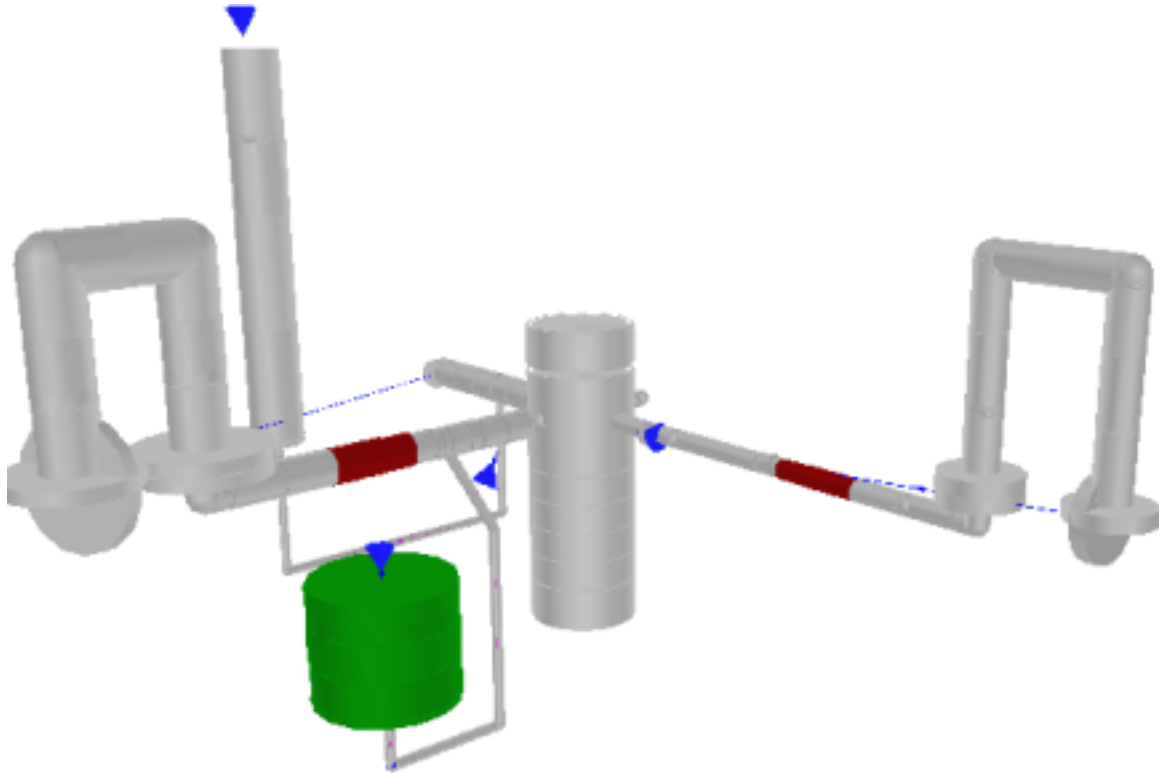


TRACE Plug-in Users Manual

Symbolic Nuclear Analysis Package (SNAP)



Version 4.3.0 - August 02, 2021

Applied Programming Technology, Inc.

**240 Market St., Suite 301
Bloomsburg PA 17815-1951**

TRACE Plug-in Users Manual

Applied Programming Technology, Inc.

by Ken Jones, Bill Dunsford, John Rothe, and Don Ulshafer

Copyright © 2007-2020

***** Disclaimer of Liability Notice *****

The Nuclear Regulatory Commission and Applied Programming Technology, Inc. provide no express warranties and/or guarantees and further disclaims all other warranties of any kind whether statutory, written, oral, or implied as to the quality, character, or description of products and services, its merchantability, or its fitness for any use or purpose. Further, no warranties are given that products and services shall be error free or that they shall operate on specific hardware configurations. In no event shall the US Nuclear Regulatory Commission or Applied Programming Technology, Inc. be liable, whether foreseeable or unforeseeable, for direct, incidental, indirect, special, or consequential damages, including but not limited to loss of use, loss of profit, loss of data, data being rendered inaccurate, liabilities or penalties incurred by any party, or losses sustained by third parties even if the Nuclear Regulatory Commission or Applied Programming Technology, Inc. have been advised of the possibilities of such damages or losses.

Table of Contents

1. Introduction	1
1.1. Installation Instructions	1
2. Creating a TRACE Model	3
2.1. Importing an Existing ASCII File	3
2.2. Importing from TPR	4
2.3. Creating a New TRACE Model	4
3. Editing A Model	5
3.1. TRACE Plug-in Preferences	6
3.1.1. Default Initial Conditions	6
4. Editing a Restart Case	7
5. Editing Model Properties	9
5.1. Model Options	10
6. Hydraulic Components	15
6.1. Break	15
6.1.1. Break Type	15
6.1.2. Break Table	16
6.1.3. Break Scale Factors	17
6.1.4. Break State Controllers	17
6.1.5. Break Contan Coupling	17
6.1.6. Break Trace Species	18
6.1.7. Break Control Connections	18
6.2. Channel	18
6.2.1. Channel General Properties	19
6.2.2. Channel Canister Properties	20
6.2.3. Channel Fuel Properties	20
6.2.4. Channel Rod Locations	21
6.2.5. Advanced Channel Features	22
6.3. Fill	23
6.3.1. Fill Type	23
6.3.2. Fill Table	24
6.3.3. Fill Contan Coupling	24
6.3.4. Fill Scale Factors	24
6.3.5. Fill State Controllers	25
6.3.6. Fill Control Connections	25
6.4. Feedwater Heater	26
6.4.1. Feedwater Heater Properties	26
6.4.2. Feedwater Heater Valve	26
6.5. Pipe	27
6.5.1. Pipe Properties	27
6.5.2. Pipe Geometry	28
6.5.3. Pipe Initial Conditions	29
6.5.4. Pipe Friction Data	31
6.5.5. Pipe Wall Heat Structure	32
6.5.6. Pipe Wall Power	33
6.5.7. Pipe Fluid Power	33
6.5.8. Pipe Leak Paths	34
6.6. Plenums	34
6.6.1. Plenum Junctions	35

