugacity criteria is used indirectly through chemical potentials, which must also be equal in the equilibria:

$$\mu_w^i = \mu_w^H \quad (24.5)$$

remembering that

$$f_i = x_i P \exp((\mu_i - \mu_i^{gi})/RT). \quad (24.6)$$

### 24.2.1. Modified van der Waals and Platteeuw (Parrish and Prausnitz) method

The classic model for determination of equilibrium pressures and temperatures was developed by van der Waals and Platteeuw. This model was later extended by Parrish and Prausnitz [26] to take into account multiple "guests" in the hydrate structures. The condition of equilibrium used in the *vdwP* model is the equality of the chemical potential of water in the hydrate phase and in the other phases, which can be liquid, solid or both.

**Chemical potential of water in the hydrate phase** In the modified var der Waals method, the chemical potential of water in the hydrate phase is calculated by:

$$\mu_w^H = \mu_w^\beta + RT \sum_m \nu_m \ln(1 - \sum_j \theta_{mj}), \quad (24.7)$$

where  $\mu_w^\beta$  is the chemical potential of water in the empty hydrate lattice (something like an "ideal" chemical potential) and  $\nu_m$  is the number of  $m$  cavities by water molecule in the lattice. The fraction of cavities  $m$ -type cavities occupied by the gaseous component  $l$  is

$$\theta_{ml} = (C_{ml} f_i) / ((1 + \sum_j C_{mj} f_j)), \quad (24.8)$$

where  $C_{mj}$  is the Langmuir constant and  $f_i$  is the fugacity of the gaseous component  $l$ . The Langmuir constant takes into account the interactions between the gas and the molecules of water in the cavities. Using the Lennard-Jones-Devonshire cell theory, van der Waals e Platteeuw showed that the Langmuir constant can be given by

$$C(T) = 4\pi/kT \int_0^\infty \exp[(-w(r))/kT]r^2 dr, \quad (24.9)$$

where T is the absolute temperature, k is the Boltzmann constant and  $w(r)$  is the spherically symmetric potential which is a function of the cell radius, the coordination number and the nature of the gas-water interaction. In this method, the Kihara potential with a spherical core is used,

$$w(r) = 2ze[\sigma^{12}/(R^{11}r)(\delta^{10} + a/R\delta^{11}) - \sigma^6/(R^5r)(\delta^4 + a/R\delta^5)], \quad (24.10)$$

$$\delta^N = [(1 - r/R - a/R)^{-N} - (1 + r/R - a/R)^{-N}]/N, \quad (24.11)$$

where N is equal to 4, 5, 10 or 11; z and R are, respectively, the coordination number and the cavity cell radius.

#### 24.2.1.1. Supported hydrate formers CH<sub>4</sub>, C<sub>2</sub>H<sub>6</sub>, C<sub>3</sub>H<sub>8</sub>, iC<sub>4</sub>H<sub>10</sub>, H<sub>2</sub>S, N<sub>2</sub> and CO<sub>2</sub>.

**Model applicability range** Temperature: 211 to 303 K; Pressure: 1 to 600 atm.

#### 24.2.2. Klauda and Sandler

The model proposed by Klauda and Sandler [27] uses spherically symmetric Kihara potentials determined from viscosity data and the second virial coefficient, in opposition to the traditional models which adjust these parameters to experimental hydrate data. In general, this method predicts hydrate formation data more precisely than the other models.

#### Fugacity of water in the hydrate phase

$$\begin{aligned} f_w^H &= \exp(A_g^\beta \ln T + (B_g^\beta)/T + 2,7789 + D_g^\beta T) \times \\ &\exp V_w^\beta [P - \exp(A_g^\beta \ln T + (B_g^\beta)/T + 2,7789 + D_g^\beta T)]/RT \times \\ &\exp [\sum_m \nu_m \ln (1 - \sum_j (C_{ml} f_l)/(1 + \sum_j C_{mj} f_j))] \end{aligned} \quad (24.12)$$

The A, B and D constants are specific for each hydrate former and represent the vapor pressure of the component in the empty hydrate lattice.  $V_w^\beta$  represents the basic hydrate molar volume (without the presence of guests) and the Langmuir constant (C) is calculated by the following equation:

$$C(T) = 4\pi/kT \int_0^{R-a} \exp[(-w(r))/kT]r^2 dr \quad (24.13)$$

In the Klauda e Sandler method the spherically symmetric Kihara potential is also used,

$$w(r) = 2ze[\sigma^{12}/(R^{11}r)(\delta^{10} + a/R\delta^{11}) - \sigma^6/(R^5r)(\delta^4 + a/R\delta^5)] \quad (24.14)$$

$$\delta^N = [(1 - r/R - a/R)^{-N} - (1 + r/R - a/R)^{-N}]/N \quad (24.15)$$

with a modifications in the potential to include the effects of the second and third cell layers,

$$w(r) = w(r)^{[1]} + w(r)^{[2]} + w(r)^{[3]}. \quad (24.16)$$

**Supported hydrate formers** CH4, C2H6, C3H8, iC4H10, H2S, N2 and CO2.

**Model applicability range** Temperature: 150 to 320 K; Pressure: 1 to 7000 atm

### 24.2.3. Chen and Guo

Chen and Guo [28] developed a model based in a formation mechanism based in two steps, the first being a quasi-chemical reaction to form the "basic hydrate" and the second as being a small gas absorption process in the linking cages of the basic hydrate. The results showed that this model is capable of predict hydrate formation conditions for pure gases and mixtures.

**Fugacity of water in the hydrate phase** In the Chen and Guo model, a different approximation is used for the equilibrium condition. Here the equilibrium is verified by means of an isofugacity criteria of the hydrate formers in the hydrate and vapor phase. The fugacity of the component in the vapor phase is calculated by:

$$f_i^H = f_i^0(1 - \theta_i)^\alpha, \quad (24.17)$$

where  $\alpha$  depends on the structure of the hydrate formed, and

$$f_i^0 = f^0(P)f^0(T)f^0(x_w\gamma_w), \quad (24.18)$$

$$f^0(P) = \exp(\beta P/T), \quad (24.19)$$

$$f^0(T) = A' \exp(B'/(T - C')), \quad (24.20)$$

$$f^0(x_w\gamma_w) = (x_w\gamma_w)^{(-1/\lambda_2)}, \quad (24.21)$$

where  $\beta$  and  $\lambda_2$  depend on the structure of the hydrate formed and  $A'$ ,  $B'$  and  $C'$  depends on the hydrate former.  $x_w$  and  $\gamma_m$  are, respectively, the water molar fraction and activity coefficient in the liquid phase.

In the Chen and Guo model, the Langmuir constants are calculated with an Antoine-type equation with parameters obtained from experimental data, for a limited range of temperature:

$$C(T) = X \exp(Y/(T - Z)) \quad (24.22)$$

**Supported hydrate formers** CH4, C2H6, C3H8, iC4H10, H2S, N2, CO2 and nC4H10.

**Model applicability range** Temperature: 259 to 304 K, Pressure: 1 to 700 atm.

### 24.3. Petroleum Cold Flow Properties

#### 24.3.1. Refraction Index

##### API Procedure 2B5.1

$$I = 0.02266 \exp(0.0003905 \times (1.8MeABP) + 2.468SG - 0.0005704(1.8MeABP) \times SG) \times \\ \times (1.8MeABP)^{0.0572} SG^{-0.72} \quad (24.23)$$

$$r = \left( \frac{1 + 2I}{1 - I} \right)^{1/2} \quad (24.24)$$

where

$r$  Refraction Index

$SG$  Specific Gravity

$MeABP$  Mean Averaged Boiling Point (K)

#### 24.3.2. Flash Point

##### API Procedure 2B7.1

$$PF = \{[0.69 \times ((t_{10ASTM} - 273.15) \times 9/5 + 32) - 118.2] - 32\} \times 5/9 + 273.15 \quad (24.25)$$

where

$PF$  Flash Point (K)

$t_{10ASTM}$  ASTM D93 10% vaporized temperature (K)

#### 24.3.3. Pour Point

##### API Procedure 2B8.1

$$PFL = [753 + 136(1 - \exp(-0.15v_{100})) - 572SG + 0.0512v_{100} + 0.139(1.8MeABP)] / 1.8 \quad (24.26)$$

where

$PFL$  Pour Point (K)

$v_{100}$  Viscosity @ 100 °F (cSt)

#### 24.3.4. Freezing Point

##### API Procedure 2B11.1

$$PC = -2390.42 + 1826SG + 122.49K - 0.135 \times 1.8 \times MeABP \quad (24.27)$$

where

$PC$  Freezing Point (K)

$K$  API characterization factor (API)

#### 24.3.5. Cloud Point

##### API Procedure 2B12.1

$$PN = \left[ 10^{(-7.41+5.49 \log(1.8MeABP)-0.712 \times (1.8MeABP)^{0.315}-0.133SG)} \right] / 1.8 \quad (24.28)$$

where

$PN$  Cloud Point (K)

#### 24.3.6. Cetane Index

##### API Procedure 2B13.1

$$\begin{aligned} IC &= 415.26 - 7.673API + 0.186 \times (1.8MeABP - 458.67) + 3.503API \times \\ &\quad \times \log(1.8MeABP - 458.67) - 193.816 \times \log(1.8MeABP - 458.67) \end{aligned} \quad (24.29)$$

where

$IC$  Cetane Index

$API$  API degree of the oil

## 24.4. Chao-Seader Parameters

The Chao-Seader parameters needed by the CS/GS models are the Modified Acentric Factor, Solubility Parameter and Liquid Molar Volume. When absent, the Modified Acentric Factor is taken as the normal acentric factor, either read from the databases or calculated by using the methods described before in this document. The Solubility Parameter is given by

$$\delta = \left( \frac{\Delta H_v - RT}{V_L} \right)^{1/2} \quad (24.30)$$

where

$\Delta H_v$  Molar Heat of Vaporization

$V_L$  Liquid Molar Volume at 20 °C

# Part VII.

## How-To Tutorials

### 25. Methane Steam Reforming

#### 25.1. Introduction

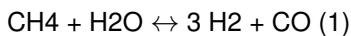
This tutorial is based on the document entitled "Simulation of a Methane Steam Reforming Reactor", which can be found [here](#).

It deals with the task of developing a numerical model to predict the conversion and hydrogen yield within a steam reforming reactor.

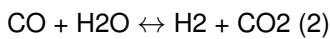
#### 25.2. Background

Natural gas has been proposed as a source of hydrogen for fuel cell vehicle applications because of the existing infrastructure. In a process known as steam reforming, natural gas and steam are reacted into mostly carbon monoxide and hydrogen with some carbon dioxide also produced. There can also be excess water in the reformatte stream.

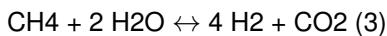
The steam reforming reaction is given as:



In the steam reformer, the water gas shift reaction also takes place as:



Adding together the steam reforming and water gas shift reactions gives the overall reaction:



The equilibrium constants can be expressed in terms of partial pressures (in atm) and temperature in degrees Kelvin. The subscript on the following equilibrium constants refers to the equation number given above:

$$K_1 = \frac{P_{\text{H}_2}^3 P_{\text{CO}}}{P_{\text{CH}_4} P_{\text{H}_2\text{O}}} = \exp(30.42 - 27106/T)$$

$$K_2 = \frac{P_{\text{H}_2} P_{\text{CO}_2}}{P_{\text{CO}} P_{\text{H}_2\text{O}}} = \exp(-3.798 + 4160/T)$$

$$K_3 = \frac{P_{\text{H}_2}^4 P_{\text{CO}_2}}{P_{\text{CH}_4} P_{\text{H}_2\text{O}}^2} = \exp(34.218 - 31266/T)$$

In the reactor, methane ( $\text{CH}_4$ ) and water ( $\text{H}_2\text{O}$ ) are fed as reactants and carbon dioxide ( $\text{CO}_2$ ), carbon monoxide ( $\text{CO}$ ), and hydrogen ( $\text{H}_2$ ) are produced over a nickel catalyst on an alumina support.

In laboratory experiments, a nonreacting inert gas such as helium ( $\text{He}$ ) may also be present. In the most general form, the governing conservation equations for each of these species is given below, where  $\dot{m}_i$  denotes the molar flow rate of species  $i$  in mol/h,  $W$  denotes the catalyst weight in g, and  $R_i$  denotes the reaction rate of equation  $i$  in units of mol/(g-h):

$$\frac{dF_{CH_4}}{dW} = -(R_1 + R_3) \text{ with } F_{CH_4}(W=0) = F_{CH_4}^0$$

$$\frac{dF_{H_2O}}{dW} = -(R_1 + R_2 + 2R_3) \text{ with } F_{H_2O}(W=0) = F_{H_2O}^0$$

$$\frac{dF_{H_2}}{dW} = (3R_1 + R_2 + 4R_3) \text{ with } F_{H_2}(W=0) = F_{H_2}^0$$

$$\frac{dF_{CO}}{dW} = (R_1 - R_2) \text{ with } F_{CO}(W=0) = F_{CO}^0$$

$$\frac{dF_{CO_2}}{dW} = (R_2 + R_3) \text{ with } F_{CO_2}(W=0) = F_{CO_2}^0$$

$$\frac{dF_{He}}{dW} = 0 \text{ with } F_{He}(W=0) = F_{He}^0$$

The reaction rates are given by:

$$R_1 = \frac{\frac{k_1}{P_{H_2}^{2.5}} \left[ P_{CH_4} P_{H_2O} - \frac{P_{H_2}^3 P_{CO}}{K_1} \right]}{DEN^2}$$

$$R_2 = \frac{\frac{k_2}{P_{H_2}} \left[ P_{CO} P_{H_2O} - \frac{P_{H_2} P_{CO_2}}{K_2} \right]}{DEN^2}$$

$$R_3 = \frac{\frac{k_3}{P_{H_2}^{3.5}} \left[ P_{CH_4} P_{H_2O}^2 - \frac{P_{H_2}^4 P_{CO_2}}{K_3} \right]}{DEN^2}$$

$$DEN = 1 + K_{CH_4} P_{CH_4} + K_{CO} P_{CO} + K_{H_2} P_{H_2} + \frac{K_{H_2O} P_{H_2O}}{P_{H_2}}$$

Furthermore, the coefficients in the equations above are given by the Arrhenius relationships as:

$$k_1 = 4.22 \times 10^{15} \exp(-240100 / RT)$$

$$k_2 = 1.96 \times 10^6 \exp(-67130 / RT)$$

$$k_3 = 1.02 \times 10^{15} \exp(-243900 / RT)$$

$$K_{CH4} = 6.65 \times 10^{-4} \exp(38280 / RT)$$

$$K_{H2O} = 1.77 \times 10^5 \exp(-88680 / RT)$$

$$K_{H2} = 6.12 \times 10^{-9} \exp(82900 / RT)$$

$$K_{CO} = 8.23 \times 10^{-5} \exp(70650 / RT)$$

Note that in the above expressions, R = 8.314 J/(mol-K) is the gas constant.

### 25.3. Problem Statement

Consider a feed of 10000 mol/h CH<sub>4</sub>, 10000 mol/h H<sub>2</sub>O, and 100 mol/h H<sub>2</sub> to a steam reforming reactor that operates at 1000 K and a 1 atm feed pressure. Determine the overall methane conversion as a function of catalyst weight up to 382 g.

The overall methane conversion as found on the original reference is equal to **76%**. We'll try to obtain the same result in DWSIM.

### 25.4. DWSIM Model (Classic UI)

1. Create a New Steady State Simulation. Close the Simulation Wizard.



*Remember to Save your simulation at the end of each step.*

2. Go to **Edit > Simulation Settings > Compounds**, and select Methane, Hydrogen, Water, Carbon Monoxide and Carbon Dioxide to add these compounds to the simulation.

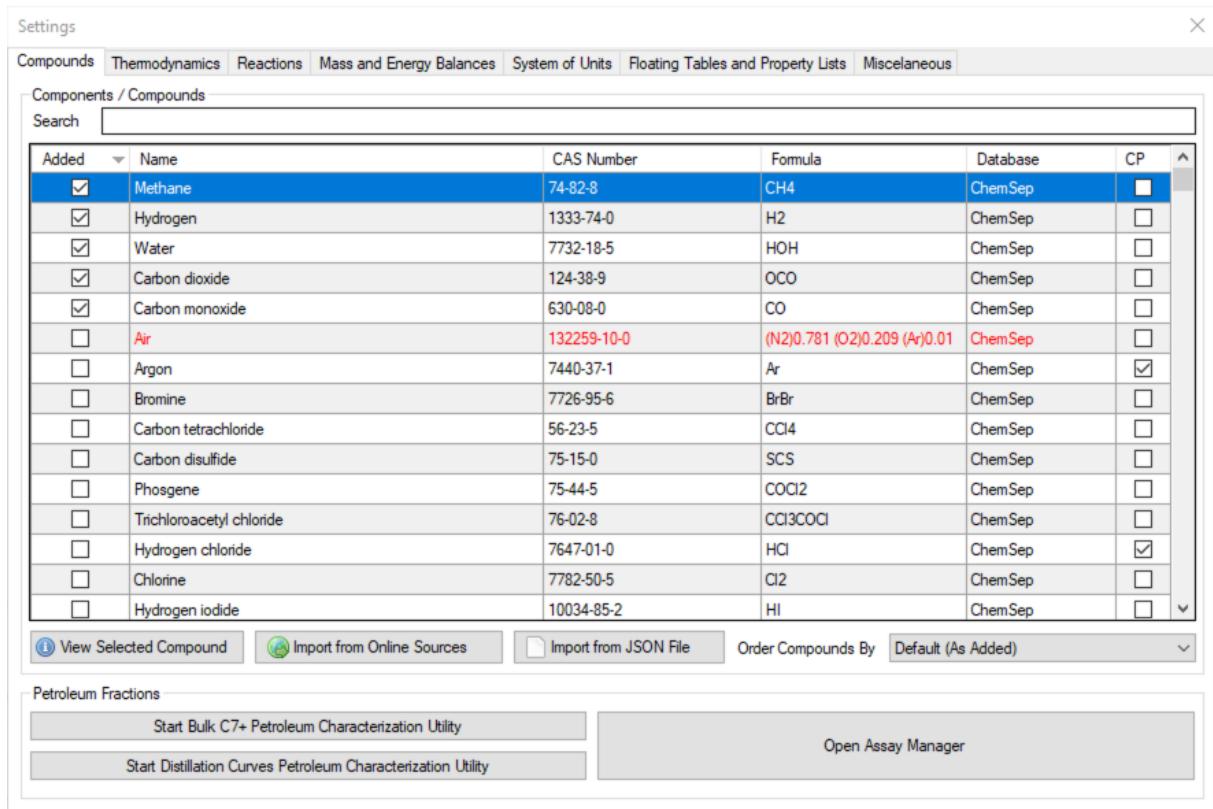


Figure 81: Compound Selection

3. Go to **Thermodynamics** tab, select **Peng-Robinson (PR)** on the Available Property Packages section and click **Add**.

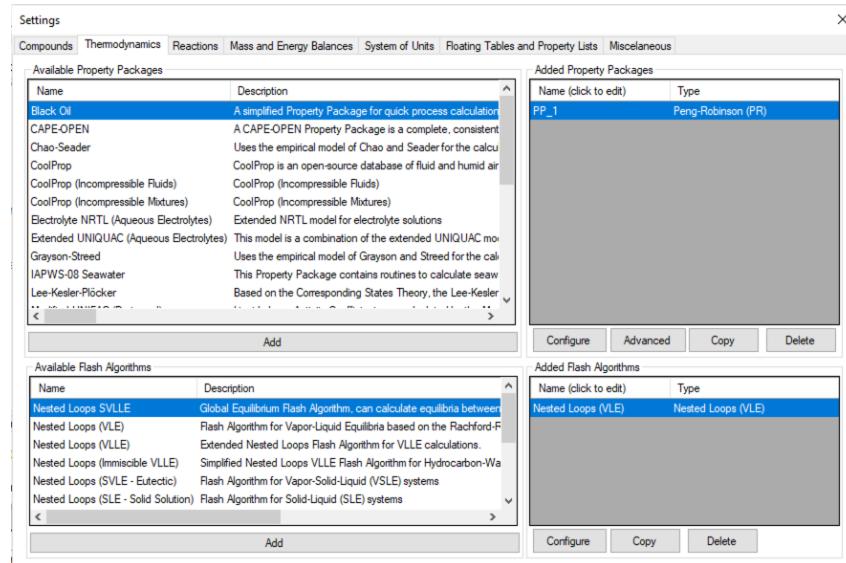


Figure 82: Property Package Selection

4. Go to the **System of Units** tab and create a new System of Units, with the following setup:

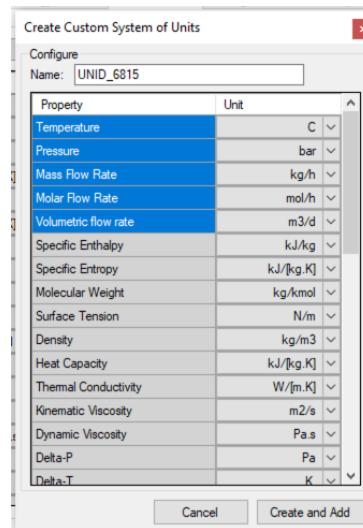


Figure 83: New System of Units

5. After creating this Units Set, select it on the System of Units combobox.
6. Go to **Reactions** and create three Heterogeneous Catalytic reactions, with the following configuration. The denominator expression is the same for all three reactions:  $(1+1.77E+5*\exp(-88680/8.314/T)*R1/P1 + 6.12E-9*\exp(82900/8.314/T)*P1 + 8.23E-5*\exp(70650/8.314/T)*R2)^2$