6.7. Pumps	35
6.7.1. Pump General Properties	35
6.7.2. Pump Speed Values	36
6.7.3. Pump Initial Fluid Conditions	36
6.7.4. Pump Alternate Inertia	36
6.7.5. Pump Friction Factors	37
6.7.6. Pump Rated Values	37
6.7.7. Pump Homologous Curves	37
6.7.8. Pump Control Connections	38
6.8. Prizers	39
6.8.1. Prizer Properties	39
6.9. Separator	40
6.9.1. Separator Properties	40
6.10. Single Junctions	40
6.11. Tees	41
6.11.1. Tee Properties	41
6.12. Turbines	41
6.12.1. Turbine Properties	42
6.13. Valves	42
6.13.1. Valve properties	43
6.13.2. Valve Control Connections	43
6.14. Jet Pump	44
6.14.1. Jet Pump Properties	44
6.15. Vessels	45
6.15.1. Vessel Geometry	45
6.15.2. Volumetric and Edge Data	46
6.15.3. Vessel Boundary Interfaces	48
6.16. Vessel Junction Components	48
6.16.1. Initialization	49
6.16.2. Editing	52
6.16.3. Drawn Representation	53
7. Control Systems	57
7.1. Signal Variables	57
7.1.1. General Signal Variables	57
7.1.2. Component Signal Variable	57
7.1.3. Volume Signal Variable	58
7.1.4. Edge Signal Variables	58
7.1.5. Heat Signal Variables	59
7.1.6. Control Signal Variables	59
7.1.7. Power Signal Variables	60
7.2. Control Blocks	60
7.2.1. Arithmetic Blocks	60
7.2.2. Calculus Blocks	61
7.2.3. Controller Blocks	61
7.2.4. Logical Control Blocks	62
7.2.5. Manipulation Blocks	62
7.2.6. Time Blocks	63
7.2.7. Trigonometry Blocks	63
7.2.8. Non-Standard Control Blocks	64
7.3. Trips	64

7.3.1. Trip Input Source	65
7.3.2. Trip Signal Type	65
7.3.3. Trip Setpoint Data	65
7.3.4. Trip Signal Expressions	66
7.4. Control System Display Annotations	66
8. Thermal Components	67
8.1. Heat Structures	67
8.1.1. Creating a Heatstructure	67
8.1.2. Connecting A Heat Structure	68
8.1.3. Radial Geometry	69
8.1.4. Built-in Materials	71
8.2. User-Defined Materials	71
8.3. Radiation Enclosures	71
9. Power Components	73
9.1. Power	73
9.1.1. Power Properties	73
9.1.2. Power Shape	74
9.2. Fluid Power	75
9.2.1. Powered Components	76
10. Countercurrent Flow Limitation	77
11. Contain Components	79
12. Extern Components	81
13. ASCII I/O	83
14. Retrieve Initial Conditions	85
14.1. Managing Initial Conditions	86
15. Model Validation Tests	89
15.1. Loop Check	89
15.2. Non-Condensable Partial Pressure Test	89
15.3. Junction Flow Area Test	90
15.4. Abrupt Area Change Test	90
15.5. Edge Hydro-Diameter Test	90
15.6. PUMPFRICT Consistency Test	90
15.7. Vessel Connection Test	90
15.8. Adjacent Cell Volume Test	90
15.9. Channel Surface Test	91
16. Renodalization	93
16.1. 1D Hydraulic	93
16.1.1. Split	94
16.1.2. Split Uniform	95
16.1.3. Merge	95
16.1.4. Elevation Change	95
16.1.5. Announce Changes	96
16.1.6. Irreversible Loss Behavior	96
16.1.7. Renodalization Options Panel	96
16.2. 3D Hydraulic	98
16.2.1. Vessel Axial Renodalization	98
16.2.2. Vessel Radial Renodalization	100
16.2.3. Vessel Azimuthal Renodalization	100
16.3. Heat Structure	101
16.3.1. Radial Renodalization	102

16.3.2. Axial Renodalization	102
16.4. Reference Model	102
17. Loop Closure	105
18. Model Notebooks	107
19. Batch Commands	111
19.1. TRACE Variable Indexes	115
20. Resource File Import/Export	117
21. PARCS Mapping	119
21.1. Initial Map Creation	119
21.2. Mapping Configuration	121
21.3. TRACE-PARCS Coupled Jobstream	123
22. FRAPCON Fuel Rod Import	127
22.1. Single Fuel Rod Import & Creation	127
22.2. Multiple Fuel Rod Importing	129
22.2.1. FRAPCON Fuel Rod Job Stream	131
22.2.2. FRAPCON Fuel Rod Import Data	131
23. TRACE Parallel Task Sets	133
23.1. Getting Started	134
23.2. Parallel Task Dialog	135

Chapter 1. Introduction

The Symbolic Nuclear Analysis Package (SNAP) consists of a suite of integrated applications designed to simplify the process of performing thermal-hydraulic analysis. SNAP provides a highly flexible framework for creating and editing input for engineering analysis codes as well as extensive functionality for submitting, monitoring and interacting with the analysis codes. The modular plug-in design of the software allows functionality to be tailored to the specific requirements of each analysis code.

TRACE version 5.0 was developed by the NRC based off the TRAC-M code, which combined TRAC-B and TRAC-P. This manual describes the SNAP Plug-in features that are unique to the TRACE plug-in. Detailed documentation of the TRACE options are provided in the TRACE input manual which is accessible through SNAP. This manual should be considered a supplement to the TRACE input manual. Please refer to the TRACE User's Manual and the SNAP User's Manual for specific information on TRACE input requirements and SNAP general usage, respectively.

1.1. Installation Instructions

The installed SNAP plug-ins are loaded when the ModelEditor is opened or when the Calculation Server is started. These plug-ins are loaded from the **plugins** directory found in the directory where SNAP was installed. The SNAP installation directory defaults to C:\Program Files\snap\ under Microsoft Windows and /usr/local/snap/ under MAC, Linux and Unix systems. The TRACE plug-in is always included during installation.

Chapter 2. Creating a TRACE Model

A TRACE model can be created in the ModelEditor by either importing an existing ASCII model or by creating a new empty model. Once a TRACE model has been created in the ModelEditor, it can be saved to a file and later reopened using the File Open menu item.

2.1. Importing an Existing ASCII File

The SNAP Model Editor supports importing TRACE compliant ASCII input files. These files are commonly referred to as decks. ASCII files may be imported into the SNAP Model Editor from the **Import** sub menu in the **File** menu. If the **TRACE** option is not displayed, verify that the plug-in is properly installed.

This will open the file import dialog shown below in [Figure 2.1, “File Import Dialog”](#). Select **ASCII** as the file type on the right side of the dialog. The selected file will be imported into the Model Editor as a new TRACE model.

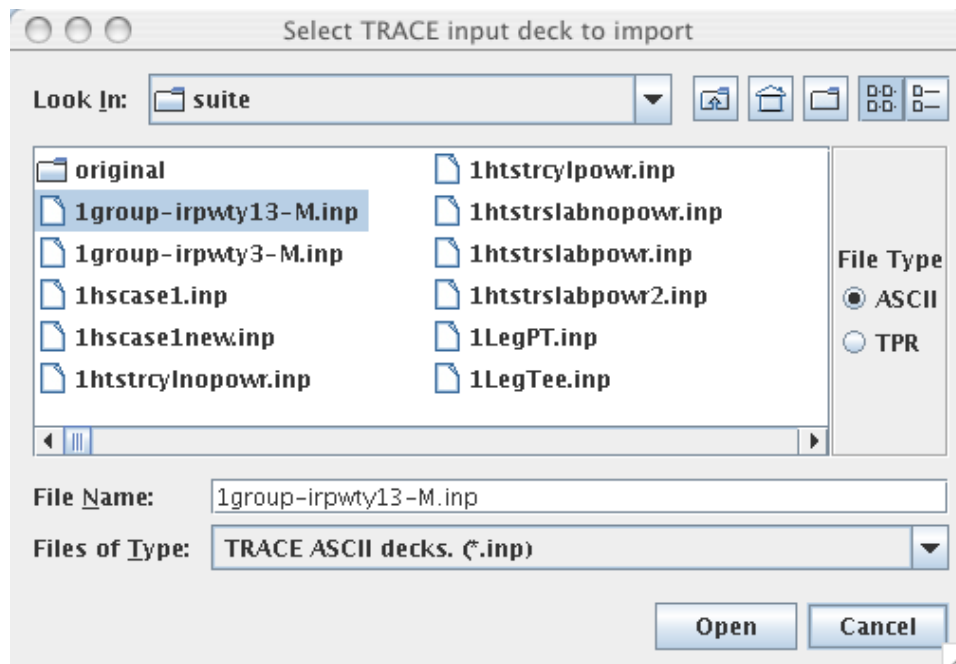


Figure 2.1. File Import Dialog

Comment lines that may exist in the deck are generally ignored during the import process. Decks exported by SNAP can include description and comment lines for any component. These lines are written to the deck using ***d:** and ***c:** as prefixes to identify description and comment data, respectively, as shown below:

```
*
*d: Simple vessel containing 7 axial levels
*c: This is where comments can be placed.
*c: This is line two of the comment.
*****
type          num      userid      component name
vessel         20        1          vessel
*      nasx      nrsx      ntsx      ncsr      ivssbf
*          7          1          1          1          0
*      idcu      idcl      idcr      icru      icrl
```

SNAP uses these prefixes to allow embedded comments and descriptions to be retained when importing a deck that was exported by the Model Editor. This mechanism can also be used to identify comments in legacy decks that should be retained on import.

2.2. Importing from TPR

The SNAP Model Editor can also import a TRACE model stored in a platform independent binary restart file (TPR) generated from TRACE. The import process is very similar to the ASCII import. In the file selection dialog displayed [Figure 2.1, “File Import Dialog”](#), selecting the TPR as the file type will allow the user to select a TPR file. Once the persistent data has been read in, the user must select a timestep for the state data. [Figure 2.2, “TPR Edit Selection Dialog”](#) displays an example timestep selection dialog. The state data includes all the transient data such as fluid state, heat structure temperatures and control block initial conditions.

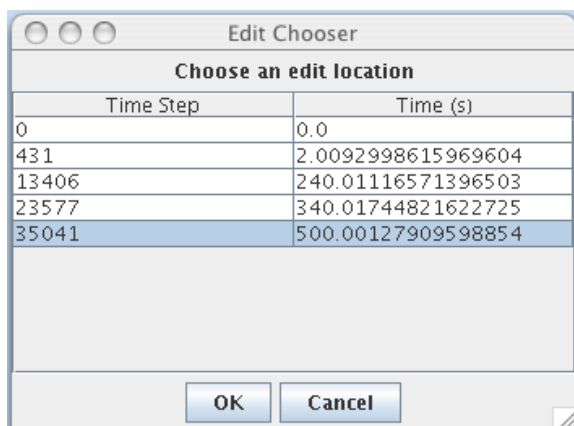


Figure 2.2. TPR Edit Selection Dialog

2.3. Creating a New TRACE Model

A TRACE model can be created by selecting TRACE from the list of available plug-ins in the New Model Dialog. This will create a blank model with a single open view. From here new components can be inserted, or pasted into the model.

Chapter 3. Editing A Model

Once a model is open in the ModelEditor (via importing ASCII, opening an existing MED file or creating a new model) it can be modified in a variety of ways. New components can be added to the model through the navigator shown below in [Figure 3.1, “Model Navigator”](#) or by using the insert tool in a 2D view. Existing components can be modified, connected, and disconnected through the property panel or the 2D view.