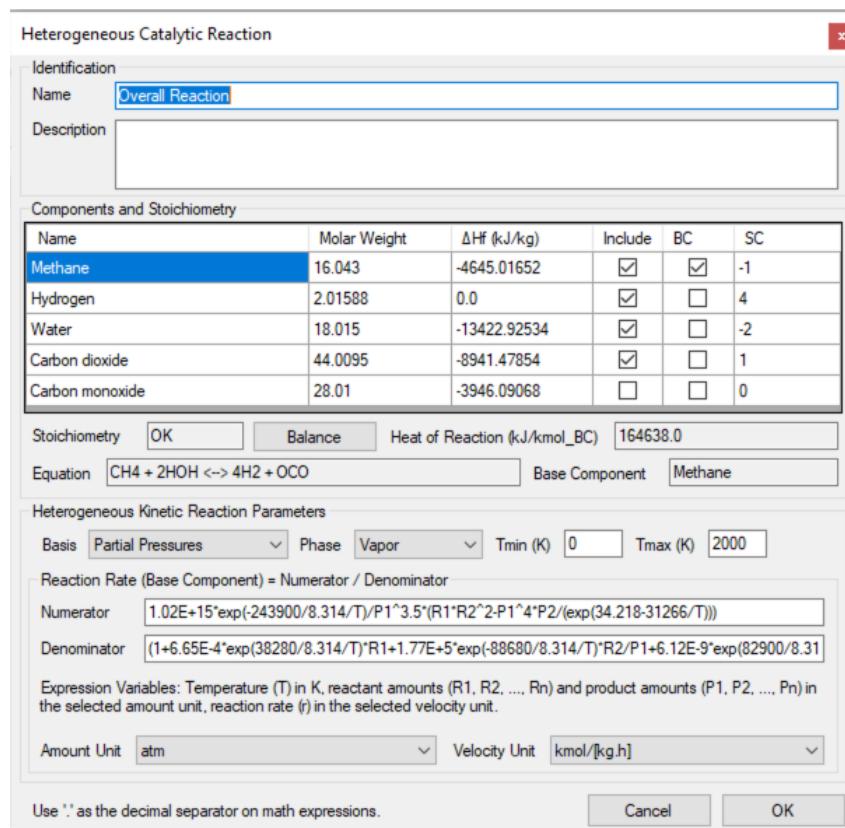


Figure 84: Overall Reaction setup

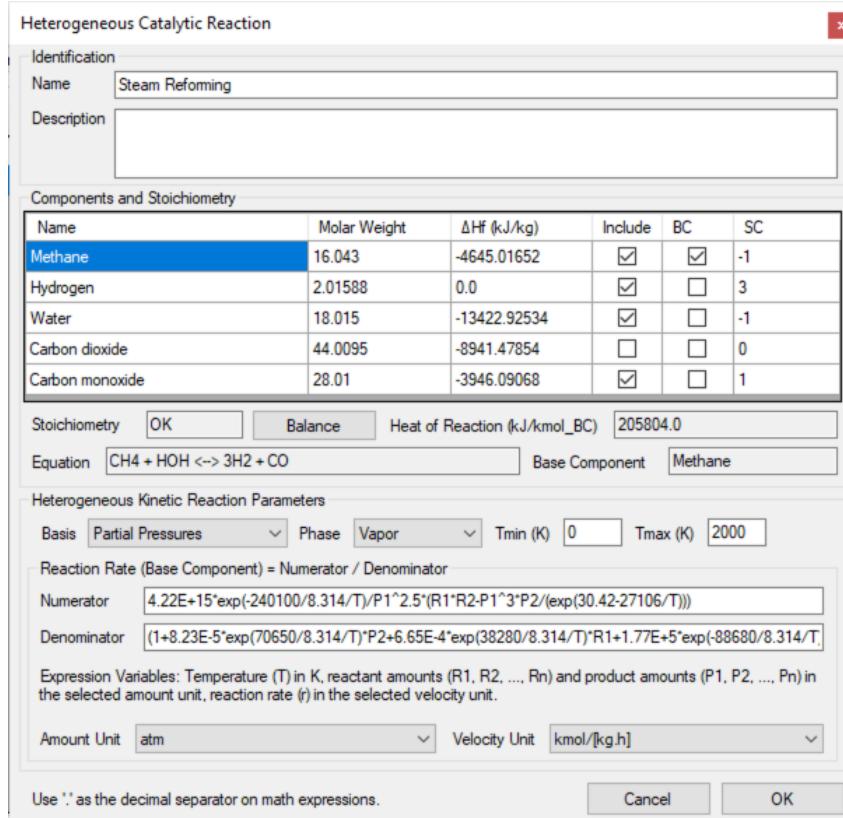


Figure 85: Steam Reforming Reaction setup

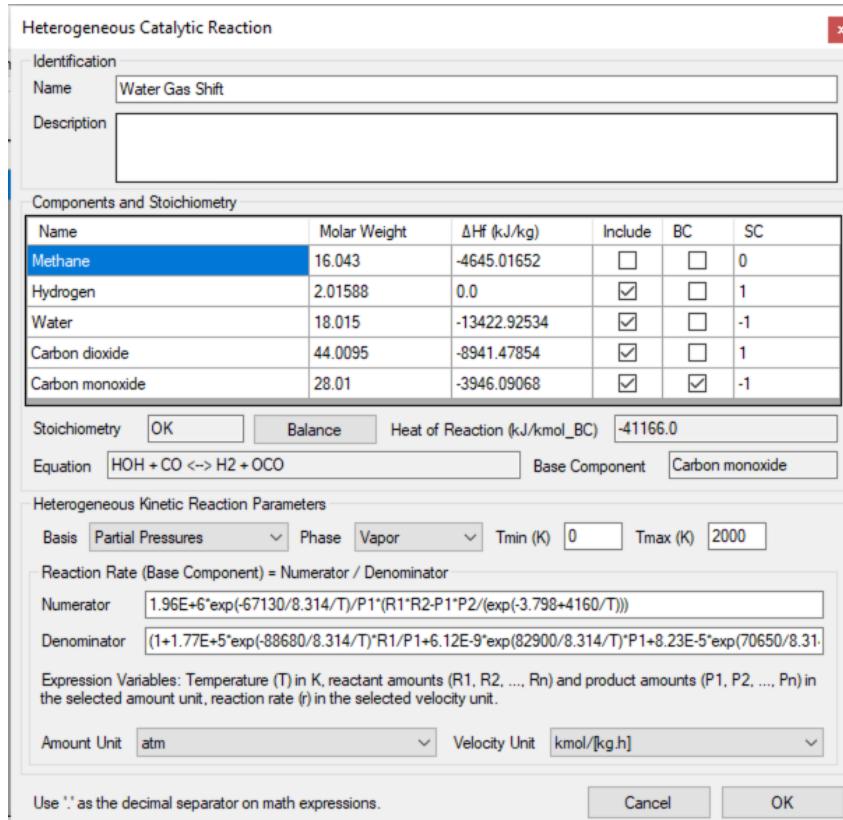


Figure 86: Water Gas Shift Reaction setup

7. Close the Settings panel, and drag two material streams, one energy stream and one PFR to the Flowsheet PFD. Connect the streams to the PFR as shown on the following figure:

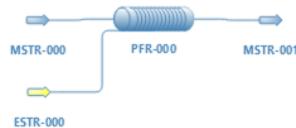


Figure 87: PFD setup

8. Configure the inlet stream (MSTR-000) as follows:

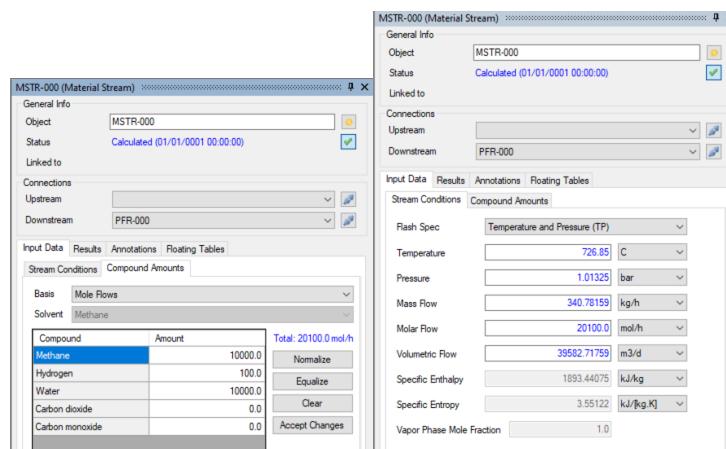


Figure 88: Inlet Stream setup

9. Configure the PFR as follows:

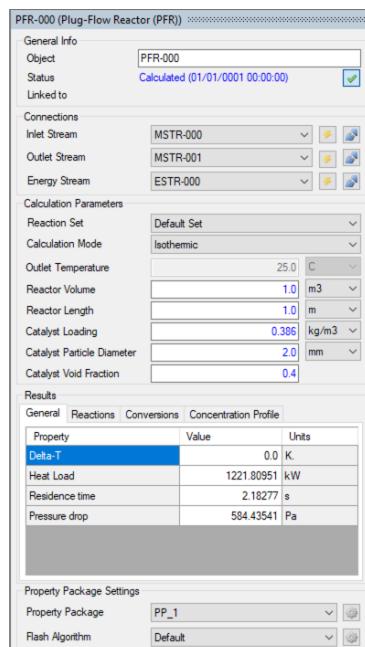


Figure 89: PFR setup

10. **Note:** when you create new reactions, they are automatically added to the **Default Reaction Set**. When you add new reactors to the flowsheet, they are automatically configured to use all supported and active reactions on the Default Reaction Set. You can create, edit and remove Reaction Sets at any time, and associate the individual reactors with different Reaction Sets too.
11. Run the simulation (press F5 or click on the Solve Flowsheet button on the toolbar). Wait for the calculation to finish.
12. Once finished, you should get the following results (methane conversion ~ **76.4%**):

Results	
General	Reactions
Compound	Conversion (%)
Methane	76.40568
Hydrogen	23702.63119
Water	84.21496

Figure 90: Final Methane conversion

13. Create a new Sensitivity Analysis case to study the influence of the temperature on Methane conversion from 700 to 1000 C. Go to **Tools > Sensitivity Analysis** and click on **Create New**.
14. Setup the Independent and Dependent Variables as follows:

Independent Variables			
Object	MSTR-000	Property	Temperature
Lower Limit	700	Number of Points	5
Upper Limit	1000	Unit	C
		Current Value	726.85
Dependent Variables			
Object	PFR-000	Property	Unit
	1	Methane: Conversion	%

Figure 91: Sensitivity Analysis case setup

15. Go to Results and run the Case. Wait for the calculations to finish.
16. Once finished, click on **Send Data to New Worksheet**.

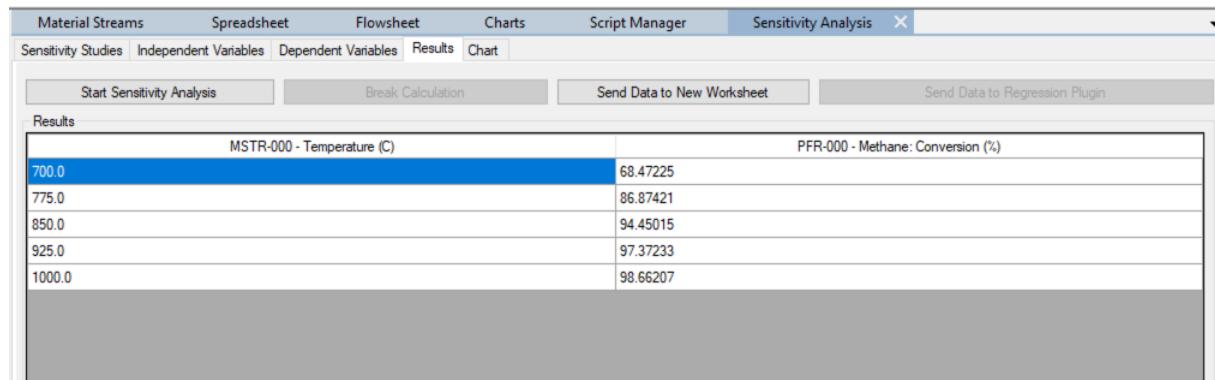


Figure 92: Analysis results

17. With the data range selected on the flowsheet, right-click on it and select **Create 2D XY Chart from Selection**.

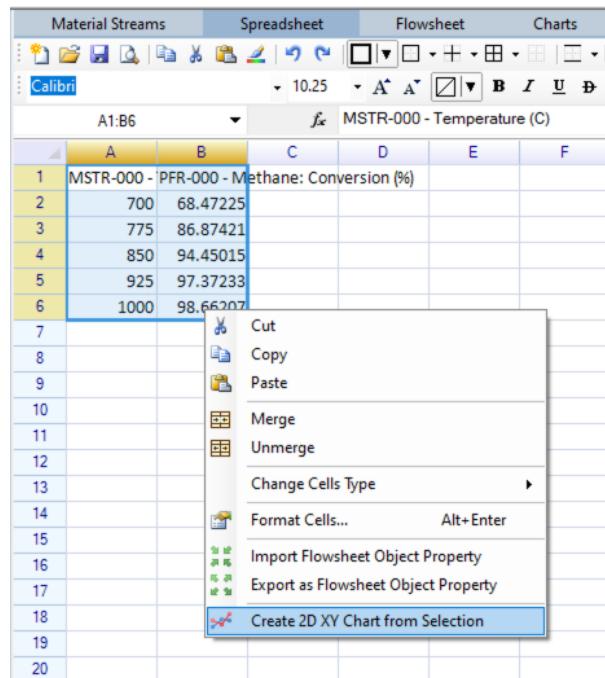


Figure 93: Create Chart from Spreadsheet Range

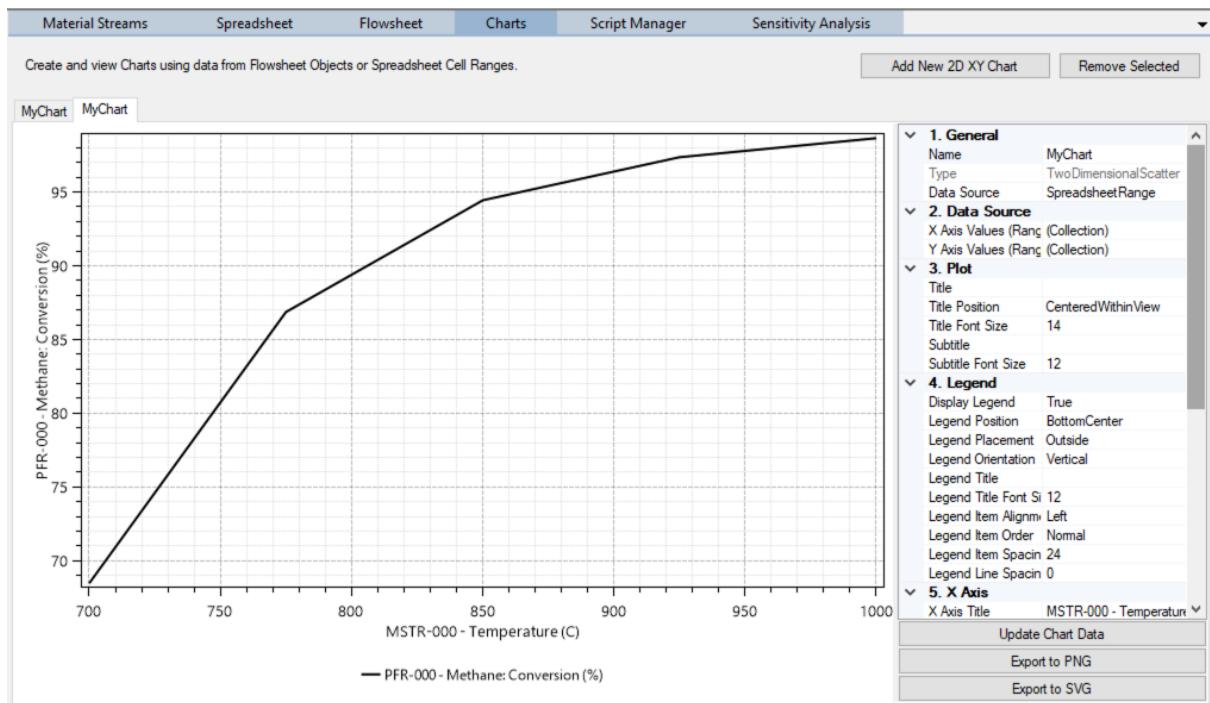


Figure 94: Created Chart

18. You can also view the concentration profile of the PFR using the Charts utility. Click on **Add New 2D XY Chart**, select the PFR as the data source of the chart, select *Concentration Profile* as the **Chart Type** and click on **Update Chart Data**:

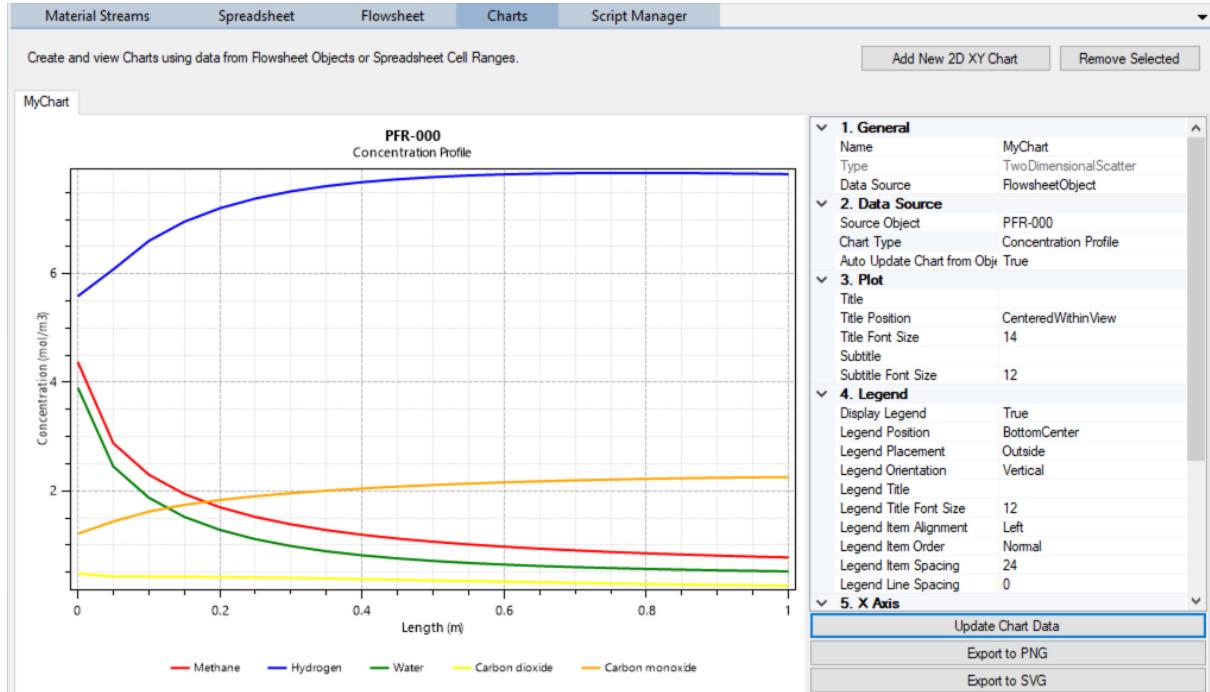
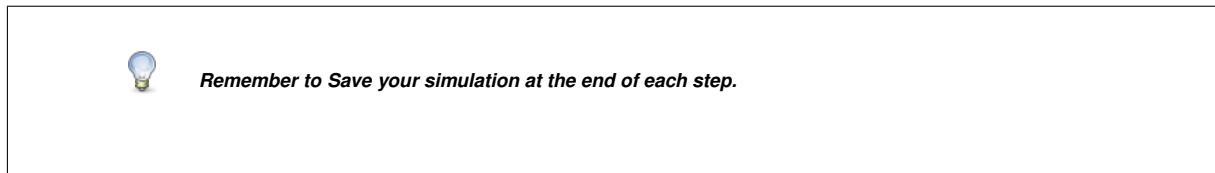


Figure 95: PFR concentration profile

## 25.5. DWSIM Model (Cross-Platform UI)

1. Create a New Simulation. Close the Simulation Wizard.



2. Go to **Setup > Compounds**, and select Methane, Hydrogen, Water, Carbon Monoxide and Carbon Dioxide to add these compounds to the simulation.

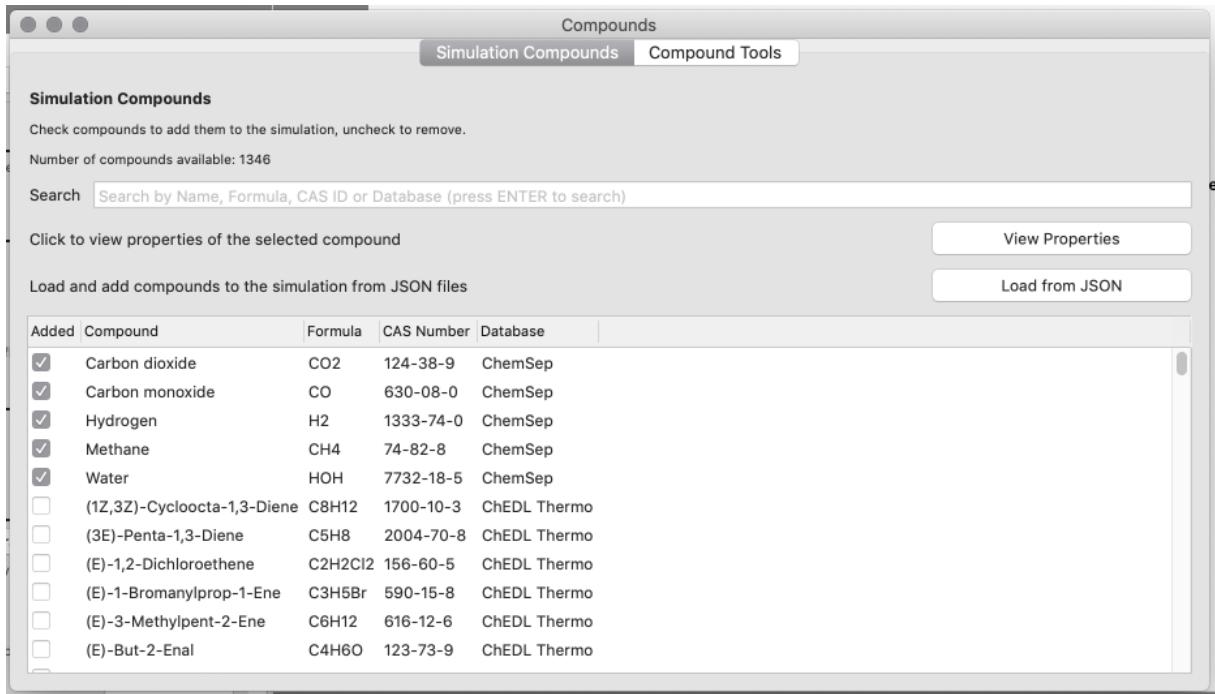


Figure 96: Compound Selection

3. Go to **Setup > Basis**, select **Peng-Robinson (PR)** on the **Add New Property Package** combobox to add a copy of this Property Package to the simulation.