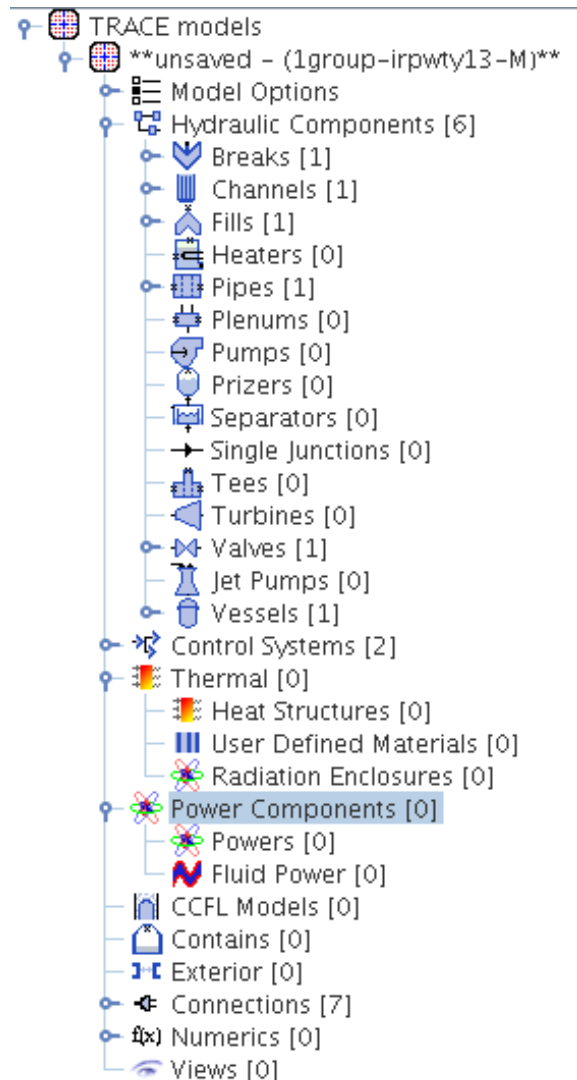


Figure 3.1. Model Navigator

[Figure 3.1, “Model Navigator”](#) shows each of the primary categories of components for a TRACE model in addition to the Connections, Numerics and View categories which are specific to the ModelEditor. The number contained in brackets to the right of a category name is the number of current elements. The following chapters are broken down in the same manner as in the Navigator, with components in alphabetical order inside their chapter.

3.1. TRACE Plug-in Preferences

Several TRACE Plug-in specific settings may be modified through the ModelEditor preferences dialog as shown in [Figure 3.2, “Plug-in Preferences”](#).

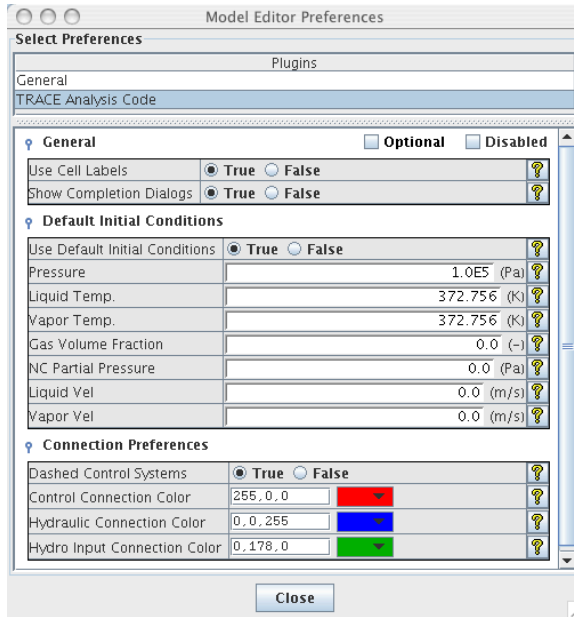


Figure 3.2. Plug-in Preferences

3.1.1. Default Initial Conditions

The default initial conditions are used for the state data in hydraulic components created inside the Model Editor. This is a way to easily build a new model that contains components with the same initial state. These values have no effect on components imported into the Model Editor or copied from an existing model.

Chapter 4. Editing a Restart Case

TRACE supports restarting finished calculations through the use of a restart deck. A restart deck contains a subset of the data in the original input deck. This includes all the namelist variables, user defined units and user defined materials. In addition to these properties that must be resupplied, the restart deck includes a fully re-supplied component entry for any component that has been modified. New components may be added and existing components may be removed from the model using a restart deck.

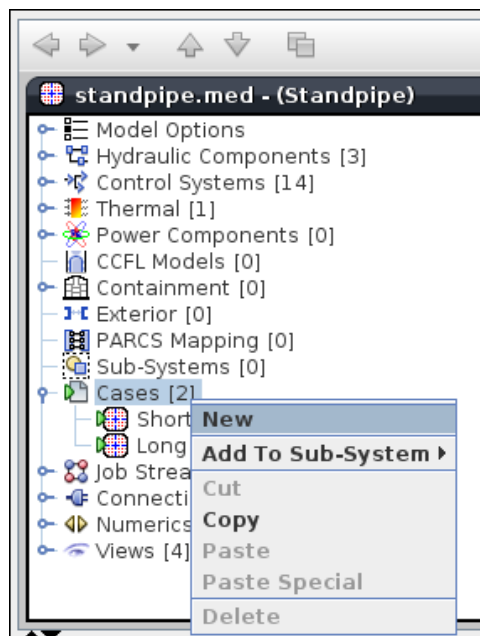


Figure 4.1. Creating a Restart Case

Restart decks are represented in the ModelEditor by "Restart Cases." Restart cases can be created by right-clicking on the *Cases* node in the Navigator and selecting the **New** pop-up menu item as shown below in [Figure 4.1, "Creating a Restart Case"](#). Once created, a restart case can be edited in either *ASCII* or *Graphical* mode (depending on the *Editing Mode* property) by pressing the **Edit** button for the *Restart Model* property. The **Show ASCII** pop-up menu item can also be used to view the contents of a case.