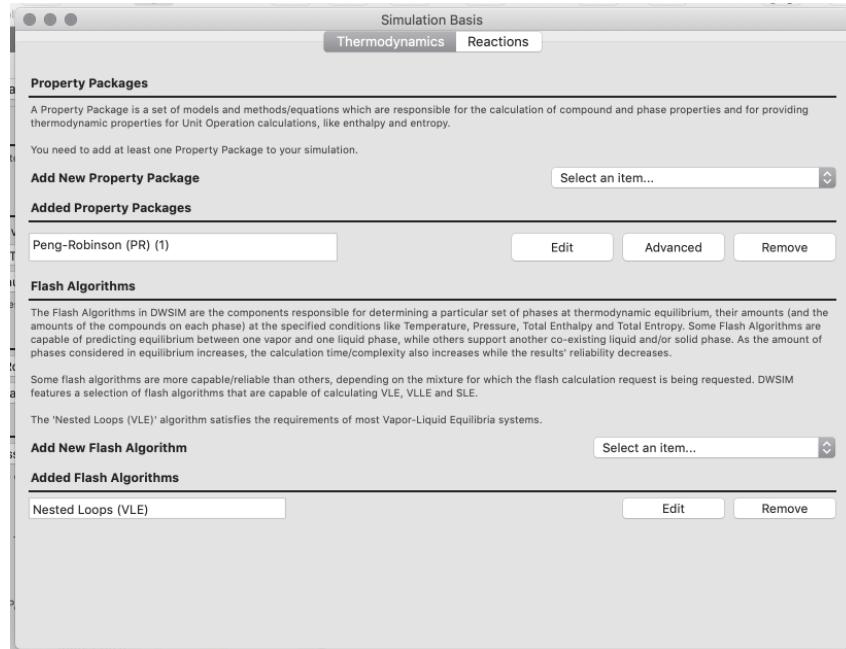


Figure 97: Property Package Selection

4. Go to the **Setup > Flowsheet Settings > System of Units** section and create a new System of Units, with the following setup:

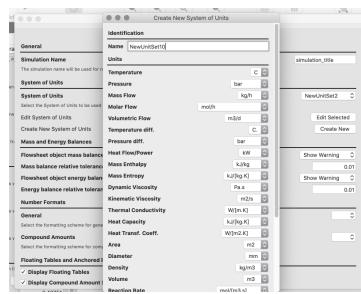


Figure 98: New System of Units

5. After creating this Units Set, select it on the System of Units combobox.
  6. Go to **Setup > Basis > Reactions** and create three new Heterogeneous Catalytic reactions, with the following configuration. The denominator expression is the same for all three reactions:
- $$(1+1.77E+5*\exp(-88680/8.314/T)*R1/P1+6.12E-9*\exp(82900/8.314/T)*P1+8.23E-5*\exp(70650/8.314/T)*R2)^{-1}$$

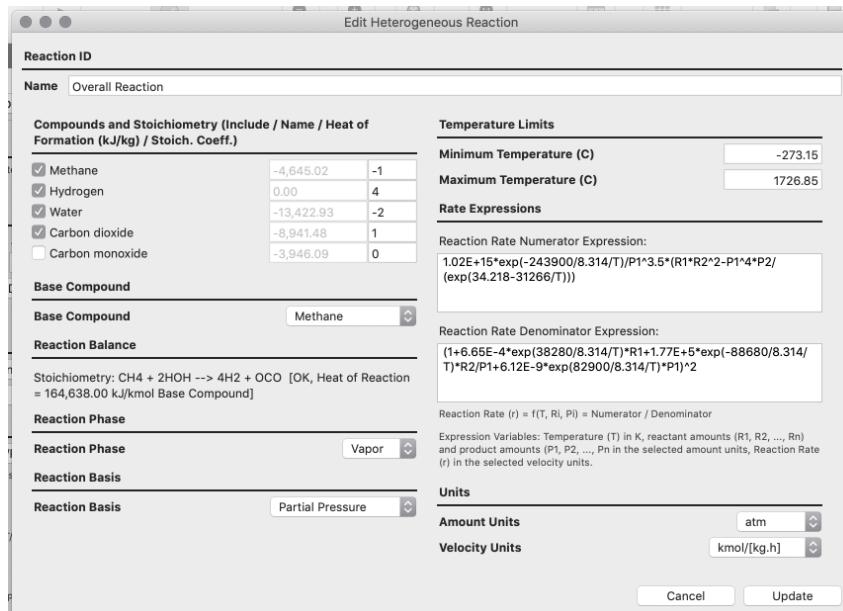


Figure 99: Overall Reaction setup

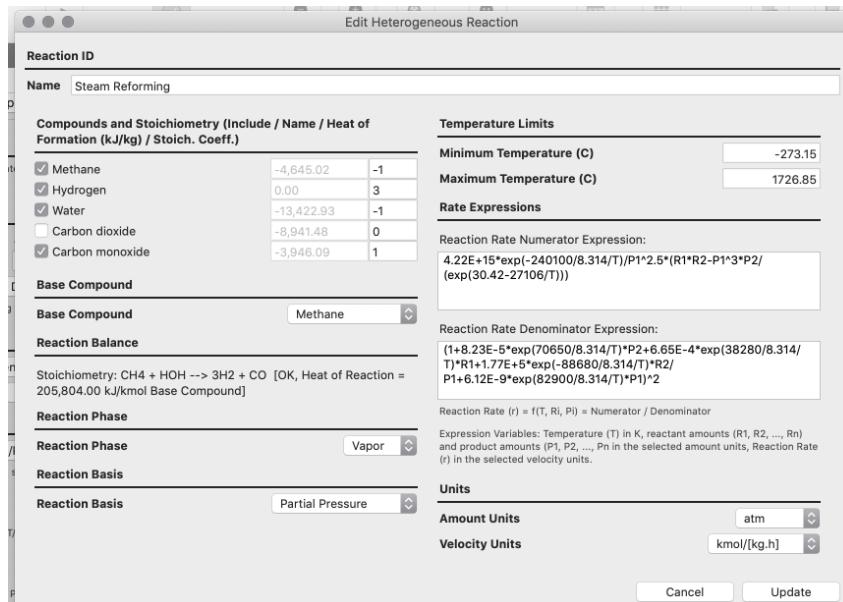


Figure 100: Steam Reforming Reaction setup

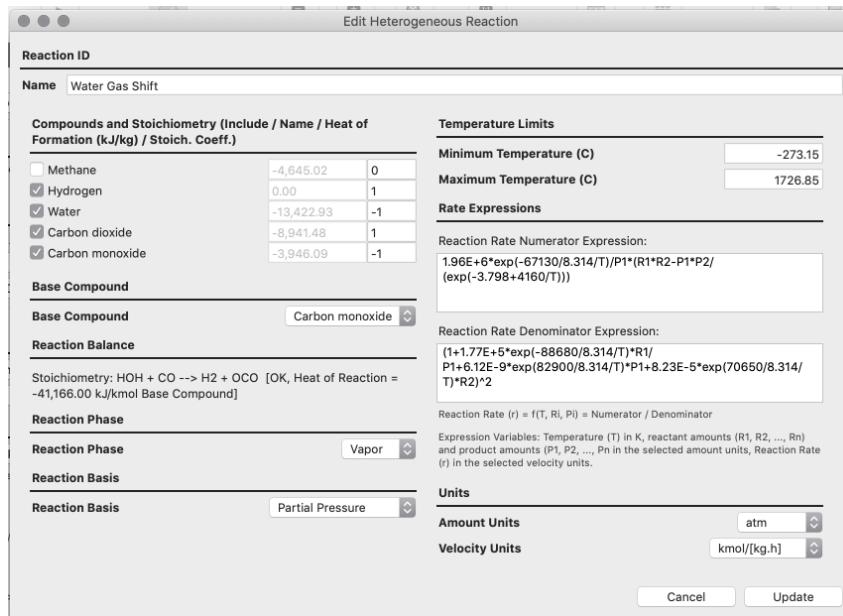


Figure 101: Water Gas Shift Reaction setup

- Close the Basis panel, and drag two material streams, one energy stream and one PFR from the **Object Palette** to the **Flowsheet PFD**. Connect the streams to the PFR as shown on the following figure:

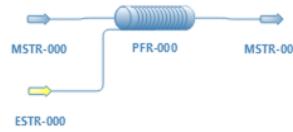


Figure 102: PFD setup

- Configure the inlet stream (MSTR-000) as follows (enter the temperature, pressure, molar flow and composition in molar fractions):

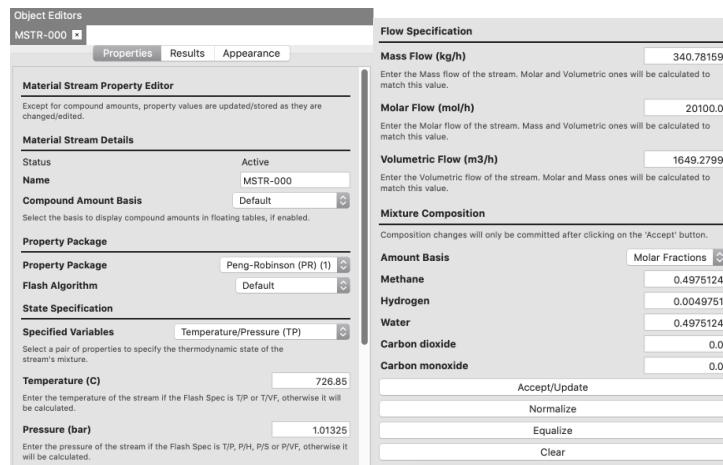


Figure 103: Inlet Stream setup

### 9. Configure the PFR as follows:

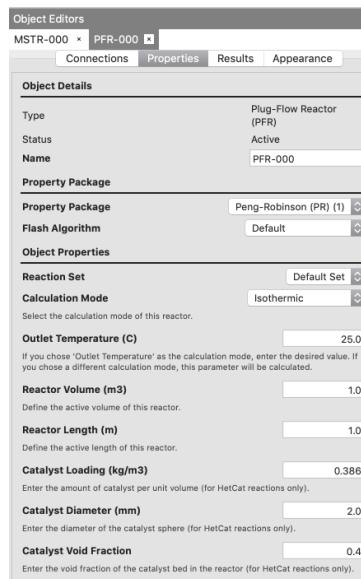


Figure 104: PFR setup

10. **Note:** when you create new reactions, they are automatically added to the **Default Reaction Set**. When you add new reactors to the flowsheet, they are automatically configured to use all supported and active reactions on the Default Reaction Set. You can create, edit and remove Reaction Sets at any time, and associate the individual reactors with different Reaction Sets too.
11. Run the simulation (press F5 or click on the **Solve Flowsheet** button on the toolbar). Wait for the calculation to finish.
12. Once finished, you should get the following results (methane conversion ~ **76.4%**):

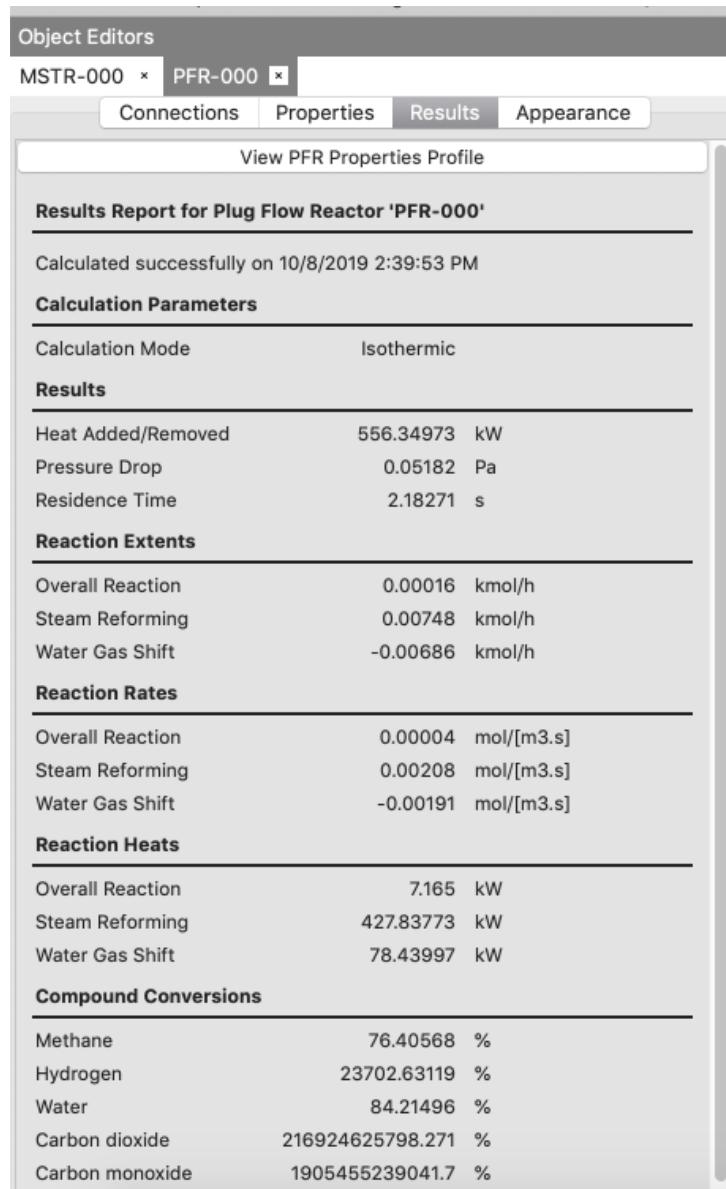


Figure 105: PFR calculation results

13. Create a new Sensitivity Analysis case to study the influence of the temperature on Methane conversion from 700 to 900 C. Go to **Tools > Sensitivity Analysis**.
14. Setup the Independent and Dependent Variables as follows:

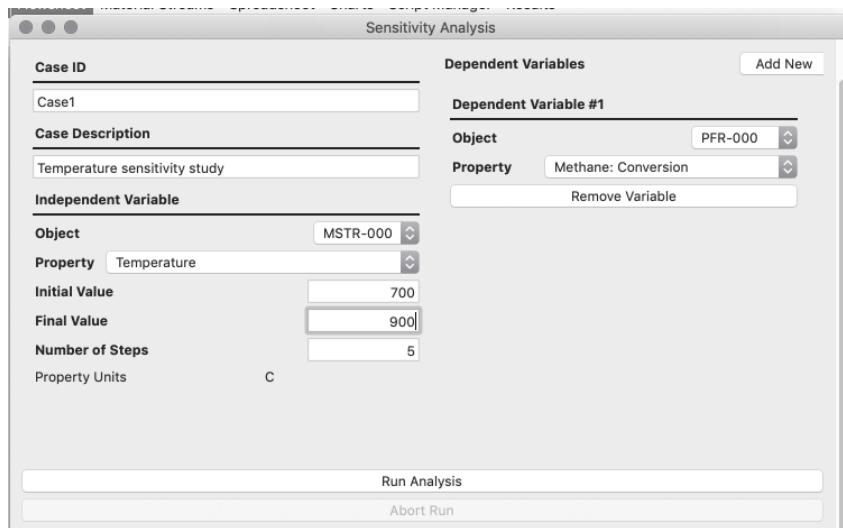


Figure 106: Sensitivity Analysis case setup

15. Run the Analysis. Wait for the calculations to finish (the **View Report** and **View Chart** buttons will become active).
16. Once finished, click on **View Chart**.

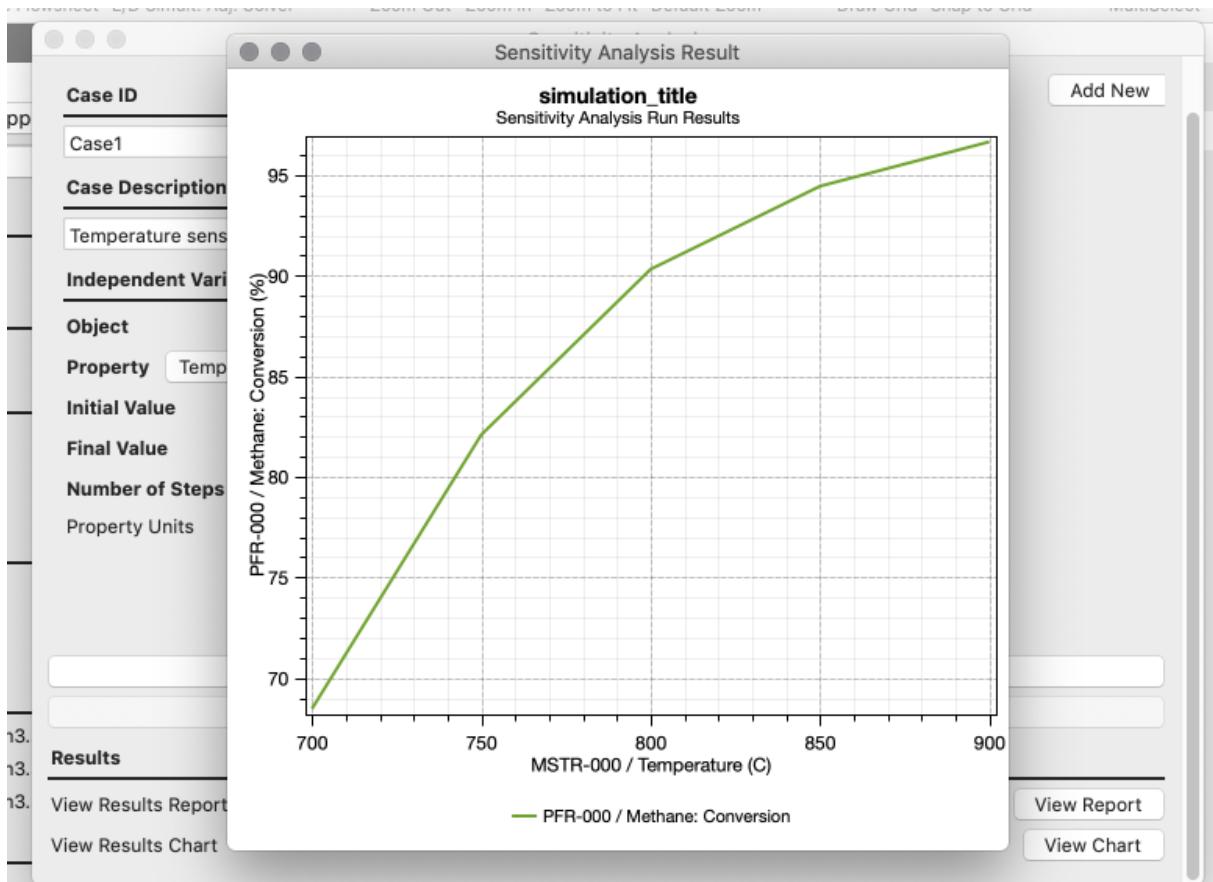


Figure 107: Sensitivity Analysis results

17. You can also view the concentration profile of the PFR using the Charts utility. Go to the **Charts tab**

on the main flowsheet window, click on **Add New 2D XY Chart**, select the PFR as the data source of the chart, select *Concentration Profile* as the **Chart Type** and click on **Update Chart Data**:

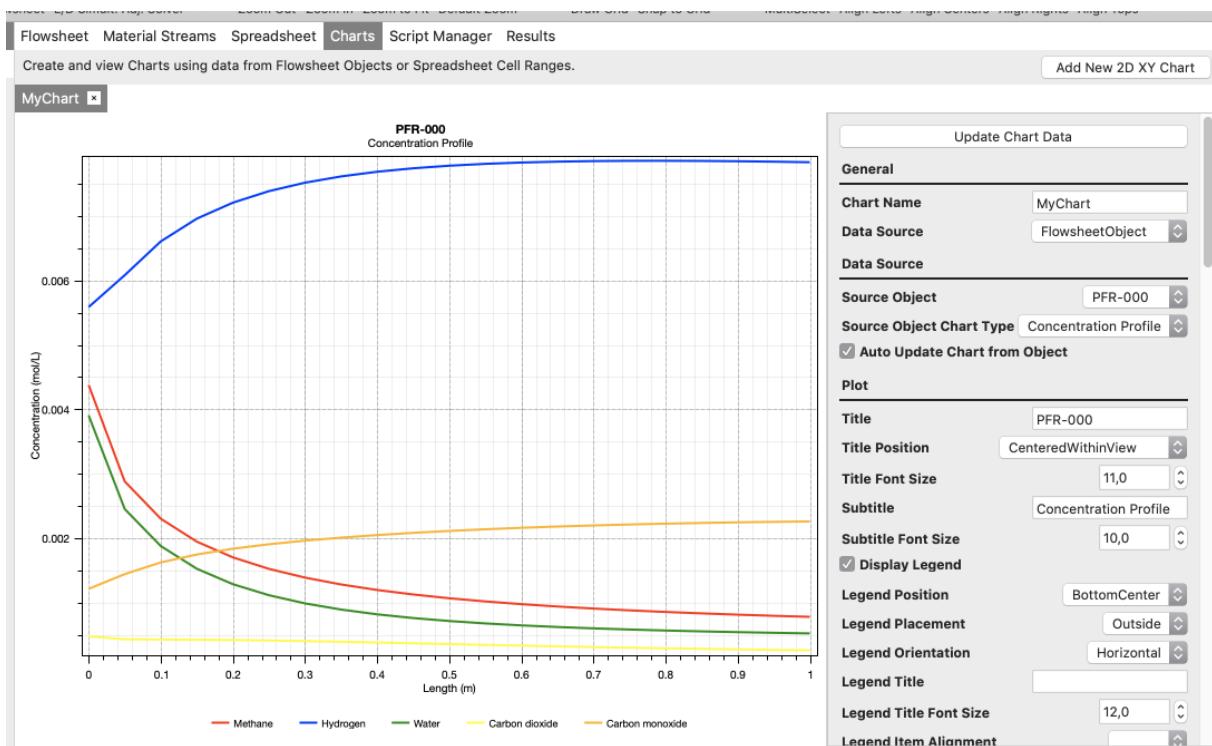


Figure 108: PFR concentration profile as visible on the Charts tool

## 26. Extractive Distillation

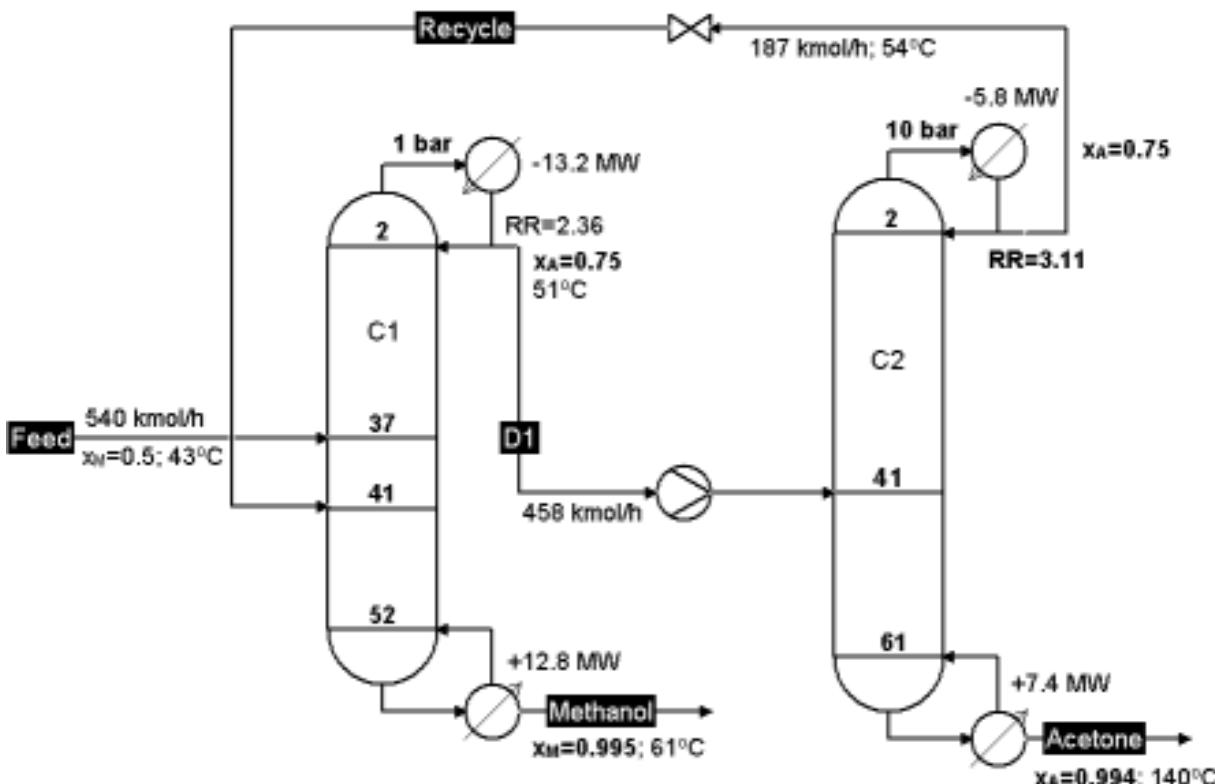
### 26.1. Introduction

This simulation is an example of Pressure Swing Azeotropic Distillation. Test case taken from COCO Simulator ([link](#)) (original author: Harry Kooijman - [www.chemsep.org](http://www.chemsep.org)). Adapted from Luyben et al., Ind. Eng. Chem. Res. (2008) 47 pp. 2696-2707.