Editing a case graphically opens the case as a "Virtual Model" in the Navigator as shown in [Figure 4.2, "Graphical Restart Editing"](#). When graphically editing a restart case, a restart panel is shown above the Navigator tree. This panel indicates that restart case is being edited and provides shortcuts for both saving and closing the restart case. The ModelEditor tracks any changes made to the model in this mode and colors the modified components red in the Navigator. Once the changes have been made they can be saved back into the Restart Case as a restart input deck by pressing the **Save** button or discarded by pressing the **Close** button.

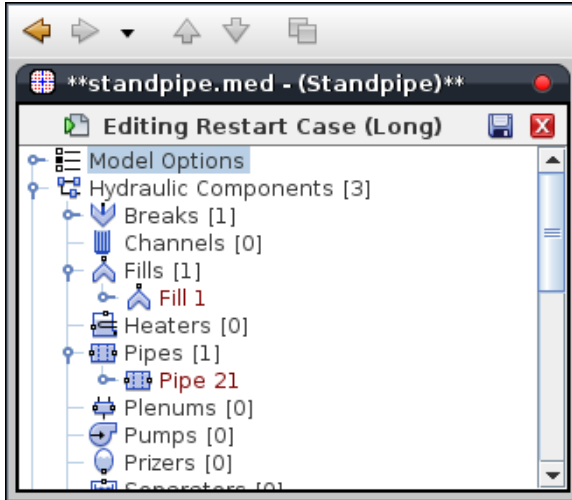


Figure 4.2. Graphical Restart Editing

Note Not all properties can be edited for a restart. These properties will be disabled when editing a restart.

The contents of a Restart Case may also be imported from or exported to a local file. To import a restart deck, select the **Import Case** item from the Case's right-click pop-up menu. Similarly, to export a restart deck, select the **Export Case** item from the Case's right-click pop-up menu.

Two additional Restart Case properties are used to assist the process of building restart models. The first is the *Initial Conditions* property which allows a previously retrieved and stored set of initial conditions to be selected for the case. This set of initial conditions will be applied to the case when it is opened for graphical editing but will not count as a change to the model. The second is the *Preserve Numerics* property which indicates that any user defined numeric that starts with a dollar(\$) should be included in the restart case by name instead of by numeric value.

Chapter 5. Editing Model Properties

Certain properties exist on a model level. These are high-level properties that affect the entire calculation or steer model boundary conditions. The TRACE plugin provides a series of editors for defining TRACE component properties. The Property View provides a number of generic inline editors for defining simple attributes. The TRUE FALSE editor is one example of a general inline editor because it is used to define nearly all logical values. Properties specified in the generic editors immediately update the ASCII view and add an undo entry to the undo stack.

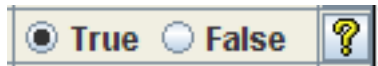


Figure 5.1. Generic Logical Editor

There are many other general case editors used when editing simple model properties. Custom editors are provided for editing properties in the situation where a property requires a more sophisticated method of specification. These editors provide the user with more intuitive means of editing and displaying complex properties. Many of the custom editors provided are tabular. Tabular editors allow multi-row editing, custom table cell editors, and column header tooltips which describe the data in each column.

Most of the custom tabular based TRACE property editors provide a pop-up menu containing a series of editing options. These options are listed below:

- Copy - The copy command copies data from either a spreadsheet or another tabular based editor.
- Paste - The paste command pastes copied data to either a spreadsheet or another tabular editor.
- Plot - The plot command provides a listing of available plot variables from the current tabular data. One independent and one or more dependent variables can be selected for plotting. Data is plotted using the APTPlot pug-in.

Note In order to plot table data, Configuration Tool must have the path to the Acegrace executable defined.

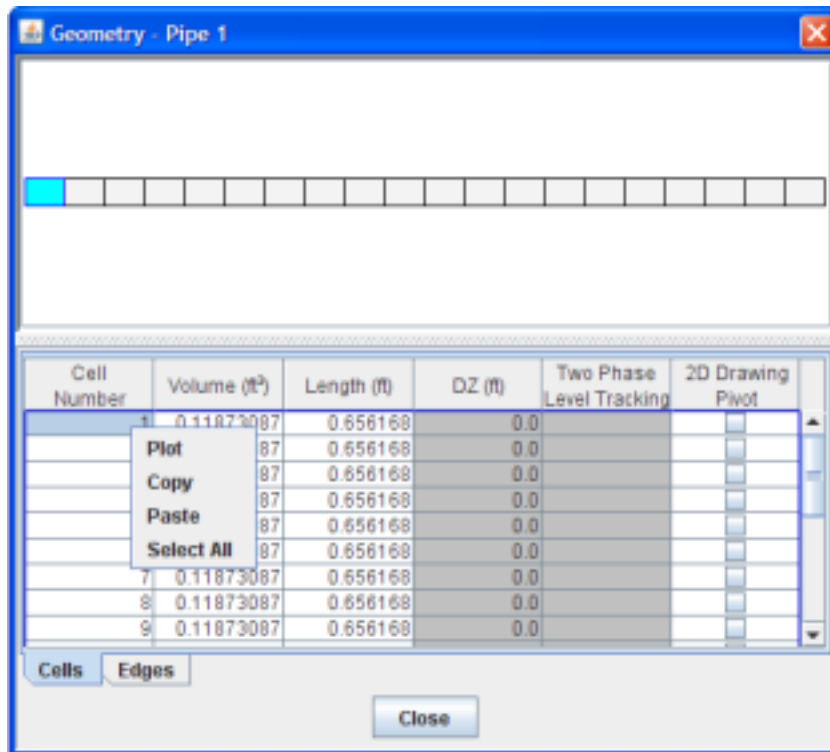


Figure 5.2. Custom Editor Options

Additional editors are displayed in the following sections.

5.1. Model Options

All of the properties of a model that appear outside of other components can be found in Model Options. This includes global properties, such as the model name, and comment section as well as the run controls. The Namelist variables, the constrained steady state controllers, and the hydraulic path steady state controllers are properties of model options and can be found in the property view. [Figure 5.3, “Namelist Properties”](#) below displays the general Namelist properties for the model. The **Show Inactive Entries** option limits the displayed properties to those namelist variables that are currently active. Entering filter text in the provided textfield updates the namelist variable to only those which contain the specified text.