### 26.2. Background

Methanol and acetone form a minimum temperature azeotrope but the composition of this azeotrope is sensitive to the pressure. We can make use of this to separate the two components into pure products by operating two columns at different pressures.

**Pressure Swing Distillation of Acetone-Methanol**  
 Ind. Eng. Chem. Res. (2008) 47 pp. 2696-2707



Copyright (c) 2012 ChemSep.org

Figure 109: Process Flowsheet.

### 26.3. DWSIM Model (Classic UI)

1. Create a New Steady State Simulation. Close the Simulation Wizard.



*Remember to Save your simulation at the end of each step.*

2. Go to **Edit > Simulation Settings > Compounds**, and select Methanol and Acetone to add these compounds to the simulation.

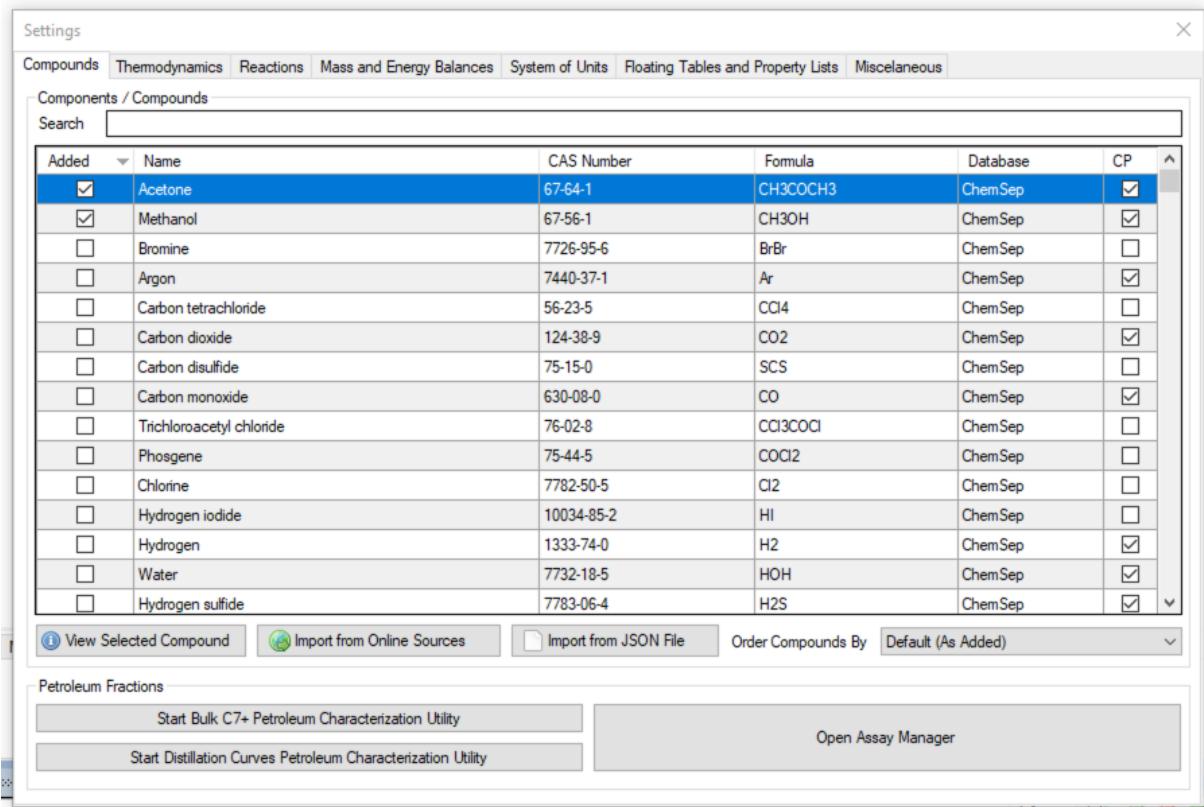


Figure 110: Compound Selection

3. Go to **Thermodynamics** tab, select **NRTL** on the Available Property Packages section and click **Add**. Add an instance of the Inside-Out VLE Flash Algorithm too.

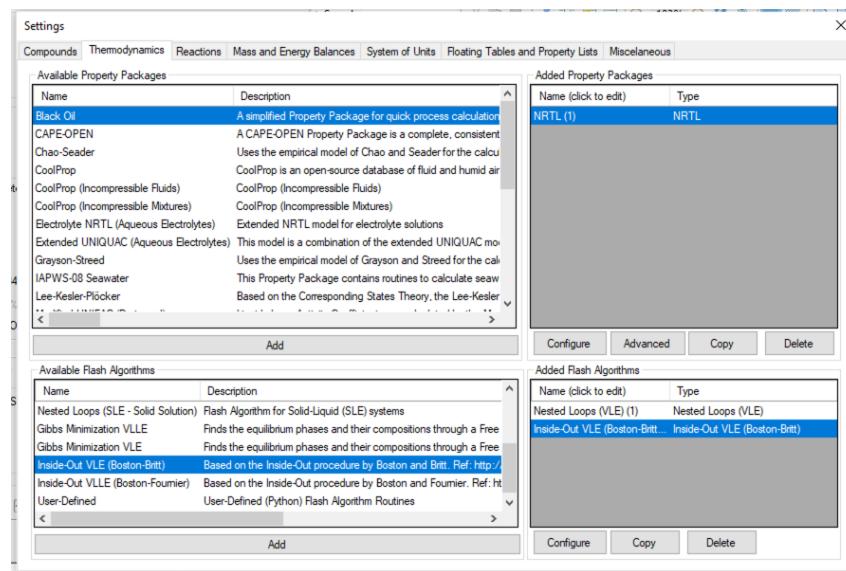


Figure 111: Property Package and Flash Algorithm Selection

4. Check if the NRTL Interaction Parameters are all set (click on **Configure** on the Added Property Packages section).

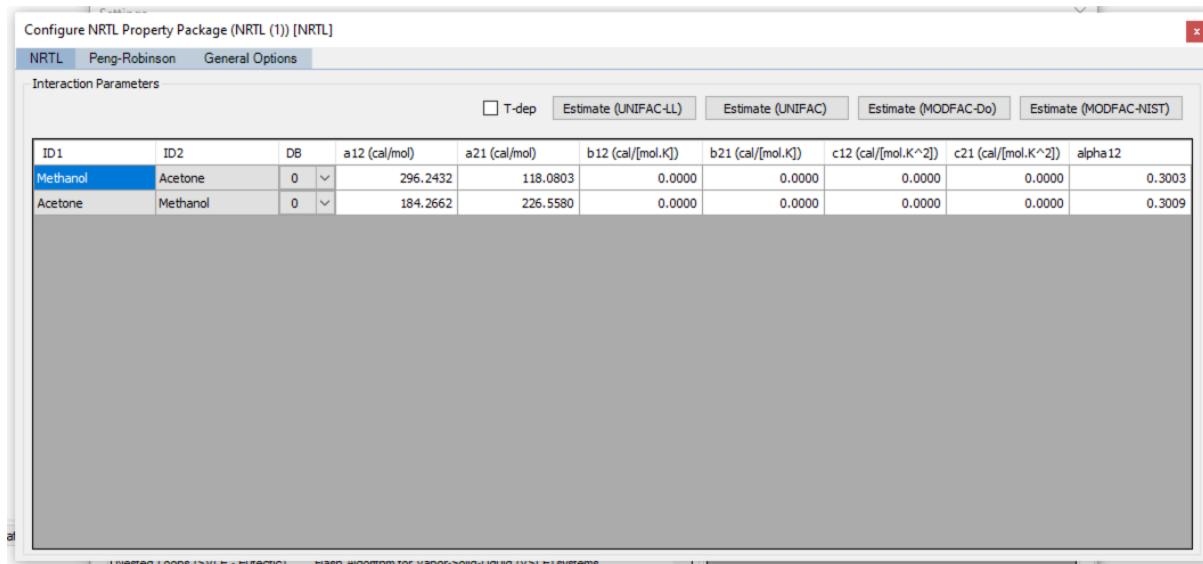


Figure 112: NRTL Interaction Parameters for Methanol/Acetone

5. Go to the **System of Units** tab and create a new System of Units, with the following setup:

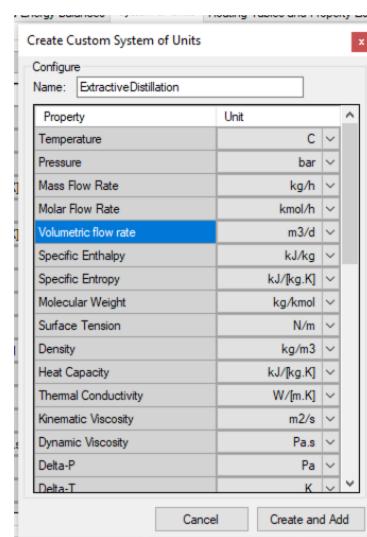


Figure 113: New System of Units

6. After creating this Units Set, select it on the System of Units combobox.

7. Add the objects to the flowsheet (streams, pump, valve, recycle and distillation columns) as depicted on the following figure, renaming them as required. You'll setup the connections between them later on this tutorial.

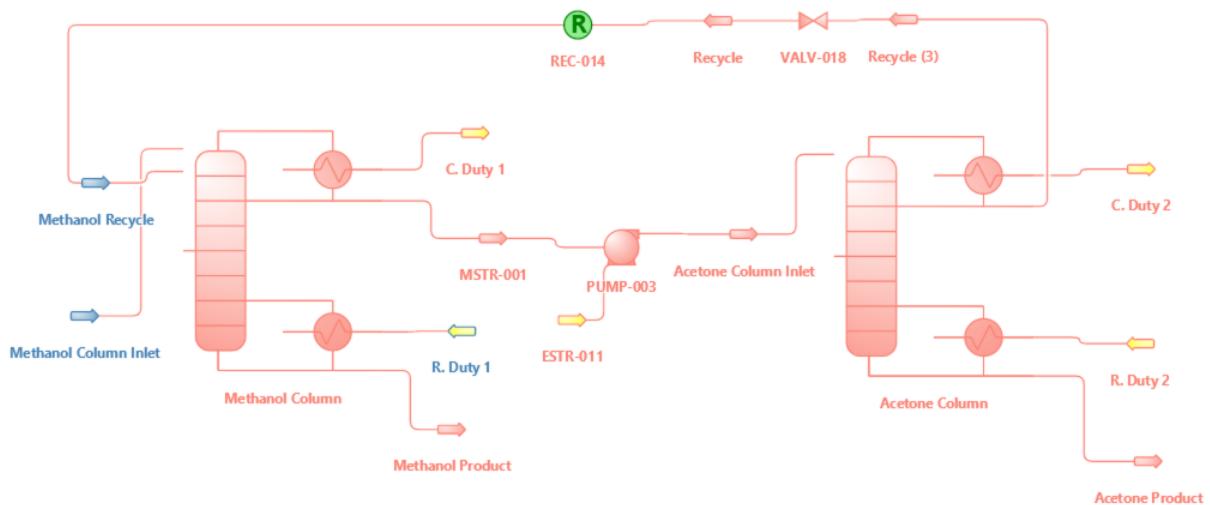


Figure 114: Process Flowsheet Diagram

8. Disable automatic calculation of the flowsheet.

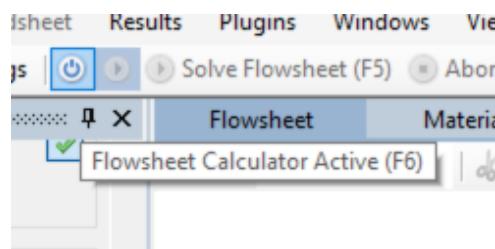


Figure 115: Enable/Disable Flowsheet Calculator/Solver

9. Setup the columns and their connections as follows:

a) Methanol Column:

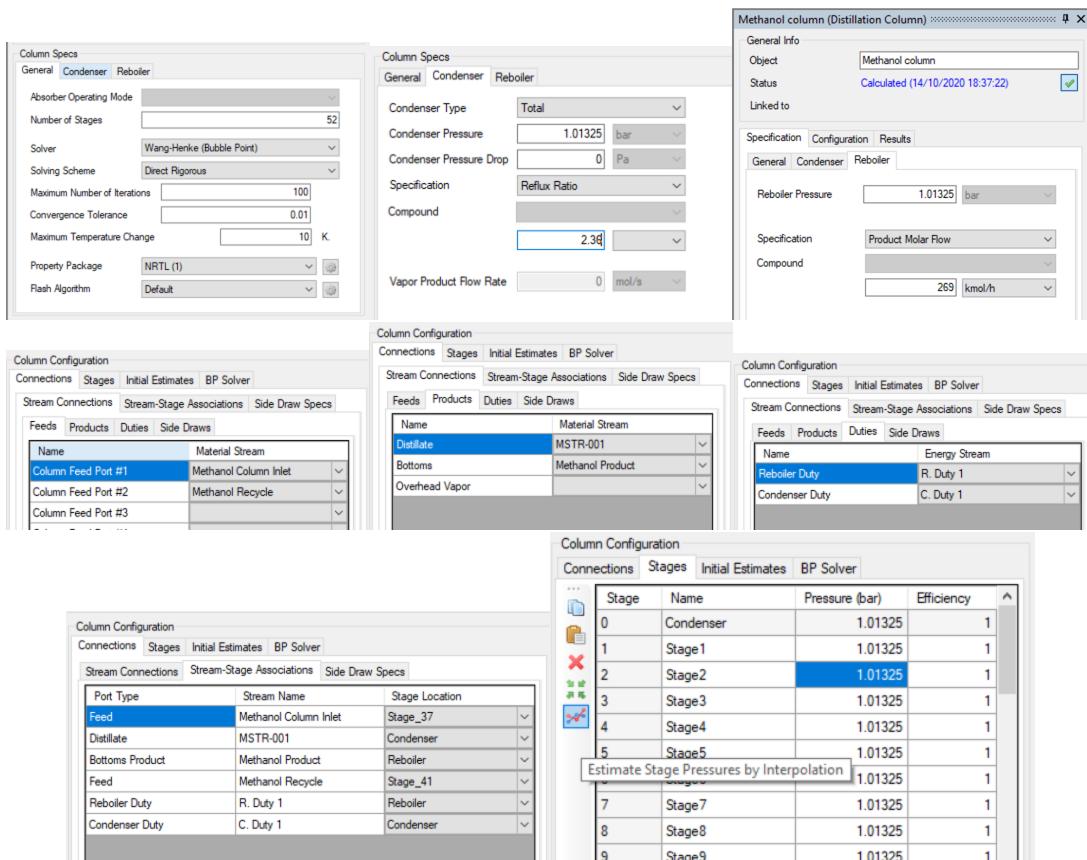


Figure 116: Methanol Column configuration

b) Acetone Column:

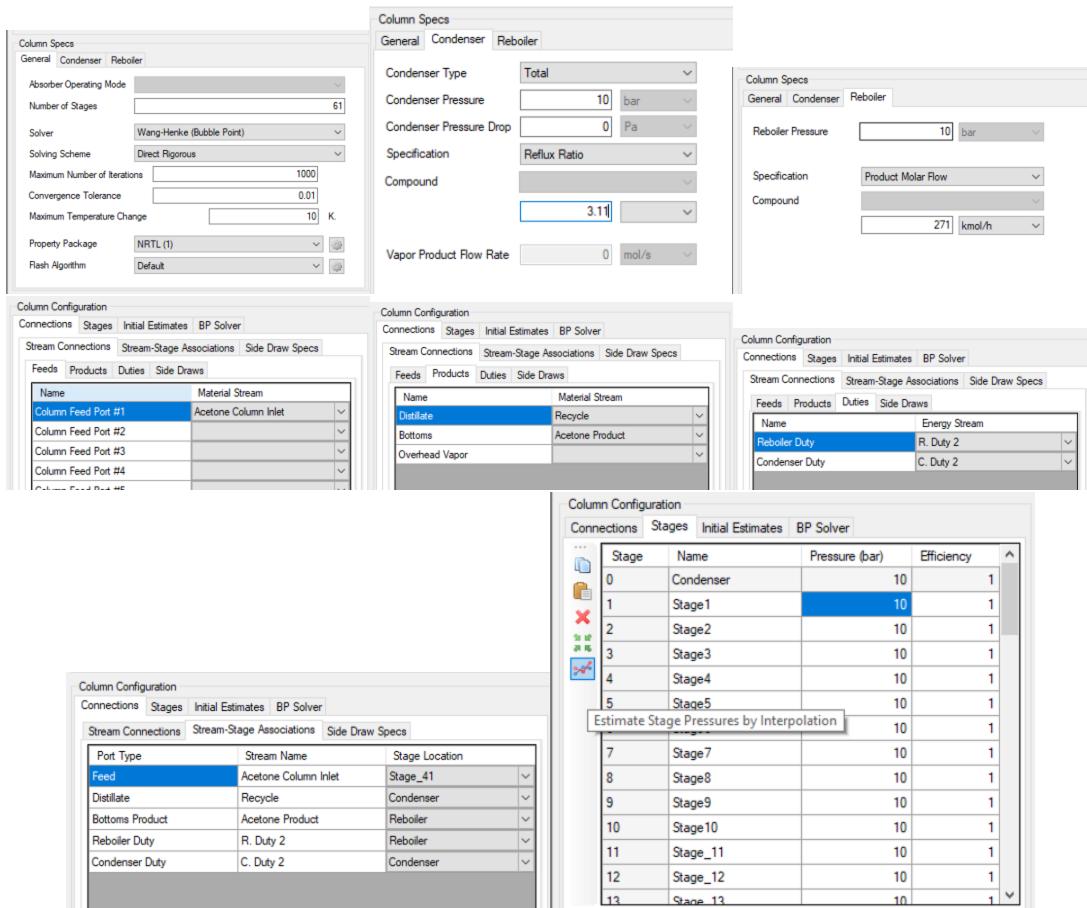


Figure 117: Acetone Column configuration

10. Enter initial estimates for the temperature profile of the Acetone Column, and check the **T** checkbox so DWSIM can use them. Insert only the boundary values (condenser and reboiler) and click on **Interpolate** to calculate the inner stage values.