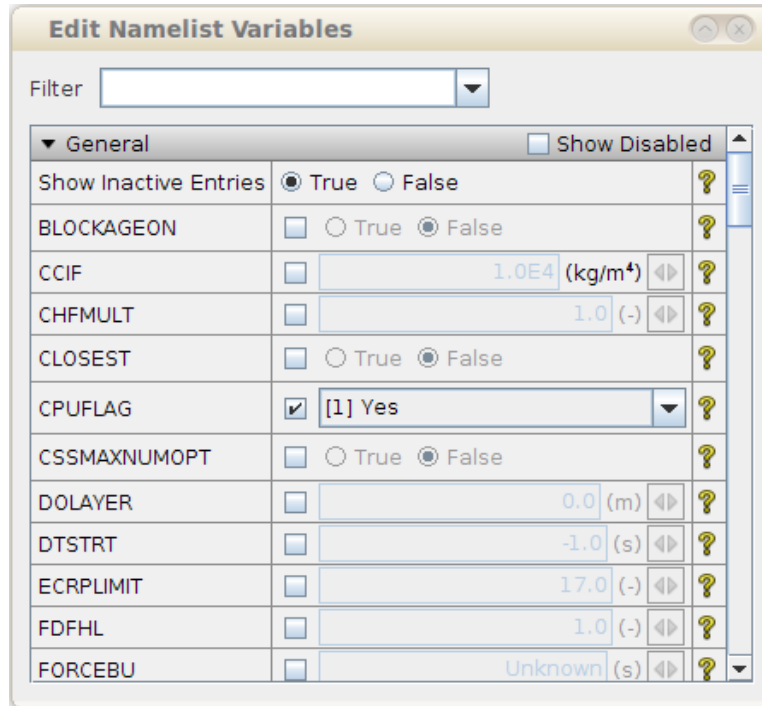


Figure 5.3. Namelist Properties

Note Both the tool-tip message, and the pop-up help contain the variable name of a Namelist variable.

The timesteps data for a particular model are also contained inside the Model Options. [Figure 5.4, “Time Step Table Dialog”](#) below displays the timestep table for an existing model. This table must have at least one entry for a Transient model.

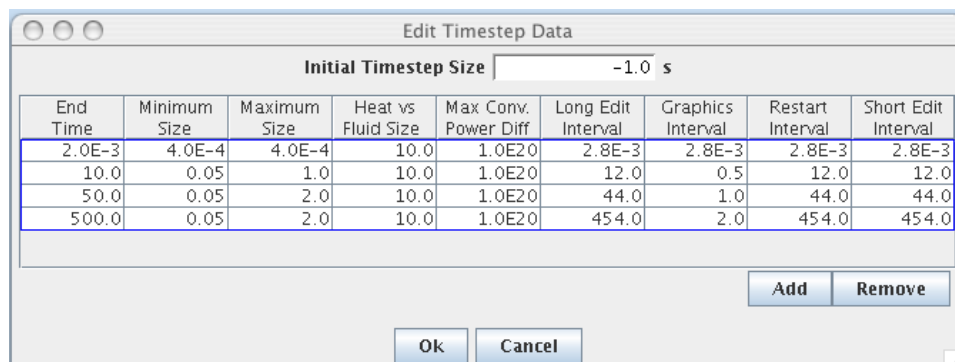


Figure 5.4. Time Step Table Dialog

One of the available namelist options provided by TRACE allows a user to define the loss coefficient format for data entered into a TRACE model. The IKFAC namelist variable allows a specification of Fric or K-Factor type loss coefficients. The IKFAC option can be edited by the namelist variable editor displayed in the figure below:

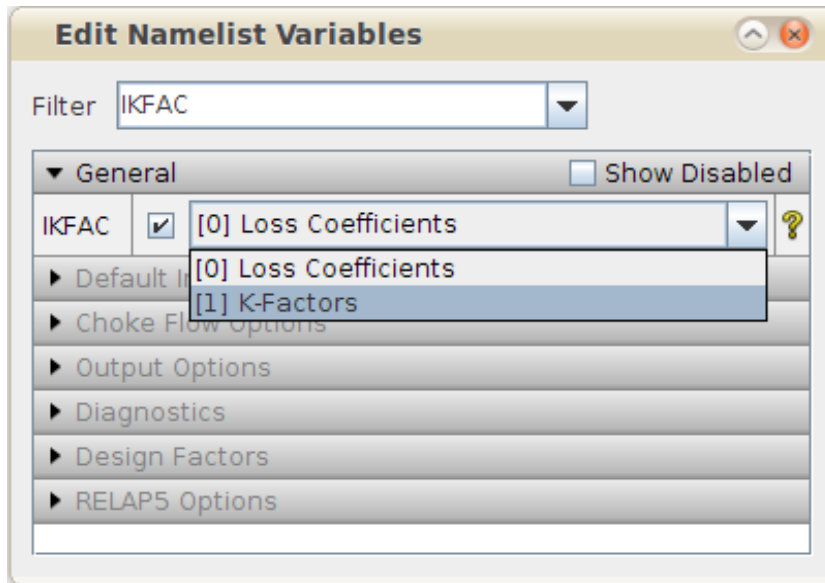


Figure 5.5. IKFAC Namelist Variable

The TRACE plug-in provides a utility to convert FRICS to/from K-Factors. Setting the IKFAC enumeration editor to a different Friction Option will cause the TRACE plug-in to generate a report describing all of the conversions required to use the new loss format. The provided report details information on only those junctions which have losses defined (Any loss coefficient of 0.0 is ignored.) Any values used by the plug-in to convert the loss coefficients, will be displayed in the report. A brief explanation of those values are provided below.

- Hydraulic Component - The owner of the edge which contains the loss coefficients to be converted.
- Edge - The edge which contains the loss coefficients to be converted. In most cases edge ordering starts at the inlet of the component and increments to the outlet side of the component. For 3D components, the edge face direction is provided along with the coordinates of the cells which share that edge.
- Source Volume - The cell number which is the source of the specified edge. For 3D components, cells are identified by coordinate locations as opposed to numbers.
- Source Cell Length - The computed dx for the cell on the source side of the specified edge.
- Target Volume - The cell number which is the target of the specified edge. For 3D components, cells are identified by coordinate locations as opposed to numbers.
- Target Cell Length - The computed dx for the cell on the target side of the specified edge.
- Hydro. Diameter - The hydraulic diameter of the specified edge.
- Forward Loss (FRIC) - The FRIC value appropriate for the current edge. If the model previously defined K-Factors, this would be the calculated loss, otherwise this is the original loss specified in the TRACE model.
- Forward Loss (KFAC) - The KFAC value appropriate for the current edge. If the model previously defined Frics, this would be the calculated loss, otherwise this is the original loss specified in the TRACE model.

Once the report has been provided the OK button on the dialog will accept the displayed changes to the loss coefficients while the cancel button will leave all loss coefficients unchanged. A sample report is shown in the following figure.

Hydro. Component	Edge	Source Volume	Source dx	Target Volume	Target dx	Hydro. Diam	For Loss (FRIC)	For Loss (K-FAC)	Rev. Loss (FRIC)	Rev. Loss (K-FAC)	For V.Loss (FRIC)	For V.Loss (K-FAC)	Rev. V.L (FRIC)
Vessel 20 (1, 1)	Asial Face	2,6,14	0.5588	2,6,14	1.0287	0.45085	2.84E-6	1.0E-5	0.0	0.0	2.84E-6	1.0E-5	0.0
Vessel 20 (1, 2)	Asial Face	2,6,14	0.5588	2,6,14	1.0287	0.45085	2.84E-6	1.0E-5	0.0	0.0	2.84E-6	1.0E-5	0.0
Tee 102 (S1, 1)	Cell 1	3,1,13	0.73772	Cell 1	1.0185	0.7874	0.12536	0.2	0.0	0.0	0.0	0.0	0.0
Tee 102 (S1, 6)	Cell 4	0.7874	Cell 5	0.82113	0.2572	0.07994	0.5	0.0	0.0	0.0	0.0	0.0	0.0
Tee 104 (S1, 4)	Cell 2	0.69068	Cell 1	0.755	0.92405	0.38345	0.6	0.0	0.0	0.0	0.0	0.0	0.0
Tee 104 (S1, 5)	Cell 2	0.7874	Cell 4	1.9681	0.3867	0.08853	0.5	0.0	0.0	0.0	0.0	0.0	0.0
Tee 104 (S1, 6)	Cell 4	1.9681	Cell 1	3.0	0.3667	3.69054	0.05	0.0	0.0	0.0	0.0	0.0	0.0
Pipe 110 (S, 1)	Cell 3	0.69068	Cell 1	0.755	0.92405	0.38345	0.6	0.0	0.0	0.0	0.0	0.0	0.0
Pipe 110 (S, 3)	Cell 2	0.755	Cell 3	0.664718	0.01543	5.43417	0.5	0.0	0.0	0.0	0.0	0.0	0.0
Pipe 110 (S, 27)	Cell 26	0.654718	Junction 121	1.59285	0.01543	4.09005	0.6	0.0	0.0	0.0	0.0	0.0	0.0
Tee 112 (S1, 1)	Cell 14	0.3048	Cell 1	0.3048	7.43E-3	0.09750	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Tee 112 (S1, 2)	Cell 1	0.3048	Cell 2	0.9906	7.43E-3	0.04588	8.0	0.0	0.0	0.0	0.0	0.0	0.0
Tee 112 (S1, 3)	Cell 2	0.9906	Cell 3	0.9906	7.43E-3	0.01875	5.0	0.0	0.0	0.0	0.0	0.0	0.0
Tee 112 (S1, 4)	Cell 3	0.9906	Cell 4	0.9906	7.43E-3	0.01875	5.0	0.0	0.0	0.0	0.0	0.0	0.0
Tee 112 (S1, 5)	Cell 4	0.9906	Cell 5	0.9906	7.43E-3	0.01875	5.0	0.0	0.0	0.0	0.0	0.0	0.0
Tee 112 (S1, 6)	Cell 5	0.9906	Cell 6	0.9906	7.43E-3	0.01875	5.0	0.0	0.0	0.0	0.0	0.0	0.0
Tee 112 (S1, 7)	Cell 6	0.9906	Cell 7	0.9906	7.43E-3	0.01875	5.0	0.0	0.0	0.0	0.0	0.0	0.0
Tee 112 (S1, 8)	Cell 7	0.9906	Cell 8	0.9906	7.43E-3	0.01875	5.0	0.0	0.0	0.0	0.0	0.0	0.0
Tee 112 (S1, 9)	Cell 8	0.9906	Cell 9	0.9906	7.43E-3	0.01875	5.0	0.0	0.0	0.0	0.0	0.0	0.0
Tee 112 (S1, 10)	Cell 9	0.9906	Cell 10	0.9906	7.43E-3	0.01875	5.0	0.0	0.0	0.0	0.0	0.0	0.0
Tee 112 (S1, 11)	Cell 10	0.9906	Cell 11	0.975538	7.43E-3	0.01990	5.0	0.0	0.0	0.0	0.0	0.0	0.0
Tee 112 (S1, 12)	Cell 11	0.875538	Cell 12	0.807212	3.5867	10.657257	5.0	0.0	0.0	0.0	0.0	0.0	0.0
Tee 112 (S1, 13)	Cell 12	0.807212	Cell 13	0.74445	0.40625	4.6506	0.4	0.0	0.0	0.0	0.0	0.0	0.0

Figure 5.6. FRICS/K-FACTORS Conversion Report

Chapter 6. Hydraulic Components

TRACE components used to model fluid dynamics are grouped together as hydraulic components. These include boundary components (Fill and Break), 1D fluid segments (Pipe, Plenum, Pump, Prizer, Tee and Valve) as well as the 3D Vessel component. Hydraulic components are found by expanding the *Hydraulic Components* node in the navigator.