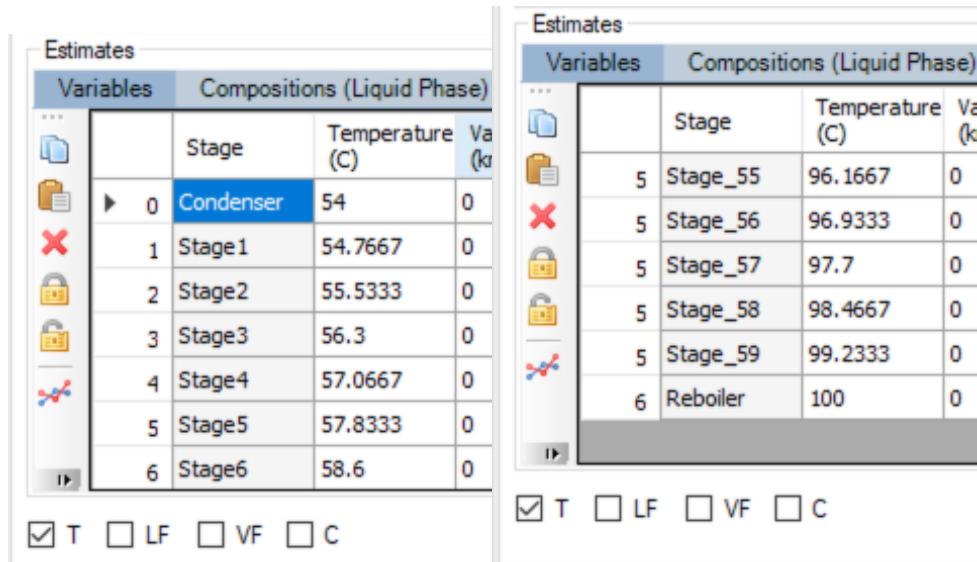


Figure 118: Acetone Column initial estimates for temperature profile

11. After the columns are correctly configured and connected to their associated streams, setup the pump, valve and recycle connections using their Editor Panels.
12. Setup the pump and valve properties as follows:

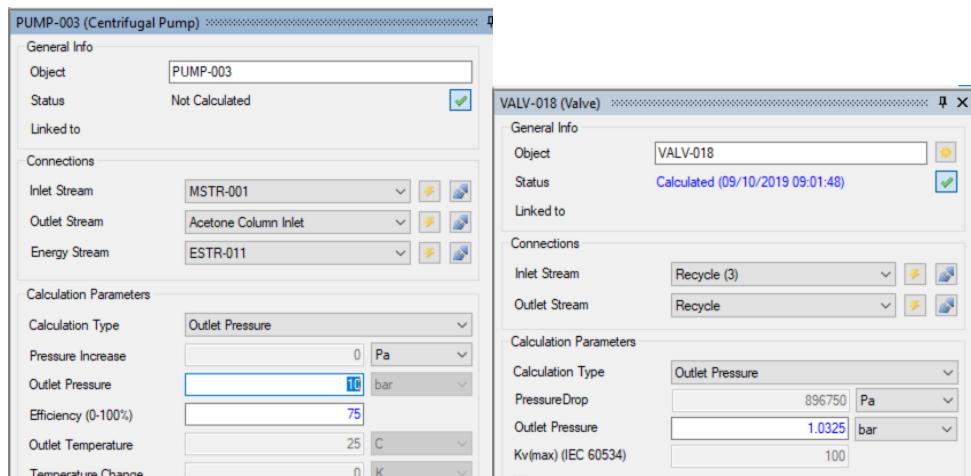


Figure 119: Pump and Valve properties

13. Configure the Methanol Inlet Stream as follows:

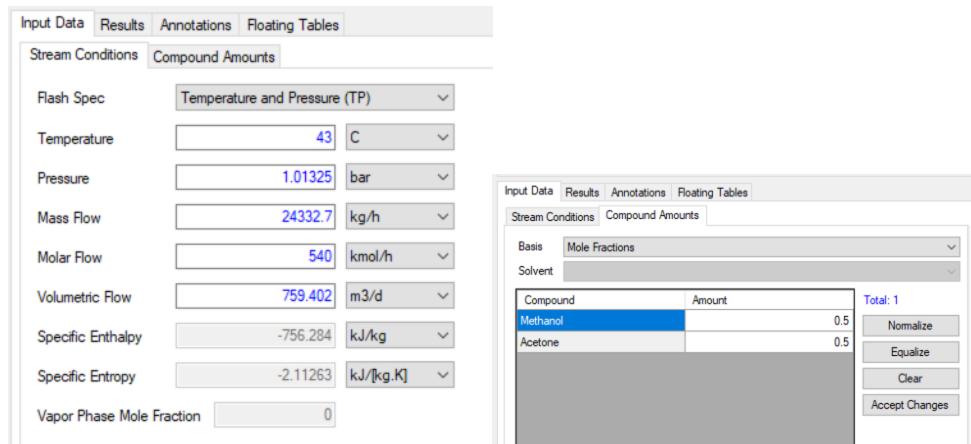


Figure 120: Methanol Inlet Stream configuration

14. Configure the Methanol Recycle Stream (initial estimates for the recycle stream) as follows:

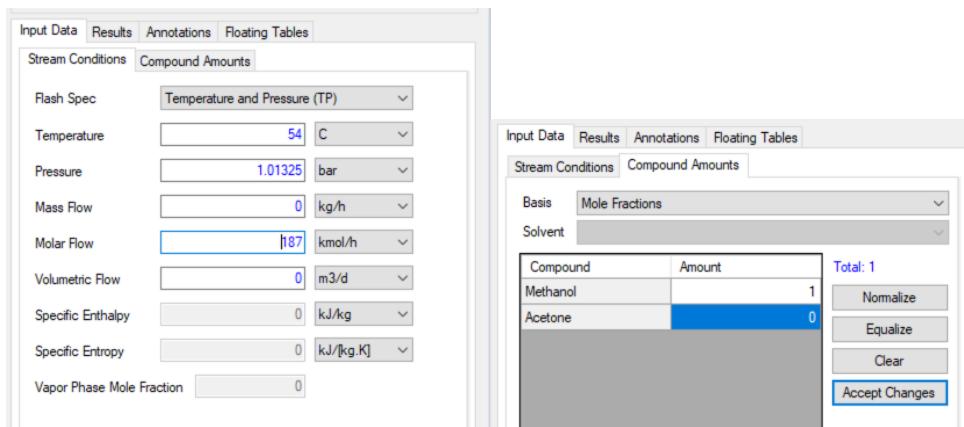


Figure 121: Methanol Recycle Stream configuration

15. Associate the Inside-Out Flash with the following streams: **MSTR-001**, **Methanol Product**, **Acetone Product** and **Recycle (3)**. This will prevent PH Flash errors from happening during the flowsheet calculation.

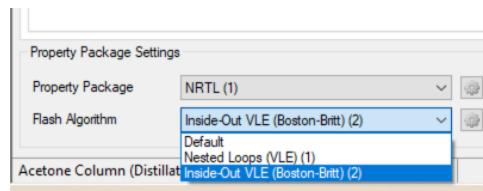


Figure 122: Associating the Inside-Out flash with the streams

16. Re-enable the solver (press F6) and calculate the flowsheet (press F5). Wait for the recycle to converge.
17. After the flowsheet solves, insert a new Property Table:

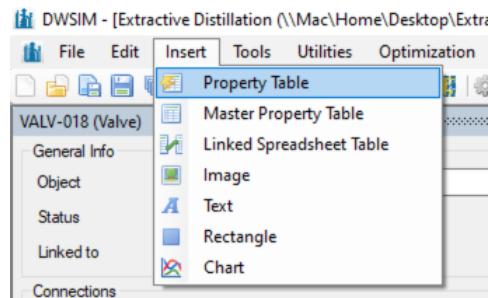


Figure 123: Inserting a Property Table

18. Double-click on the inserted table, search for the column energy streams and select Energy Flow for all of them, so these values can be shown on the Property Table.

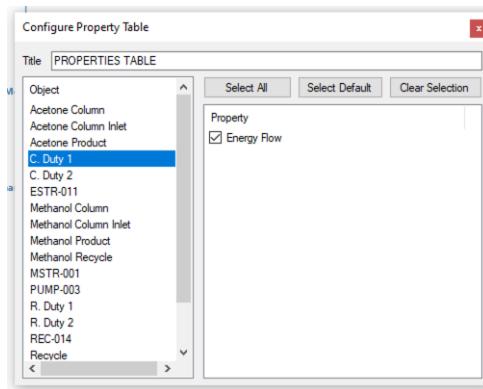


Figure 124: Setting up a Property Table

19. Compare the results obtained with the duties specified in the original problem.

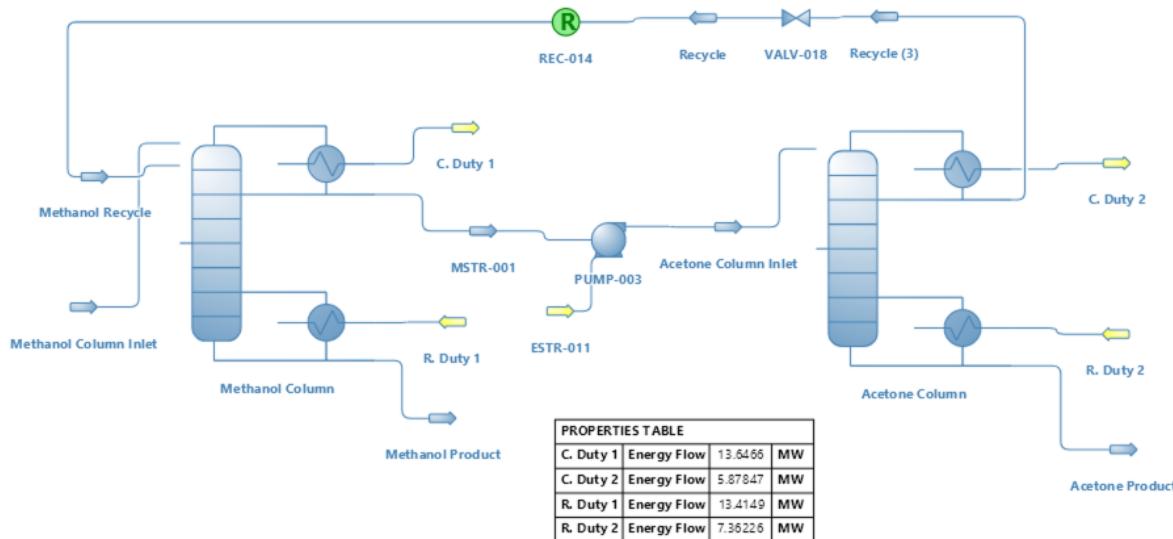


Figure 125: Final results

## 26.4. DWSIM Model (Cross-Platform UI)

1. Create a New Simulation. Close the Simulation Wizard.



*Remember to Save your simulation at the end of each step.*

2. Go to **Setup > General Settings, Interface** and disable the **Call Solver on Editor Property Update** option to avoid unnecessary calculations during the model building.
3. Go to **Setup > Compounds**, and select Methanol and Acetone to add these compounds to the simulation.

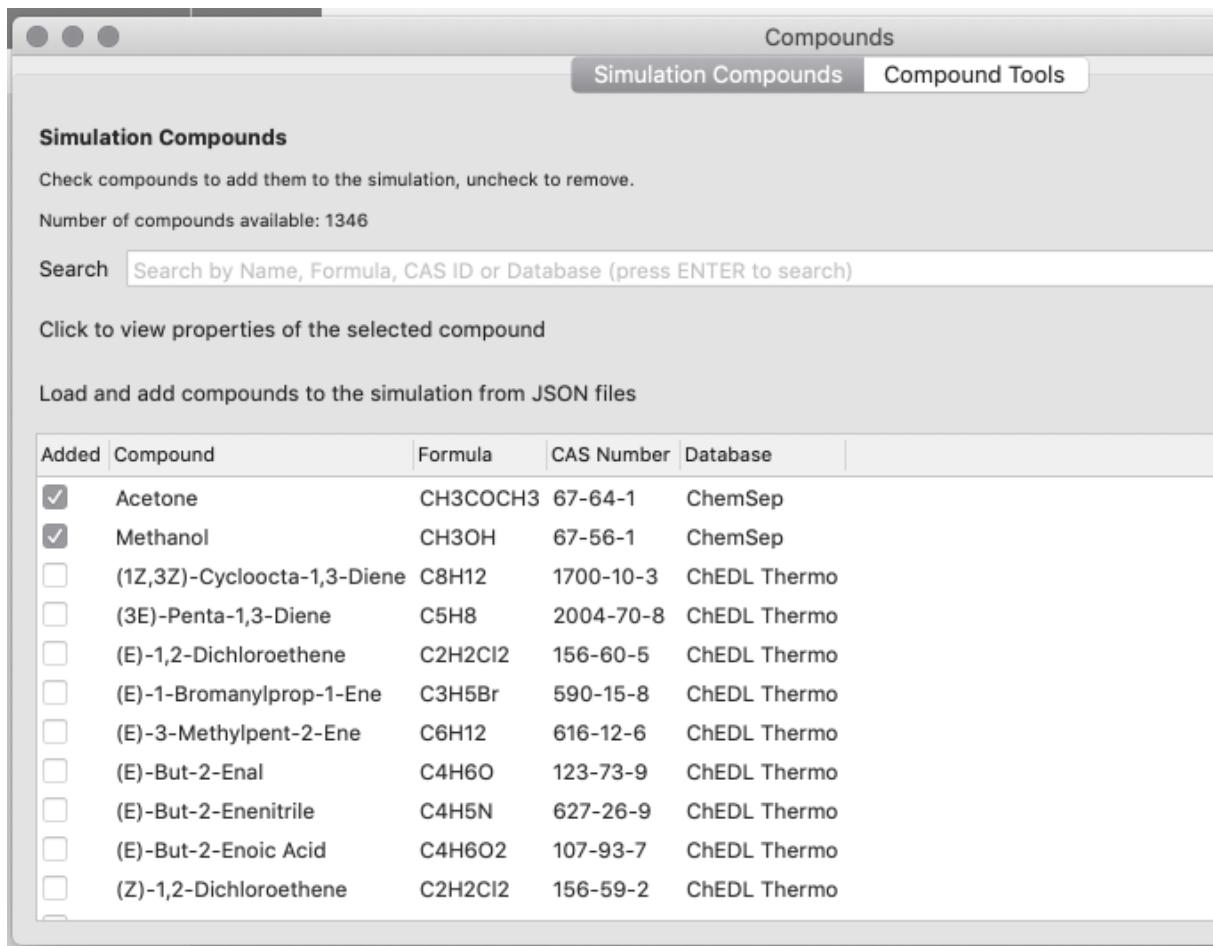


Figure 126: Compound Selection

4. Go to **Setup > Basis > Thermodynamics** tab, add a copy of the **NRTL** Property Package. Add an instance of the **Inside-Out VLE Flash Algorithm** too.

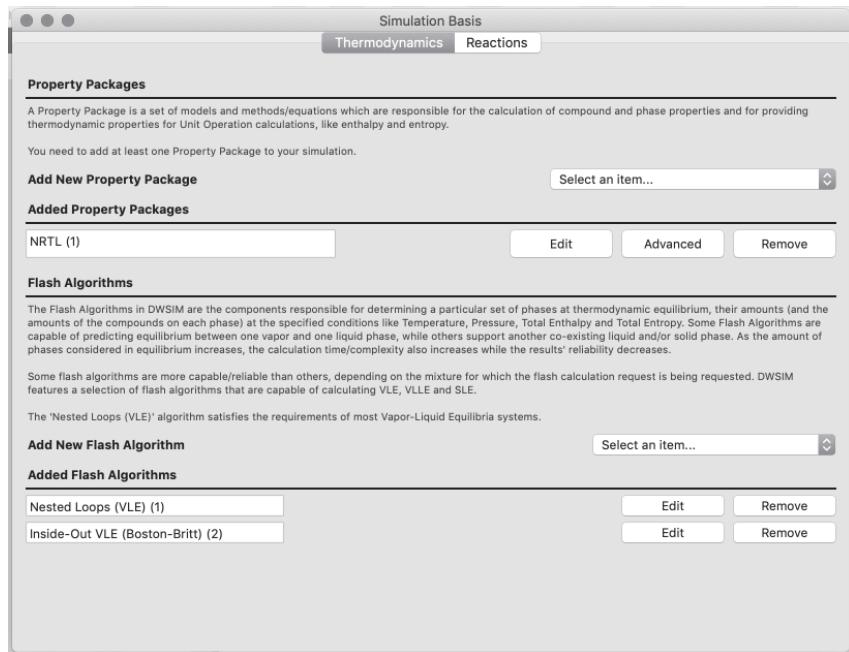


Figure 127: Property Package and Flash Algorithm Selection

- Check if the NRTL Interaction Parameters are all set (click on **Edit** on the Added Property Packages section).

Simulation Basis	
Edit 'NRTL (1)' (NRTL)	
	Interaction Parameters
Methanol/Acetone A12	296.2432
Methanol/Acetone A21	118.0803
Methanol/Acetone B12	0.0000
Methanol/Acetone B21	0.0000
Methanol/Acetone C12	0.0000
Methanol/Acetone C21	0.0000
Methanol/Acetone alpha12	0.3003
Acetone/Methanol A12	184.2662
Acetone/Methanol A21	226.5580
Acetone/Methanol B12	0.0000
Acetone/Methanol B21	0.0000
Acetone/Methanol C12	0.0000
Acetone/Methanol C21	0.0000
Acetone/Methanol alpha12	0.3009

Figure 128: NRTL Interaction Parameters for Methanol/Acetone

- Go to the **Setup > Flowsheet Settings > System of Units** section and create a new System of

Units, with the following setup:

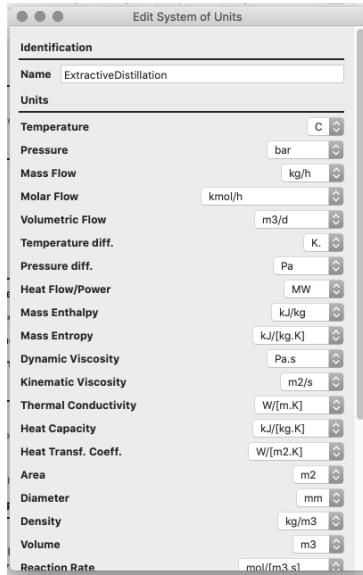


Figure 129: New System of Units

7. After creating this Units Set, select it on the System of Units combobox.
8. Add the objects to the flowsheet (streams, pump, valve, recycle and distillation columns) as depicted on the following figure, renaming them as required. You'll setup the connections between them later on this tutorial.

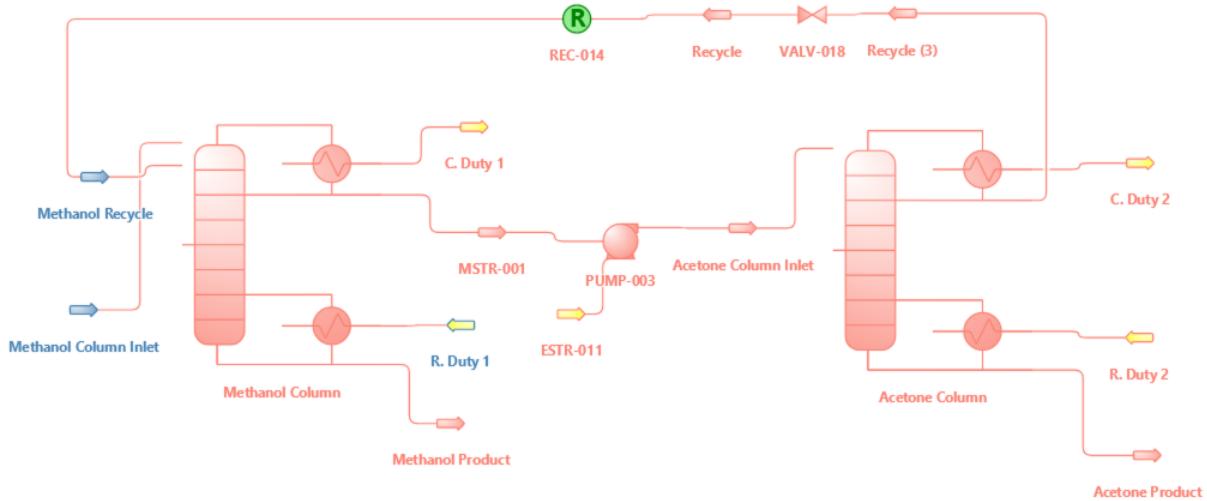


Figure 130: Process Flowsheet Diagram

9. Setup the columns and their connections as follows:

- a) Methanol Column (52 stages):

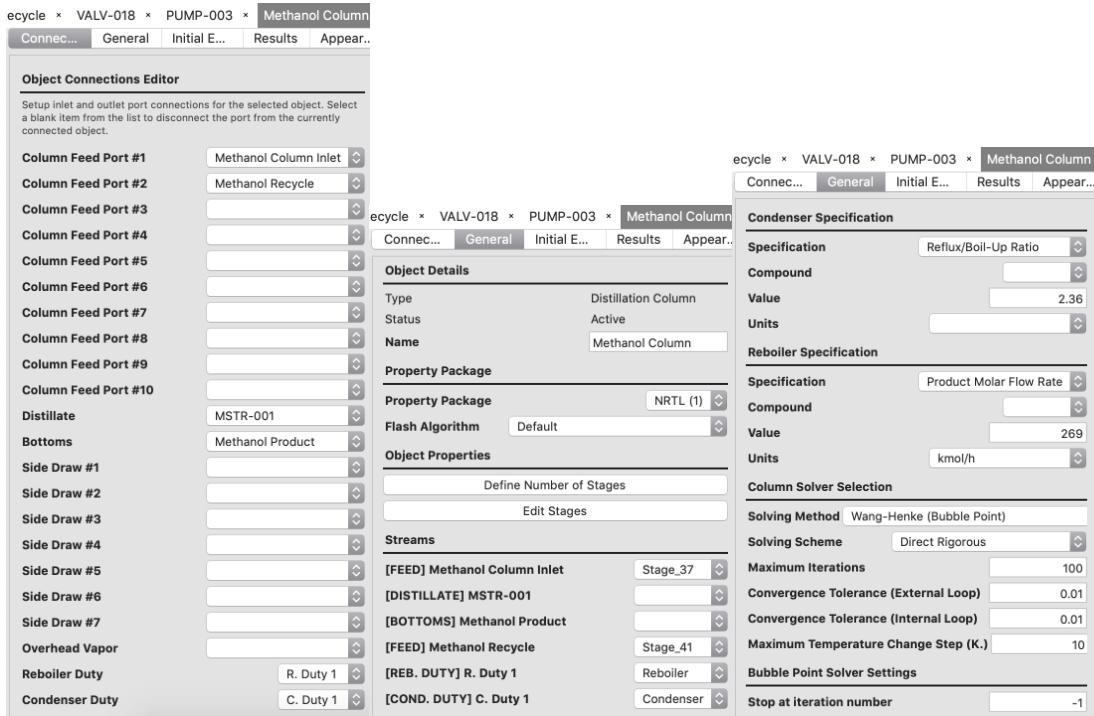


Figure 131: Methanol Column configuration

b) Acetone Column (61 stages):

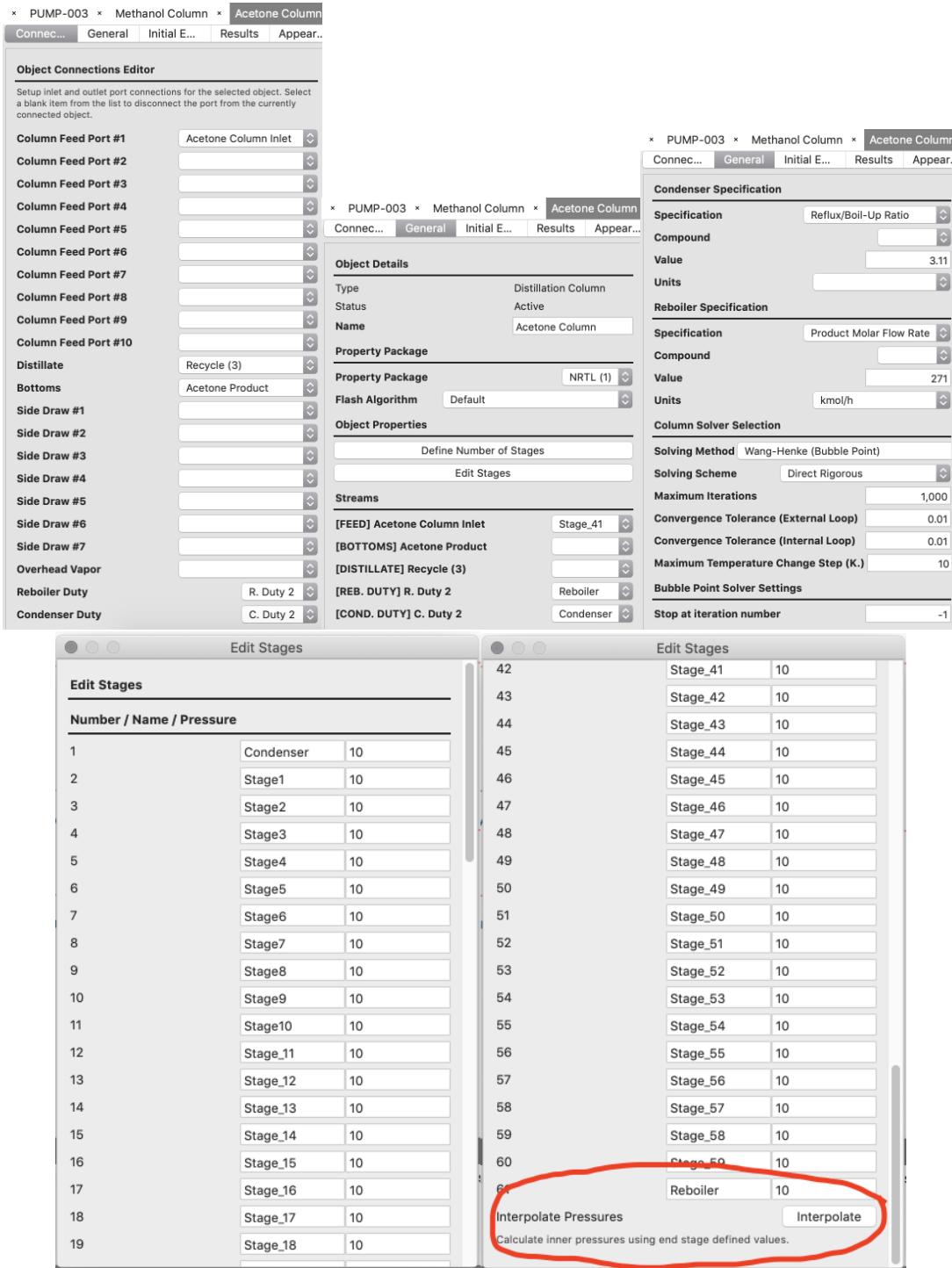


Figure 132: Acetone Column configuration

- Enter initial estimates for the temperature profile of the Acetone Column, and check the **Override Temperature Estimates** checkbox so DWSIM can use them. Values must be tab separated, inserted as in the picture below:

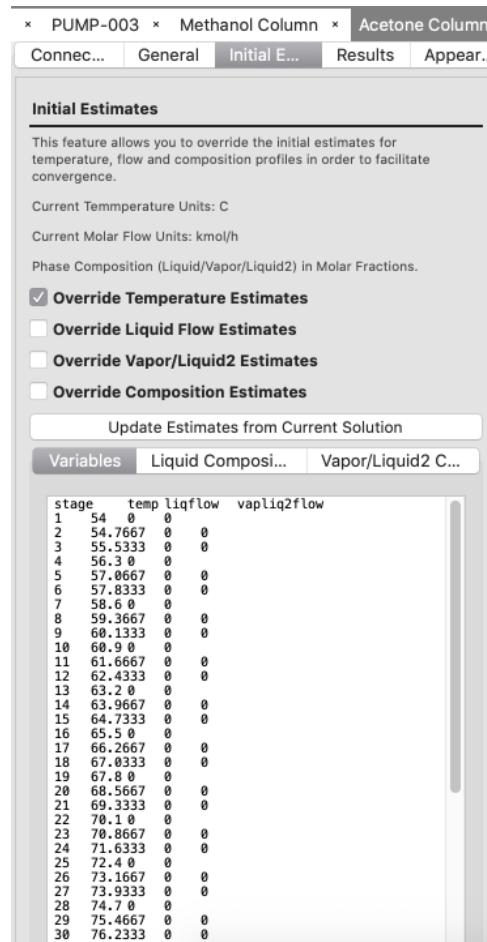


Figure 133: Acetone Column initial estimates for temperature profile

11. After the columns are correctly configured and connected to their associated streams, setup the pump, valve and recycle connections using their Editor Panels.
12. Setup the pump and valve properties as follows:

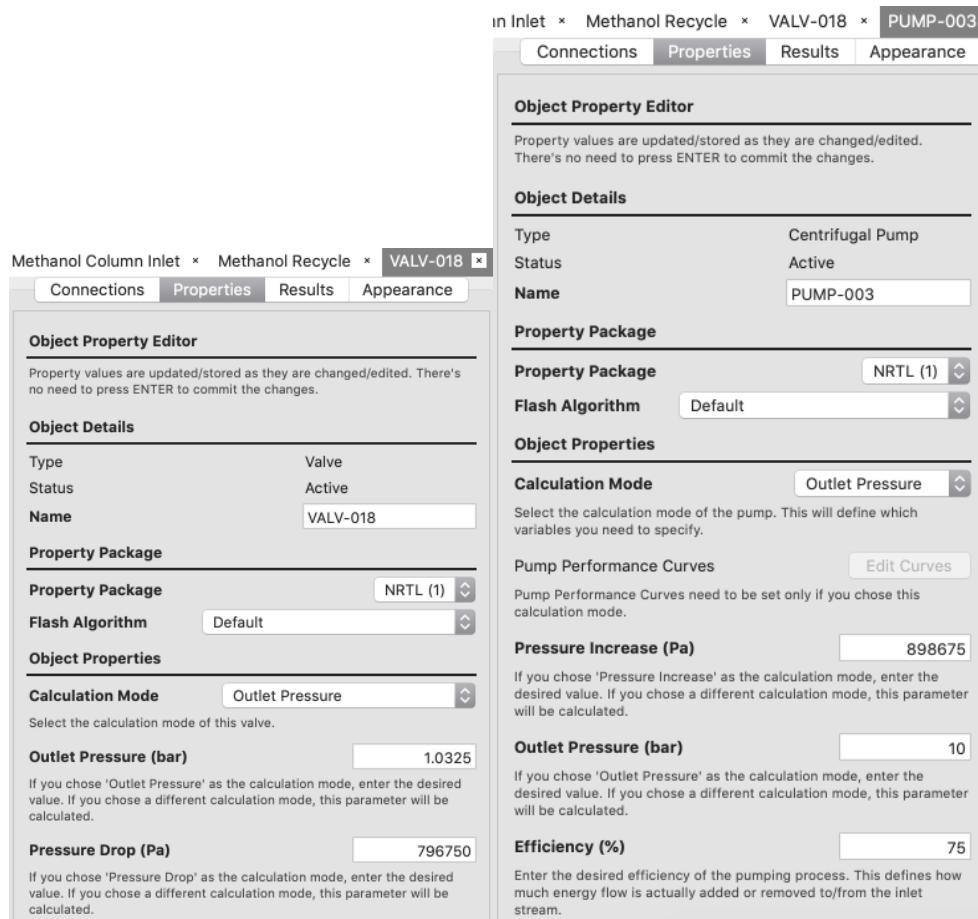


Figure 134: Pump and Valve properties

13. Configure the Methanol Inlet Stream as follows:

**Methanol Column Inlet**

Properties	Results	Appearance
<b>Temperature (C)</b>	54.9572	
Enter the temperature of the stream if the Flash Spec is T/P or T/VF, otherwise it will be calculated.		
<b>Pressure (bar)</b>	1.01325	
Enter the pressure of the stream if the Flash Spec is T/P, P/H, P/S or P/VF, otherwise it will be calculated.		
<b>Specific Enthalpy (kJ/kg)</b>	-721.122	
Enter the enthalpy of the stream if the Flash Spec is P/H, otherwise it will be calculated.		
<b>Specific Entropy (kJ/[kg.K])</b>	-2.00346	
Enter the entropy of the stream if the Flash Spec is P/S, otherwise it will be calculated.		
<b>Vapor Phase Mole Fraction (spec)</b>	0	
If the Flash Spec is T/VF or P/VF, enter the vapor phase mole fraction (quality) of the stream, otherwise it will be calculated.		
<b>Flow Specification</b>		
<b>Mass Flow (kg/h)</b>	24332.7	
Enter the Mass flow of the stream. Molar and Volumetric ones will be calculated to match this value.		
<b>Molar Flow (kmol/h)</b>	540	
Enter the Molar flow of the stream. Mass and Volumetric ones will be calculated to match this value.		
<b>Volumetric Flow (m<sup>3</sup>/d)</b>	772.504	
Enter the Volumetric flow of the stream. Molar and Mass ones will be calculated to match this value.		
<b>Mixture Composition</b>		
Composition changes will only be committed after clicking on the 'Accept' button.		
<b>Amount Basis</b>	Molar Fractions	
<b>Methanol</b>	0.5	
<b>Acetone</b>	0.5	

Figure 135: Methanol Inlet Stream configuration

14. Configure the Methanol Recycle Stream (initial estimates for the recycle stream) as follows:

Methanol Column Inlet x Methanol Recycle x

**Properties**   **Results**   **Appearance**

---

**Temperature (C)**  Enter the temperature of the stream if the Flash Spec is T/P or T/VF, otherwise it will be calculated.

**Pressure (bar)**  Enter the pressure of the stream if the Flash Spec is T/P, P/H, P/S or P/VF, otherwise it will be calculated.

**Specific Enthalpy (kJ/kg)**  Enter the enthalpy of the stream if the Flash Spec is P/H, otherwise it will be calculated.

**Specific Entropy (kJ/[kg.K])**  Enter the entropy of the stream if the Flash Spec is P/S, otherwise it will be calculated.

**Vapor Phase Mole Fraction (spec)**  If the Flash Spec is T/VF or P/VF, enter the vapor phase mole fraction (quality) of the stream, otherwise it will be calculated.

---

**Flow Specification**

**Mass Flow (kg/h)**  Enter the Mass flow of the stream. Molar and Volumetric ones will be calculated to match this value.

**Molar Flow (kmol/h)**  Enter the Molar flow of the stream. Mass and Volumetric ones will be calculated to match this value.

**Volumetric Flow (m<sup>3</sup>/d)**  Enter the Volumetric flow of the stream. Molar and Mass ones will be calculated to match this value.

---

**Mixture Composition**

Composition changes will only be committed after clicking on the 'Accept' button.

<b>Amount Basis</b>	<b>Molar Fractions</b> <input type="button" value="▼"/>
<b>Methanol</b>	<input type="text" value="1"/>
<b>Acetone</b>	<input type="text" value="0"/>

Figure 136: Methanol Recycle Stream configuration

15. Associate the Inside-Out Flash with the following streams: **MSTR-001**, **Methanol Product**, **Acetone Product** and **Recycle (3)**. This will prevent PH Flash errors from happening during the flow-sheet calculation.

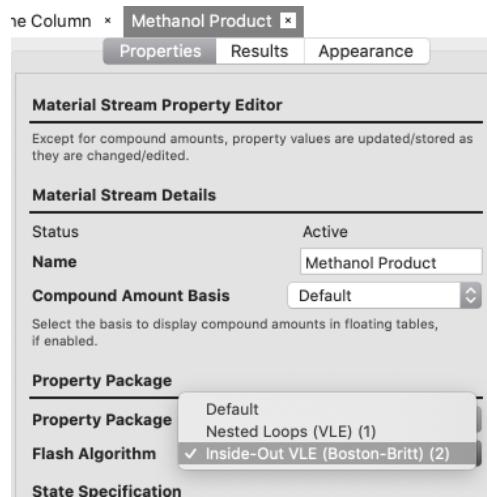


Figure 137: Associating the Inside-Out flash with the streams

16. Calculate the flowsheet (press F5). Wait for the recycle to converge.
17. After the flowsheet solves, insert a new Property Table:

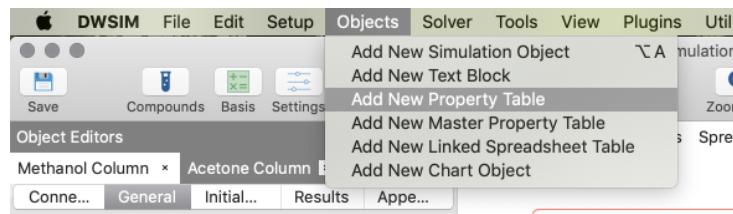


Figure 138: Inserting a Property Table

18. Double-click on the inserted table, search for the column energy streams and select Energy Flow for all of them, so these values can be shown on the Property Table.

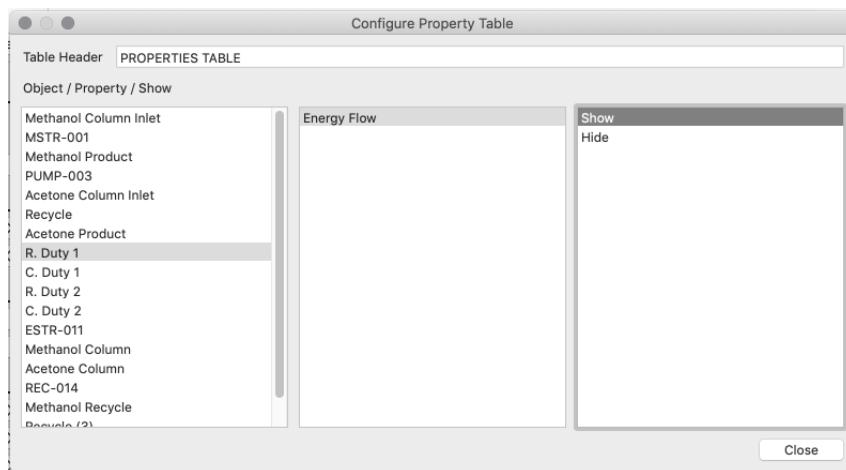


Figure 139: Setting up a Property Table

19. Compare the results obtained with the duties specified in the original problem.

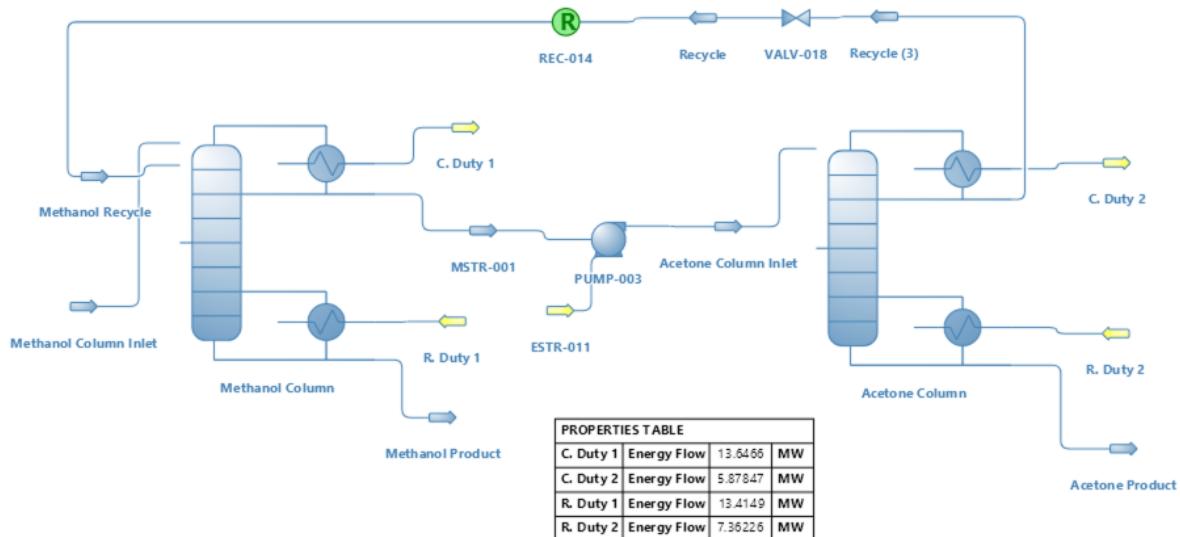


Figure 140: Final results

## 27. Flowsheet Control with Python Scripts

### 27.1. Introduction

Let us study the effect of the pressure on the temperature profile of the Acetone Column created on the previous tutorial. We will use the **IronPython Scripting**, **Spreadsheet** and **Charts** features available in DWSIM to generate, organize and analyze the results.

### 27.2. DWSIM Model (Classic UI)

1. Save the previous simulation with a different file name and remove everything from the flowsheet except the objects depicted on the following picture:

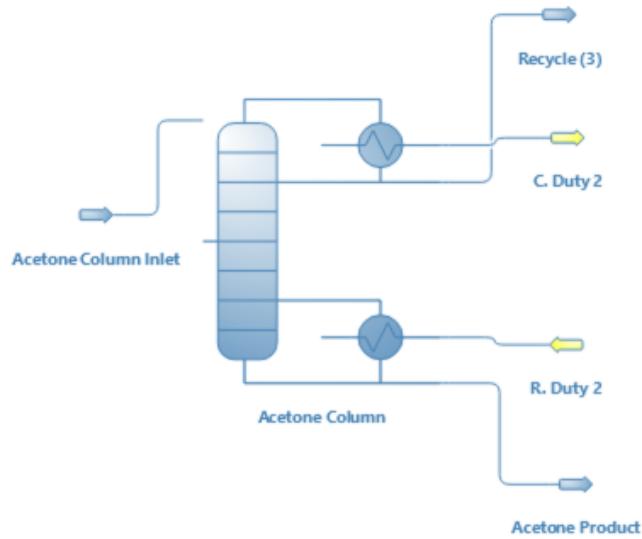


Figure 141: Process Flowsheet Diagram

2. Go to the **Script Manager** and enter the following script:

```

1 # Get Acetone Column object reference from the Flowsheet
2
3 column = Flowsheet.GetFlowsheetSimulationObject("Acetone Column")
4
5 # define the list of column pressures in bar
6
7 Plevels = [8, 9, 10, 11, 12]
8
9 # setup spreadsheet table headers
10
11 Spreadsheet.Worksheets[0].Cells["A1"].Data = "Stage"
12 Spreadsheet.Worksheets[0].Cells["B1"].Data = "P = 8 bar"
13 Spreadsheet.Worksheets[0].Cells["C1"].Data = "P = 9 bar"
14 Spreadsheet.Worksheets[0].Cells["D1"].Data = "P = 10 bar"
15 Spreadsheet.Worksheets[0].Cells["E1"].Data = "P = 11 bar"
16 Spreadsheet.Worksheets[0].Cells["F1"].Data = "P = 12 bar"
17
18 # add column of stage numbers
19
20 j = 1
21
22 for stage in column.Stages:
23
24     Spreadsheet.Worksheets[0].Cells[j, 0].Data = j
25
26     j += 1
27
28 # loop through the pressure values, set them and run the simulation, collecting the results
29
30 i = 1
31
32 for Pnew in Plevels:
33
34     for stage in column.Stages:
35
36         stage.P = Pnew * 100000 # set new stage pressures in Pa
37
38 Flowsheet.SolveFlowsheet2() # request a flowsheet calculation
39
40 j = 1
41
42 for t in column.Tf: # column.Tf is the vector of final stage temperatures in K
43
44     Spreadsheet.Worksheets[0].Cells[j, i].Data = t # write the temperature values in the corresponding column
45
46     j += 1
47
48 i += 1

```

Figure 142: Python Script

3. Run the script asynchronously (this prevents DWSIM from freezing until the calculation finishes):



Figure 143: Run Python Script (Async)

4. Go to the **Spreadsheet**, select the entire data range, click with the right mouse button and select **Create 2D XY Chart from Selection**.

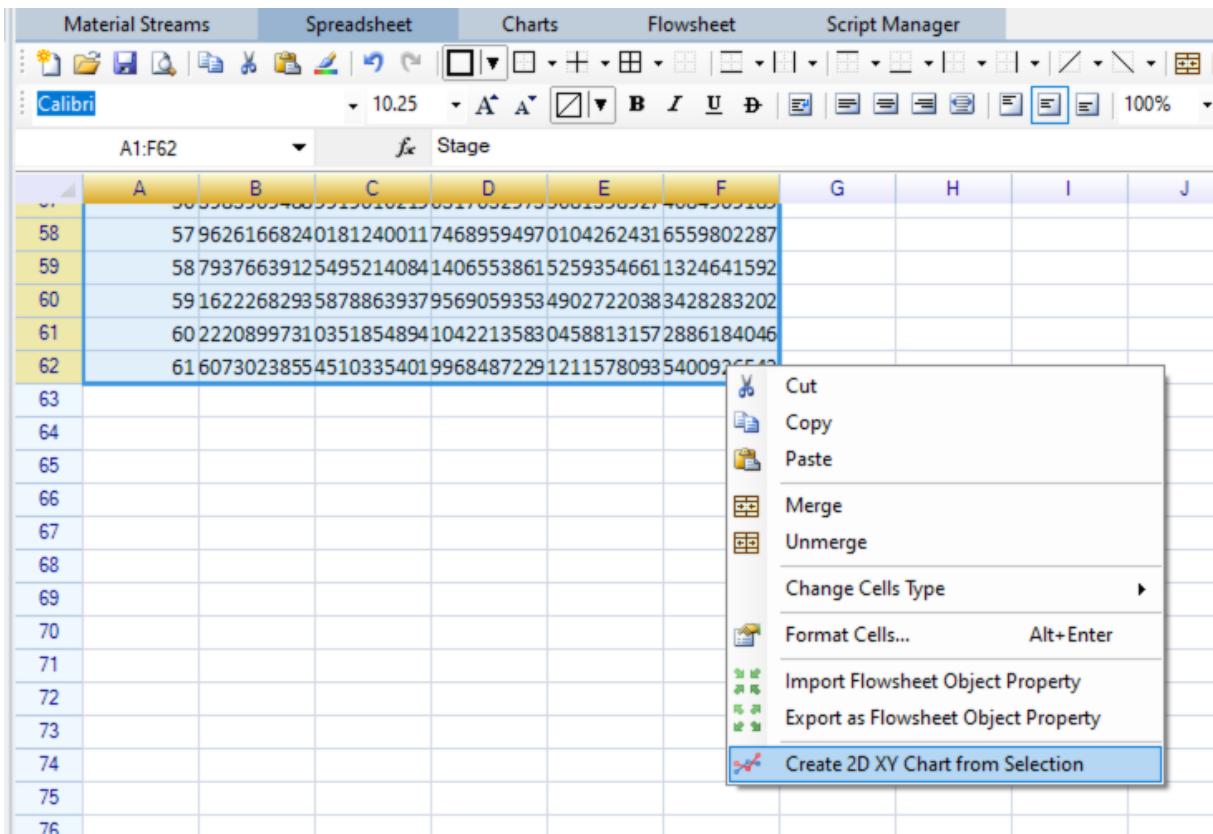


Figure 144: Create new chart from selected spreadsheet data range

5. View and configure the newly created chart as in the following picture:

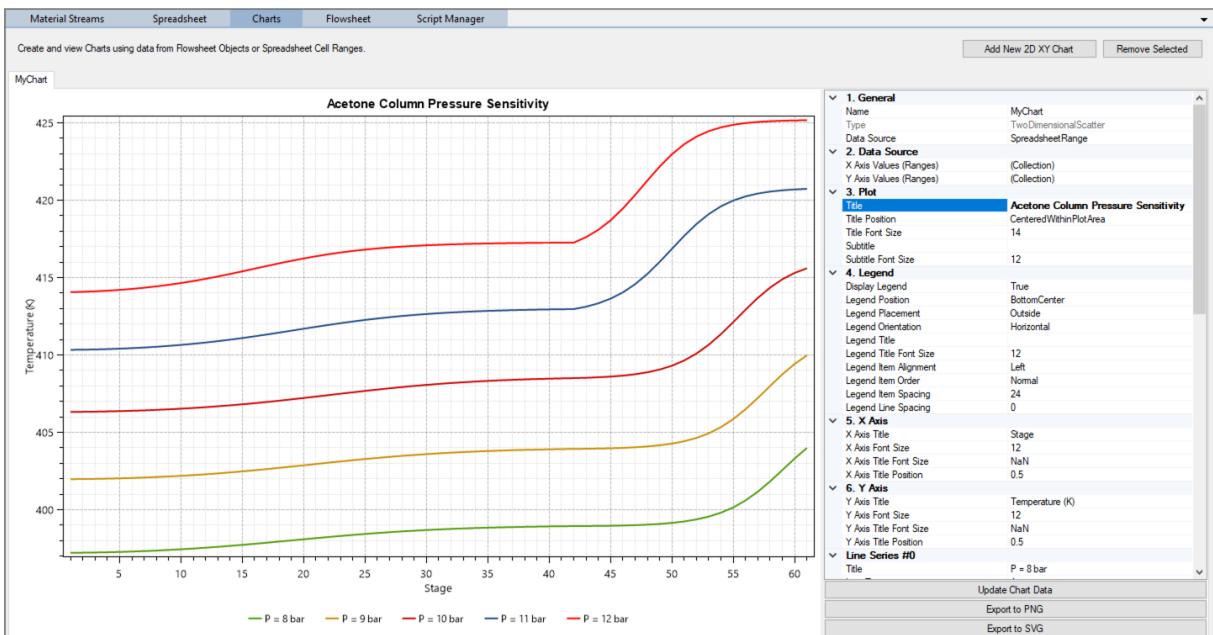


Figure 145: Pressure-Temperature dependence of the Acetone Column

6. Analyze the results obtained and discuss them with your colleagues.

### 27.3. DWSIM Model (Cross-Platform UI)

1. Save the previous simulation with a different file name and remove everything from the flowsheet except the objects depicted on the following picture:

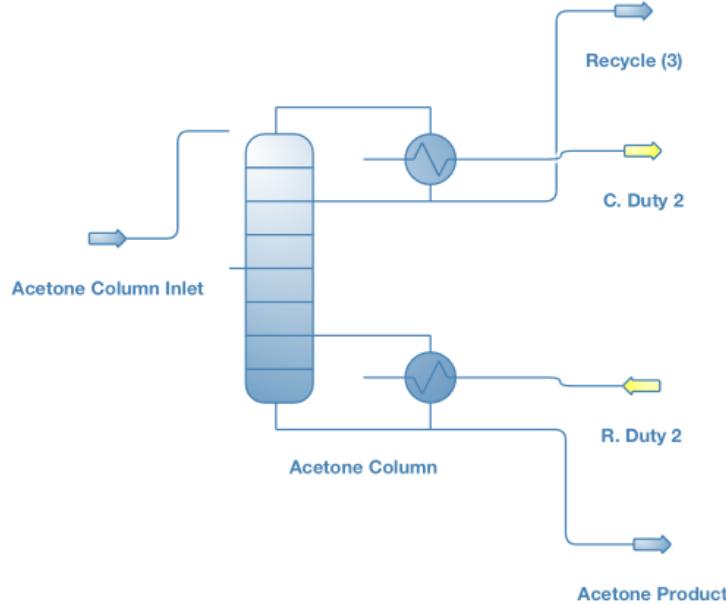


Figure 146: Process Flowsheet Diagram

2. Go to the **Script Manager**, create a New Script, select it on the list and enter the following code:

```

1 # Get Acetone Column object reference from the Flowsheet
2
3 column = Flowsheet.GetFlowsheetSimulationObject("Acetone Column")
4
5 # define the list of column pressures in bar
6
7 Plevels = [8, 9, 10, 11, 12]
8
9 # setup spreadsheet table headers
10
11 Spreadsheet.Worksheets[0].Cells["A1"].Data = "Stage"
12 Spreadsheet.Worksheets[0].Cells["B1"].Data = "P = 8 bar"
13 Spreadsheet.Worksheets[0].Cells["C1"].Data = "P = 9 bar"
14 Spreadsheet.Worksheets[0].Cells["D1"].Data = "P = 10 bar"
15 Spreadsheet.Worksheets[0].Cells["E1"].Data = "P = 11 bar"
16 Spreadsheet.Worksheets[0].Cells["F1"].Data = "P = 12 bar"
17
18 # add column of stage numbers
19
20 j = 1
21
22 for stage in column.Stages:
23
24     Spreadsheet.Worksheets[0].Cells[j, 0].Data = j
25
26     j += 1
27
28 # loop through the pressure values, set them and run the simulation, collecting the results
29
30 i = 1
31
32 for Pnew in Plevels:
33
34     for stage in column.Stages:
35
36         stage.P = Pnew * 100000 # set new stage pressures in Pa
37
38 Flowsheet.SolveFlowsheet2() # request a flowsheet calculation
39
40 j = 1
41
42 for t in column.Tf: # column.Tf is the vector of final stage temperatures in K
43
44     Spreadsheet.Worksheets[0].Cells[j, i].Data = t # write the temperature values in the corresponding column
45
46     j += 1
47
48 i += 1

```

Figure 147: Python Script

3. Run the script asynchronously (this prevents DWSIM from freezing until the calculation finishes):

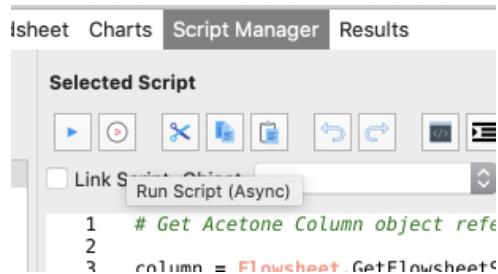


Figure 148: Run Python Script (Async)

4. Go to the **Spreadsheet**, select the entire data range, click with the right mouse button and select **Create Chart from Selected Range**.

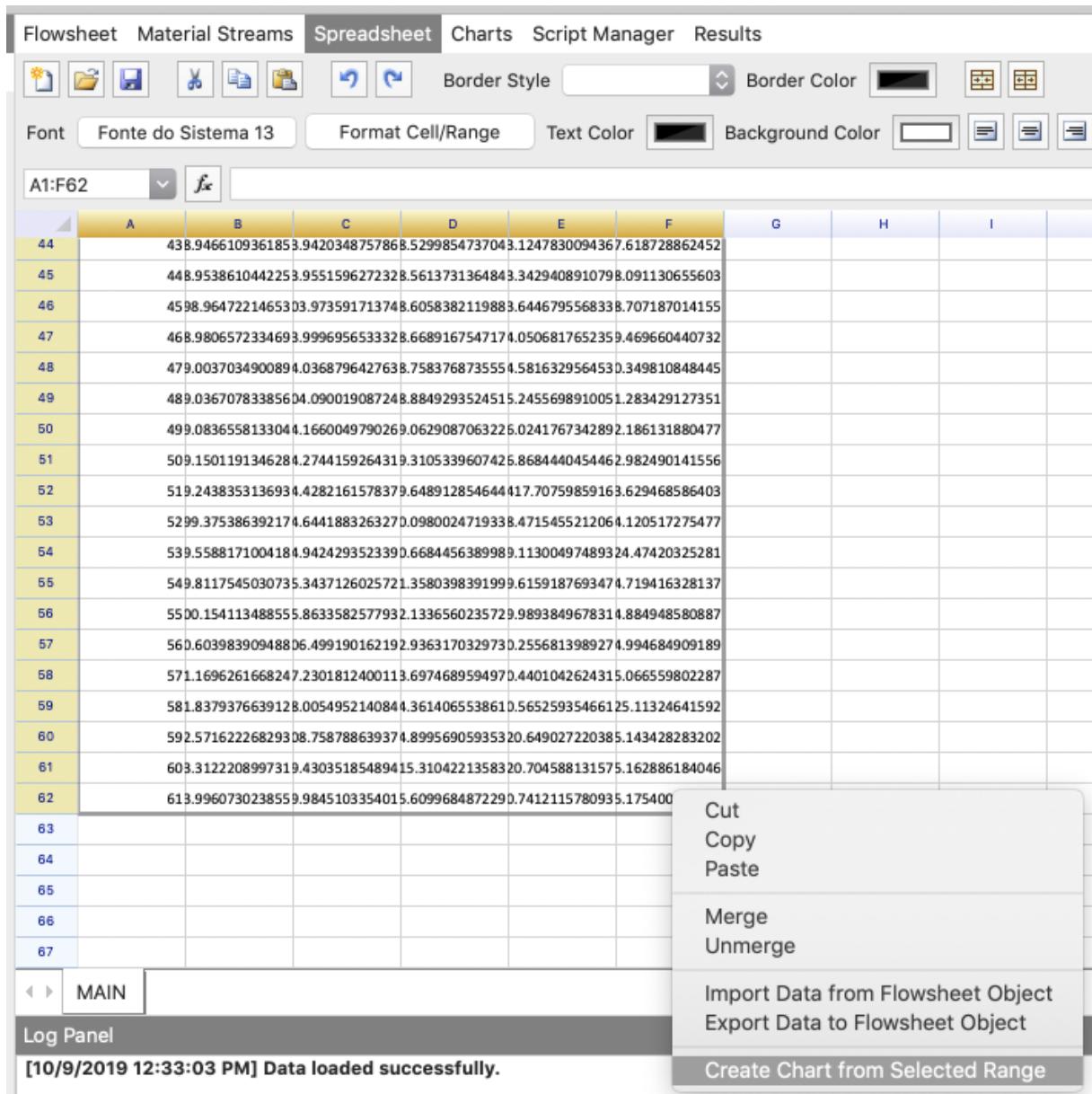


Figure 149: Create new chart from selected spreadsheet data range

5. View and configure the newly created chart as in the following picture:

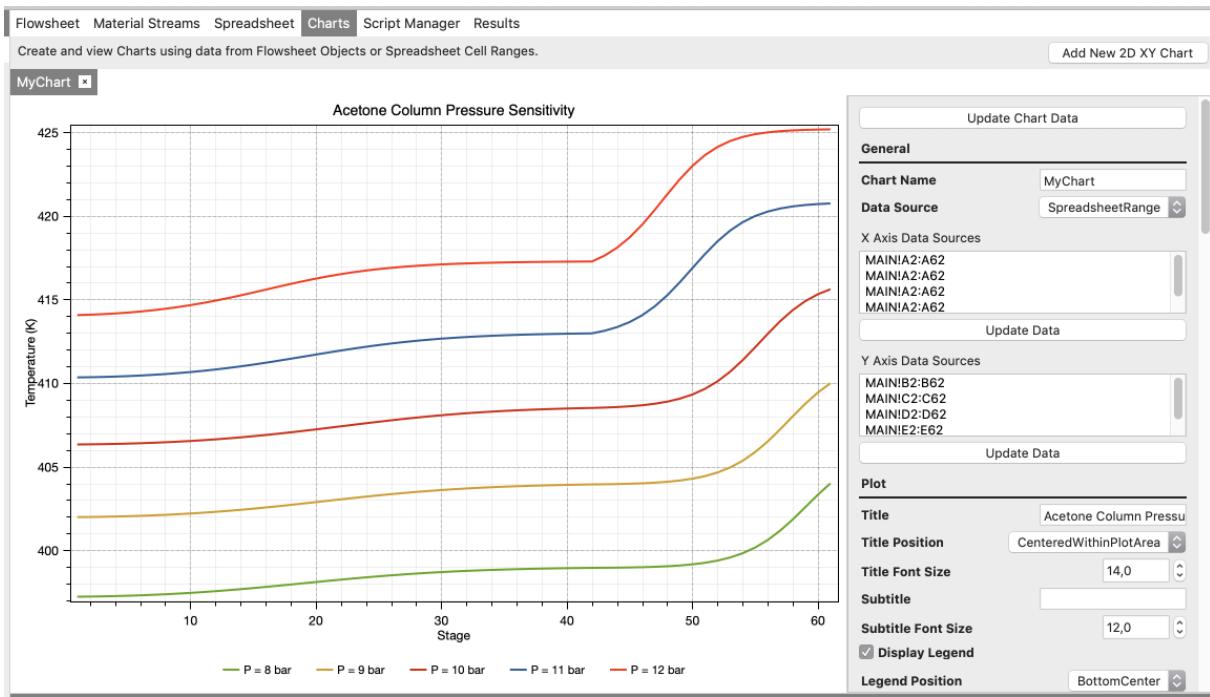


Figure 150: Pressure-Temperature dependence of the Acetone Column

6. Analyze the results obtained and discuss them with your colleagues.

## 28. Basic Dynamic Simulation Tutorial

### 28.1. Introduction

In this tutorial, we will learn how to do a dynamic simulation of a water storage tank, adding a PID Controller to keep the liquid level inside the tank around a desired value.

### 28.2. DWSIM Model (Classic UI)

Create and configure a new simulation. Add Water as the only compound and use the Steam Tables Property Package.

#### 28.2.1. Model Building

1. Build your model as in the following picture:

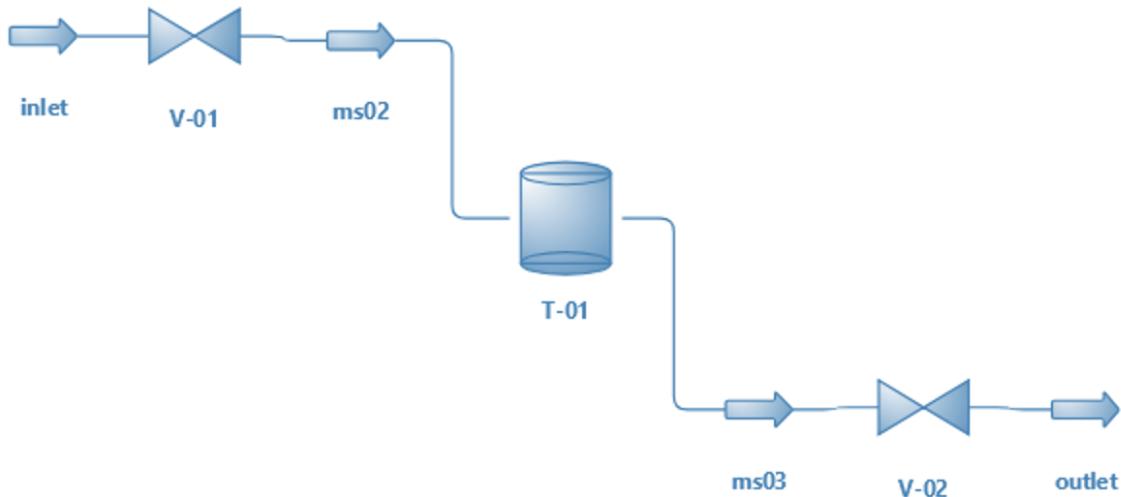


Figure 151: Water Tank model

2. Enable/Activate **Dynamic Mode**.
3. Set the inlet stream Pressure to 130000 Pa and Mass Flow to 10 kg/s. Set its Dynamic Spec to 'Flow'.

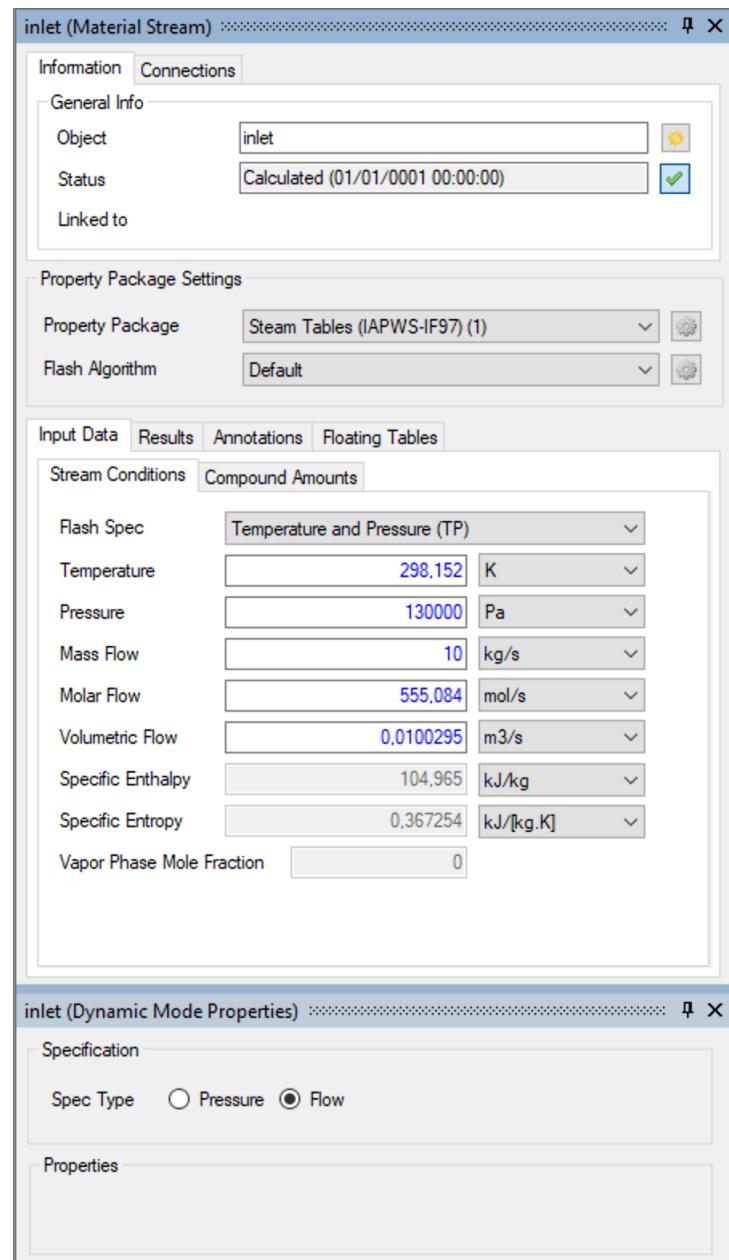


Figure 152: Inlet stream properties.

4. Set V-01 calculation type to Liquid Kv, Kvmax to 100, check the Opening vs Kv/Kvmax box and set the valve opening to 50%.
5. Set T-01 volume to 2 m<sup>3</sup> and available liquid height to 2 m. Set the "Reset Content" property to 1.
6. Set V-02 calculation type to Liquid Kv, Kvmax to 400, check the Opening vs Kv/Kvmax box and set the valve opening to 50%.
7. Set the outlet stream Pressure to 101325 Pa. Set its Dynamic Spec to 'Pressure'.

With the above settings, the flow rate of water entering the tank will be fixed at 10 kg/s. The dynamic model for the Tank considers the liquid height contribution (static pressure) for the pressure of the tank's outlet stream. Since V-02's outlet stream pressure is fixed, the actual outlet flow will be calculated by the

valve using the current opening and the connected stream pressures. As a result, the liquid level inside the tank will vary according to the difference between inlet and outlet flow rates.

### 28.2.2. Dynamic Simulation

1. Add a Level Gauge and associate it with the Tank's Liquid Level property. Set its maximum value to 3 m.
2. Save the current flowsheet state as **NewState1**.
3. Go to the **Dynamics Manager** and create a new Integrator (Int1) with Integration Step equal to 5 seconds and Duration equal to 10 minutes. Add the Tank's liquid level and the openings of the two valves as monitored variables for this integrator.
4. Create a new Schedule (Sch1) and associate the previously created Integrator and Flowsheet State with it.
5. Open the Integrator Controls Panel and run the dynamic simulation. The liquid level on the tank should stabilize around 0.33 meters after 7 minutes.
6. On the Integrator Controls Panel, click on the **View Results** button. On the created Spreadsheet, select the A and B columns and create a new chart from the selection. View the generated chart.

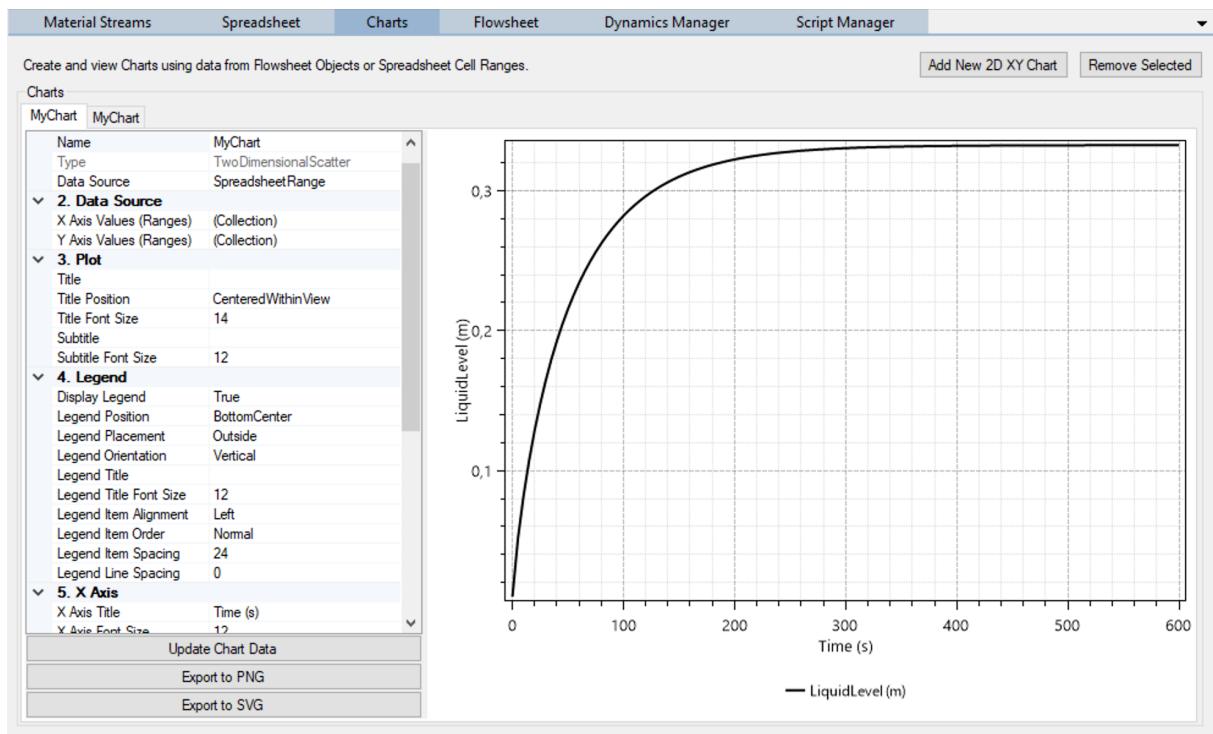


Figure 153: Liquid Level versus time.

### 28.2.3. Adding a PID Controller

1. Add a PID Controller to the flowsheet and set the V-02's opening as the manipulated variable and the Tank's liquid level as the controlled one. The set-point should be equal to 1 (m). Set Kp to 100.

2. Add a new Chart to the flowsheet and associate it with the PID's History item.
3. Open the Integrator Controls Panel and run the dynamic simulation. The liquid level on the tank should stabilize around 1.2 meters after 3 minutes.
4. Now set the controller's Ki to 10 and rerun the dynamic simulation. The liquid level on the tank should stabilize around 0.95 meters.

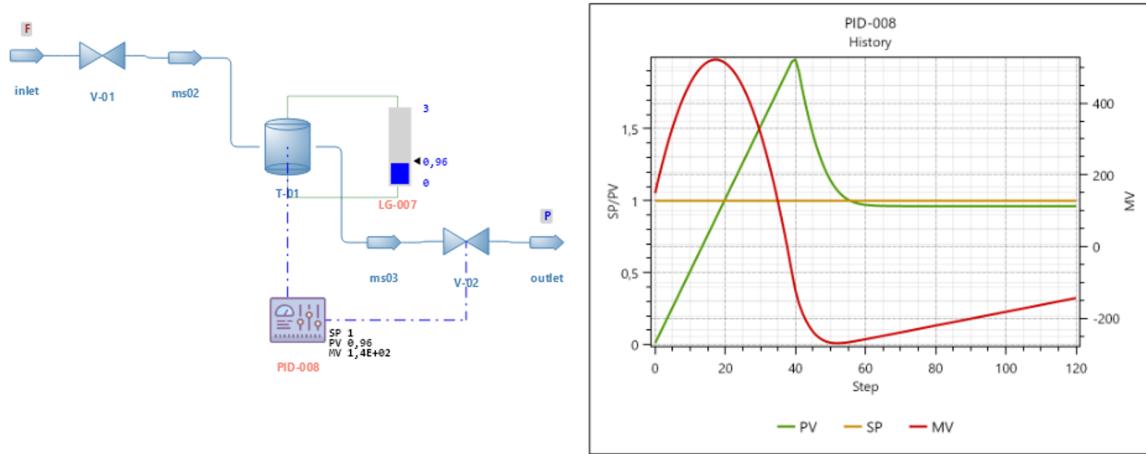


Figure 154: Liquid Level versus time with controller engaged.

#### 28.2.4. PID Controller Tuning

1. Open the PID Controller Tuning Tool, select the controller added to the simulation and run the tuning.

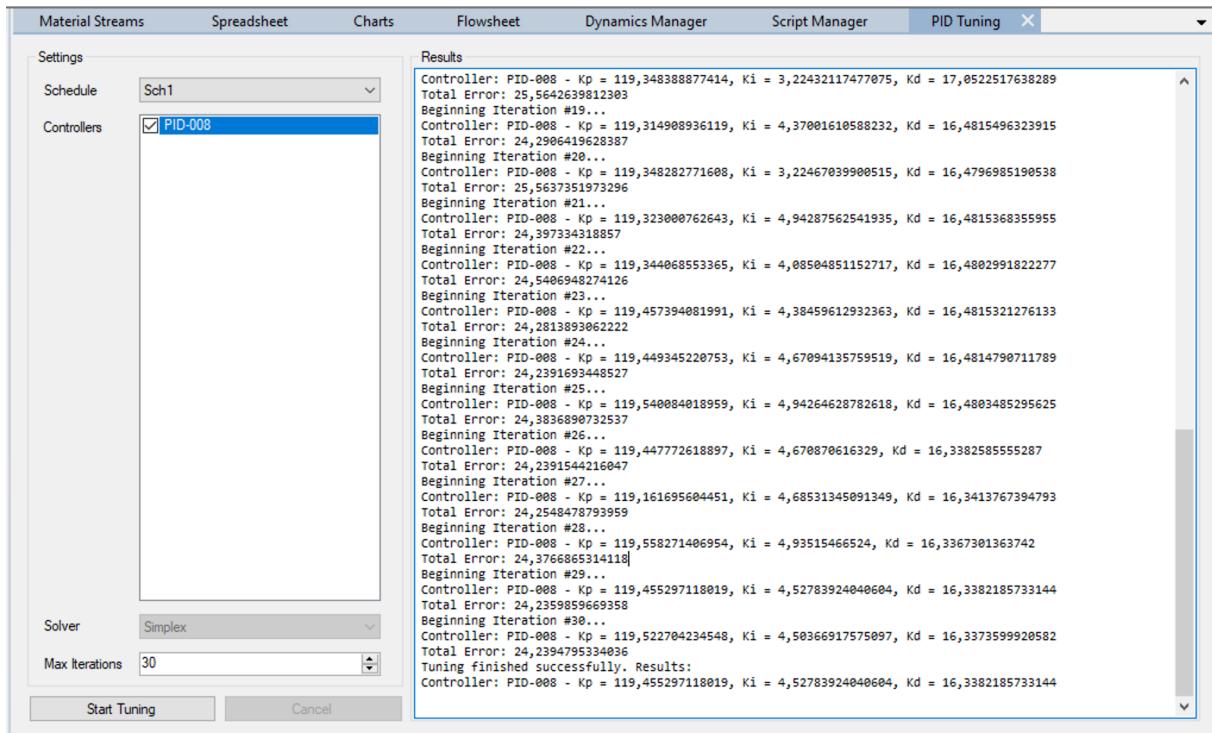


Figure 155: PID Controller Tuning Tool.

2. Open the Integrator Controls Panel and run the dynamic simulation. The liquid level on the tank should stabilize around 1 meter after 5 minutes. Notice that there is still a very high overshoot on the liquid level, even after the PID tuning. Perhaps you can try tuning it again with different initial values for Kp, Ki and Kd and/or increase the number of optimizer runs.

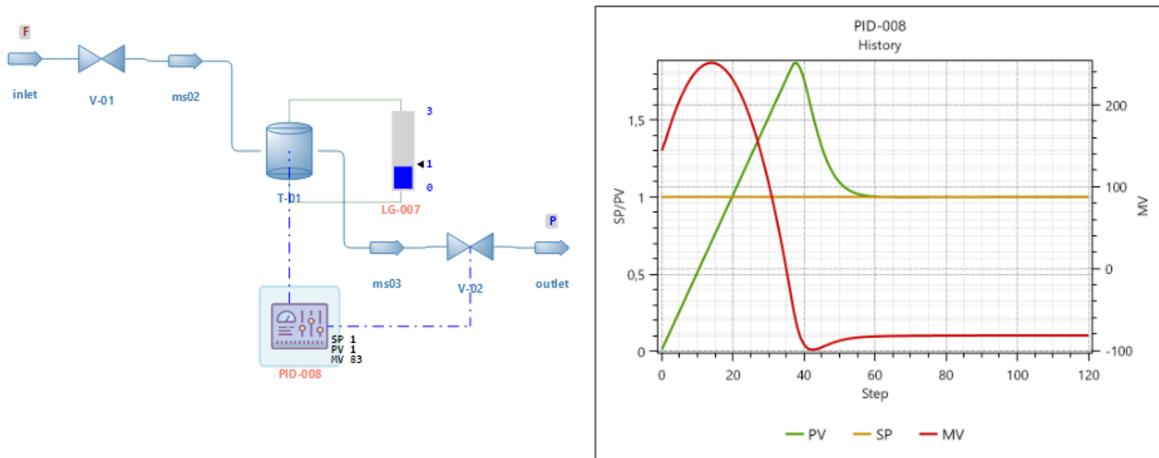


Figure 156: Liquid Level versus time with controller engaged.

### 28.3. Real-Time Mode

1. Change the flowsheet to **Control Panel Mode**. It should become dark and read-only, i.e. you cannot drag and/or add new objects.

2. Run the dynamic simulation in real-time mode by clicking on the 'clock' button on the Integrator Controls Panel.
3. After some time, click on the PID Controller and set the SP value to 1.7 (m). Watch how the system reacts to this change.

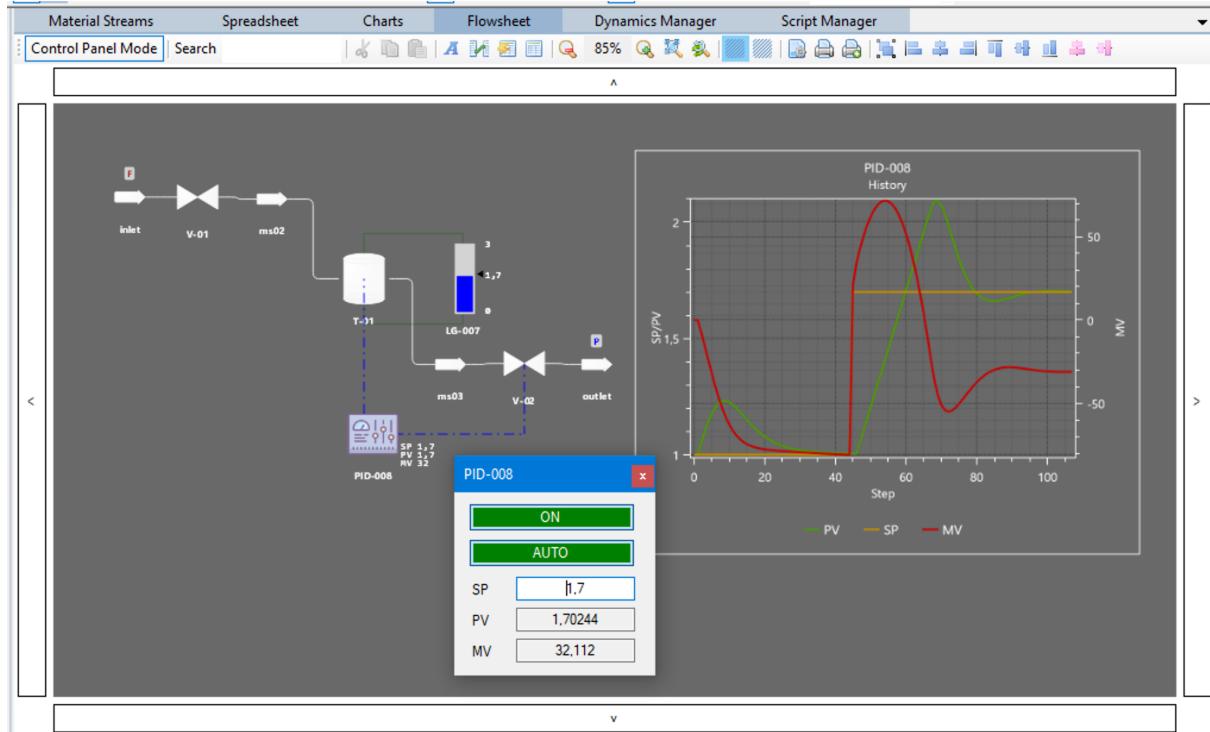


Figure 157: Control Panel mode with changes to PID parameters.

4. Remember that, after each integrator run, you can click on **View Results** and inspect the values of the monitored variables on that run.

## List of Figures

1 Classic UI on Windows 10 + .NET Framework 4.8.	6
2 Classic UI on Ubuntu Linux 16.04 + Mono 5.10.	7
3 Cross-Platform UI on Linux 16.04 + Mono 5.10 (GTK backend).	8
4 Cross-Platform UI on Windows 10 (WPF backend).	8
5 Cross-Platform UI on macOS Mojave (Cocoa backend).	9
6 DWSIM's welcome screen.	10
7 Simulation Configuration Wizard.	11
8 Simulation Configuration window.	12
9 Compound Data Importing tools.	13
10 Property Package configuration interface.	14
11 Property package configuration window (1).	15
12 Property Package configuration interface.	16
13 System of Units configuration interface.	18
14 Misc settings interface.	19
15 Property Tables settings interface.	20
16 Selected properties on the previous image are shown on the flowsheet for the Material Streams.	20
17 DWSIM simulation window.	21
18 Window repositioning.	22
19 Inserting an object to the flowsheet.	23
20 Inserting an object to the flowsheet by dragging from the Object Pallette window.	23
21 A material stream in the flowsheet.	26
22 Selected object context menu.	26
23 Connection selection menu.	27
24 Create and Connect tool.	27
25 Expander with all connections correctly configured.	28
26 Viewing object properties in the editor window.	29
27 Direct editing of a property.	29
28 Converting 50 C to the current temperature units (K).	30
29 Converted temperature value (323.15 K).	30
30 Calculated objects.	30
31 Undo action list.	31
32 Edit menu.	32
33 DWSIM's calculator control bar.	33
34 A DWSIM's calculator message.	33
35 Results report configuration.	33
36 Results report.	34
37 Sensitivity Analysis Utility (1).	35
38 Multivariate Optimization Utility (1).	36
39 Multivariate Optimization Utility (2).	37
40 Multivariate Optimization Utility (3).	38
41 Attaching Utilities through the "Add Utility" window.	39

42 Attaching Utilities through the object editors. . . . .	39
43 Accessing attached Utilities. . . . .	39
44 Utilities - True Critical Point. . . . .	40
45 Utilities - Hydrate Calculations. . . . .	40
46 Utilities - Phase Envelope. . . . .	41
47 Utilities - Binary Envelope. . . . .	41
48 Utilities - Petroleum Cold Flow Properties. . . . .	42
49 Chemical Reactions Manager. . . . .	43
50 Reaction Set editor. . . . .	44
51 C7+ petroleum fraction characterization utility. . . . .	45
52 Characterizing petroleum from distillation curves. . . . .	46
53 General Settings section. . . . .	47
54 DWSIM Launcher. . . . .	50
55 DWSIM Main Flowsheet Window with Simulation Setup Wizard. . . . .	51
56 Selecting a Compound with the Simulation Setup Wizard. . . . .	52
57 Selecting a Property Package with the Simulation Setup Wizard. . . . .	53
58 Viewing Property Packages on the Simulation Basis panel. . . . .	54
59 Editing Property Package Settings. . . . .	55
60 Viewing the Systems of Units on the Flowsheet Settings Panel. . . . .	57
61 Creating a new System of Units. . . . .	58
62 Add New Simulation Object. . . . .	59
63 Dragging Objects from the Object Palette to the Flowsheet PFD. . . . .	59
64 Editing the Connections of an Object. . . . .	61
65 Entering process data. . . . .	62
66 The Flowsheet Solver doing its work. . . . .	63
67 The Results Panel. . . . .	63
68 Viewing results from a single object. . . . .	64
69 Pure Compound Property Viewer. . . . .	65
70 Phase Envelope Utility. . . . .	66
71 Binary Envelope Utility. . . . .	67
72 Example of a valid flowsheet for dynamic mode. Both boundary streams are connected to valves. The inlet stream has a Flow specification while the outlet stream is Pressure-specified. . . . .	90
73 A block diagram of a PID controller in a feedback loop. $r(t)$ is the desired process value or setpoint (SP), and $y(t)$ is the measured process value (PV). (source: wikipedia) . . . . .	92
74 PID manual tuning summary. . . . .	95
75 Ziegler-Nichols tuning. . . . .	95
76 PID Controller Parameters. . . . .	96
77 PID Editor in Control Panel (Real-Time) mode. . . . .	96
78 A Flowsheet running in Control Panel (real-time) dynamic mode. . . . .	97
79 PID Controller Tuning Tool . . . . .	98
80 Unsupported Unit Operation in Dynamic Mode. . . . .	99

81 Compound Selection . . . . .	144
82 Property Package Selection . . . . .	144
83 New System of Units . . . . .	145
84 Overall Reaction setup . . . . .	145
85 Steam Reforming Reaction setup . . . . .	146
86 Water Gas Shift Reaction setup . . . . .	146
87 PFD setup . . . . .	147
88 Inlet Stream setup . . . . .	147
89 PFR setup . . . . .	147
90 Final Methane conversion . . . . .	148
91 Sensitivity Analysis case setup . . . . .	148
92 Analysis results . . . . .	149
93 Create Chart from Spreadsheet Range . . . . .	149
94 Created Chart . . . . .	150
95 PFR concentration profile . . . . .	150
96 Compound Selection . . . . .	151
97 Property Package Selection . . . . .	152
98 New System of Units . . . . .	152
99 Overall Reaction setup . . . . .	153
100 Steam Reforming Reaction setup . . . . .	153
101 Water Gas Shift Reaction setup . . . . .	154
102 PFD setup . . . . .	154
103 Inlet Stream setup . . . . .	155
104 PFR setup . . . . .	155
105 PFR calculation results . . . . .	156
106 Sensitivity Analysis case setup . . . . .	157
107 Sensitivity Analysis results . . . . .	157
108 PFR concentration profile as visible on the Charts tool . . . . .	158
109 Process Flowsheet . . . . .	159
110 Compound Selection . . . . .	160
111 Property Package and Flash Algorithm Selection . . . . .	160
112 NRTL Interaction Parameters for Methanol/Acetone . . . . .	161
113 New System of Units . . . . .	161
114 Process Flowsheet Diagram . . . . .	162
115 Enable/Disable Flowsheet Calculator/Solver . . . . .	162
116 Methanol Column configuration . . . . .	163
117 Acetone Column configuration . . . . .	164
118 Acetone Column initial estimates for temperature profile . . . . .	164
119 Pump and Valve properties . . . . .	165
120 Methanol Inlet Stream configuration . . . . .	165
121 Methanol Recycle Stream configuration . . . . .	166
122 Associating the Inside-Out flash with the streams . . . . .	166

123 Inserting a Property Table . . . . .	166
124 Setting up a Property Table . . . . .	167
125 Final results . . . . .	167
126 Compound Selection . . . . .	168
127 Property Package and Flash Algorithm Selection . . . . .	169
128 NRTL Interaction Parameters for Methanol/Acetone . . . . .	169
129 New System of Units . . . . .	170
130 Process Flowsheet Diagram . . . . .	170
131 Methanol Column configuration . . . . .	171
132 Acetone Column configuration . . . . .	172
133 Acetone Column initial estimates for temperature profile . . . . .	173
134 Pump and Valve properties . . . . .	174
135 Methanol Inlet Stream configuration . . . . .	175
136 Methanol Recycle Stream configuration . . . . .	176
137 Associating the Inside-Out flash with the streams . . . . .	177
138 Inserting a Property Table . . . . .	177
139 Setting up a Property Table . . . . .	177
140 Final results . . . . .	178
141 Process Flowsheet Diagram . . . . .	179
142 Python Script . . . . .	180
143 Run Python Script (Async) . . . . .	180
144 Create new chart from selected spreadsheet data range . . . . .	181
145 Pressure-Temperature dependence of the Acetone Column . . . . .	181
146 Process Flowsheet Diagram . . . . .	182
147 Python Script . . . . .	183
148 Run Python Script (Async) . . . . .	183
149 Create new chart from selected spreadsheet data range . . . . .	184
150 Pressure-Temperature dependence of the Acetone Column . . . . .	185
151 Water Tank model . . . . .	186
152 Inlet stream properties. . . . .	187
153 Liquid Level versus time. . . . .	188
154 Liquid Level versus time with controller engaged. . . . .	189
155 PID Controller Tuning Tool. . . . .	190
156 Liquid Level versus time with controller engaged. . . . .	190
157 Control Panel mode with changes to PID parameters. . . . .	191

## **List of Tables**

1 Column solving methods' configuration parameters. . . . .	79
2 Enthalpy/Entropy calculation with an EOS . . . . .	114
4 Constants for the Lee-Kesler model . . . . .	116
5 Hypo calculation methods. . . . .	135

## References

- [1] H. Z. Kister, *Distillation Design*, McGraw-Hill Professional, 1992.
- [2] J. D. Seader, E. J. Henley, *Separation Process Principles*, Wiley, 2005.
- [3] T. Tinker, *Trans. ASME*, 1958, **36**, 80.
- [4] W. McCabe, J. Smith, P. Harriott, *Unit Operations of Chemical Engineering, of Chemical Engineering Series*, McGraw-Hill Education, 2005.
- [5] M. Michelsen, J. Mollerup, *Thermodynamic Models: Fundamentals and Computational Aspects*, Tie-Line Publications, Denmark, 2007.
- [6] J. Smith, *Intro to Chemical Engineering Thermodynamics*, McGraw-Hill Companies, New York, 1996.
- [7] D.-Y. Peng, D. B. Robinson, *Industrial and Engineering Chemistry Fundamentals*, 1976, **15**, 59–64.
- [8] S. Vitu, J.-N. Jaubert, F. Mutelet, *Fluid Phase Equilibria*, 2006, **243**, 9 – 28.
- [9] G. Soave, *Chemical Engineering Science*, 1972, **27**, 1197–1203.
- [10] C. H. Whitson, M. R. Brule, *Phase Behavior (SPE Monograph Series Vol. 20)*, S. of Petroleum Engineers (Ed.), Society of Petroleum Engineers, 2000.
- [11] R. Stryjek, J. H. Vera, *The Canadian Journal of Chemical Engineering*, 1986, **64**, 820–826.
- [12] K. C. Chao, J. D. Seader, *AICHE Journal*, 1961, **7**, 599–605.
- [13] H. G. Grayson, C. W. Streed, *N/A*, 1963, **VII**.
- [14] J. W. Kang, V. Diky, M. Frenkel, *Fluid Phase Equilibria*, 2015, **388**, 128 – 141.
- [15] H. Renon, J. M. Prausnitz, *AICHE Journal*, 1968, **14**, 135–144.
- [16] A. Mohs, J. Gmehling, *Fluid Phase Equilibria*, 2013, **337**, 311 – 322.
- [17] K. Thomsen, Ph.D. thesis, Department of Chemical Engineering, Technical University of Denmark, 1997.
- [18] R. Reid, *The Properties of Gases and Liquids*, McGraw-Hill, New York, 1987.
- [19] B. I. Lee, M. G. Kesler, *AIChE Journal*, 1975, **21**, 510–527.
- [20] I. H. Bell, J. Wronski, S. Quoilin, V. Lemort, *Industrial & Engineering Chemistry Research*, 2014, **53**, 2498–2508.
- [21] M. R. Riazi, H. A. Al-Adwani, A. Bishara, *Journal of Petroleum Science and Engineering*, 2004, **42**, 195–207.
- [22] M. R. Riazi, *Characterization and properties of petroleum fractions*, ASTM, 2005.
- [23] B. K. Harrison, W. H. Seaton, *Industrial and Engineering Chemistry Research*, 1988, **27**, 1536–1540.

- [24] J. Marrero, R. Gani, *Fluid Phase Equilibria*, 2001, **183-184**, 183–208.
- [25] R. A. Heidemann, A. M. Khalil, *AIChE Journal*, 1980, **26**, 769–779.
- [26] W. R. Parrish, J. M. Prausnitz, *Industrial and Engineering Chemistry Process Design and Development*, 1972, **11**, 26–35.
- [27] J. Klauda, S. Sandler, *Industrial and Engineering Chemistry Research*, 2000, **39**, 3377–3386.
- [28] G.-J. Chen, T.-M. Guo, *Chemical Engineering Journal*, 1998, **71**, 145–151, url.