6.1. Break

A Break component is a boundary component typically used to provide a sink for liquid and vapor flows exiting the system. Breaks may have constant fluid conditions, table defined conditions, or conditions controlled by a control system. [Figure 6.1, “Break Properties”](#) displays the general properties of a newly created Break. The variable *IBTY* or *Break Type* determines how the break will behave.

Break 9 (\$9\$ bkn-loop sec pressure bc)	
General	Show Disabled
Component Name	\$9\$ bkn-loop sec pressure bc
Component Number	9
Description	
Inlet	Tee 27 Cell 3 outlet
Break Type	[0] No Tables
Temperature Table Option	[0] Enter liquid/gas temp
Fluid State Option	[0] Last Interp State Held Const.
Length	4.35 (m)
Volume	24.8 (m³)
Initial Gas Volume Fraction	1.0 (-)
Initial Mixture Temperature	535.14 (K)
Initial Pressure	4.85E6 (Pa)
Initial Noncondensable PP	0.0 (Pa)
Adjacent Pressure Flag	<input type="radio"/> True <input checked="" type="radio"/> False
Max Pressure Change Rate	1.0E5 (Pa/s)
▶ Contan Coupling	
▶ Scale Factors	
▶ State Controllers	
▶ Trace Species	

Figure 6.1. Break Properties

6.1.1. Break Type

The *Break Type* (*IBTY*) for new breaks is initialized to 0. This defines a break with constant fluid conditions. The available options for break type and what they represent are listed below in [Table 6.1, “Break Type Values”](#).

IBTY Value	Description
0	Constant Fluid Conditions
1	Pressure Controlled By Table
2	Pressure and Temperature Controlled By Table
3	Pressure, Temperature and Void Fraction Tables
4	Same as 3 with Non-condensable gas partial pressure
5	Same as 4 with the Solute to Coolant Ratio
6	Control System defined conditions
7	Contan coupled break

Table 6.1. Break Type Values

6.1.2. Break Table

The fluid condition values for break types 1 through 5 are entered in the *Break Table*. This table includes a control system component reference for obtaining the independent variable value, an optional trip reference for activating or deactivating the table, and a flag that determines how the independent variable value should be calculated. An example break table for a type 5 break is displayed below in [Figure 6.2, “Break Table”](#).

Editing Break Table

Independent Variable:

Controller Trip:

Independent Variable Form: ☐ - negative ☒ + positive

Table Variable ☐

Independent Variable	Pressure Pa
0.0	1.0E5
10.0	2.0E5

Figure 6.2. Break Table

6.1.3. Break Scale Factors

A non-constant break allows the user to apply a scale factor to the state variables. These factors are located in the Scale Factors attribute group, as displayed below in [Figure 6.3, “Break Scale Factors”](#). The specified scale factor is applied to the appropriate state variable, whether the variable is defined in a table, or is specified from a generalized state controller.

☐ Scale Factors

Pressure Scale	1.0	(-)	?
Liquid Temperature Scale	1.0	(-)	?
Vapor Temperature Scale	1.0	(-)	?
Noncondensable Scale	1.0	(-)	?
Solute Scale	1.0	(-)	?

Figure 6.3. Break Scale Factors

6.1.4. Break State Controllers

Type 6 Breaks use control systems to define the current fluid state. Any property without a controller is defined using the corresponding constant property in the general data. If interactive variables are selected, the associated properties can be adjusted during run-time to steer the calculation. The generalized state variables are displayed in [Figure 6.4, “Break State Controllers”](#).

☐ State Controllers

Vapor Volume Controller	<none>	S ⁺	?
Solute Controller	<none>	S ⁺	?
Noncondensable Controller	<none>	S ⁺	?
Pressure Controller	<none>	S ⁺	?
Liquid Temperature Controller	<none>	S ⁺	?
Vapor Temperature Controller	<none>	S ⁺	?

Figure 6.4. Break State Controllers

6.1.5. Break Contan Coupling

A Break component can be attached to a Contan compartment when the break type (IBTY) is set to 7. This enables the contan data shown below in [Figure 6.5, “Break Contan Coupling”](#). A Contan component must be defined in the model.

☐ Contan Coupling

CONTAN Compartment	5	?
CONTAN Compartment Region	Liquid Space	?
Vapor Equilibrium Fraction	1.0	(-) ?
Fluid Equilibrium Fraction	2.0	(-) ?

Figure 6.5. Break Contan Coupling

6.1.6. Break Trace Species

The mass fractions of gas and vapor species are defined in the *Trace Species* data for a Break when the trace species values are enabled in the model. The available values reflect the model properties. The trace species properties of a break are shown below in [Figure 6.6, “Break Trace Species”](#).

Trace Species

Air Fraction	1.0 (-)	?
Hydrogen Fraction	0.0 (-)	?
Helium Fraction	0.0 (-)	?
Nitrogen Fraction	0.0 (-)	?
Argon Fraction	0.0 (-)	?
Xenon Fraction	0.0 (-)	?
Krypton Fraction	0.0 (-)	?

Figure 6.6. Break Trace Species

6.1.7. Break Control Connections

An additional connection point is available for break types that allow control system inputs. This connection point is shown on the side of the Break and can be used to establish control system connections using the connect tool. The [Figure 6.7, “Break Hydro Input Dialog”](#) is displayed when creating a control system connection to a Break to allow the input to be selected.

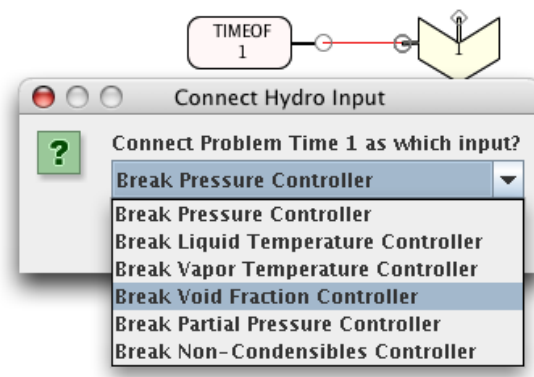
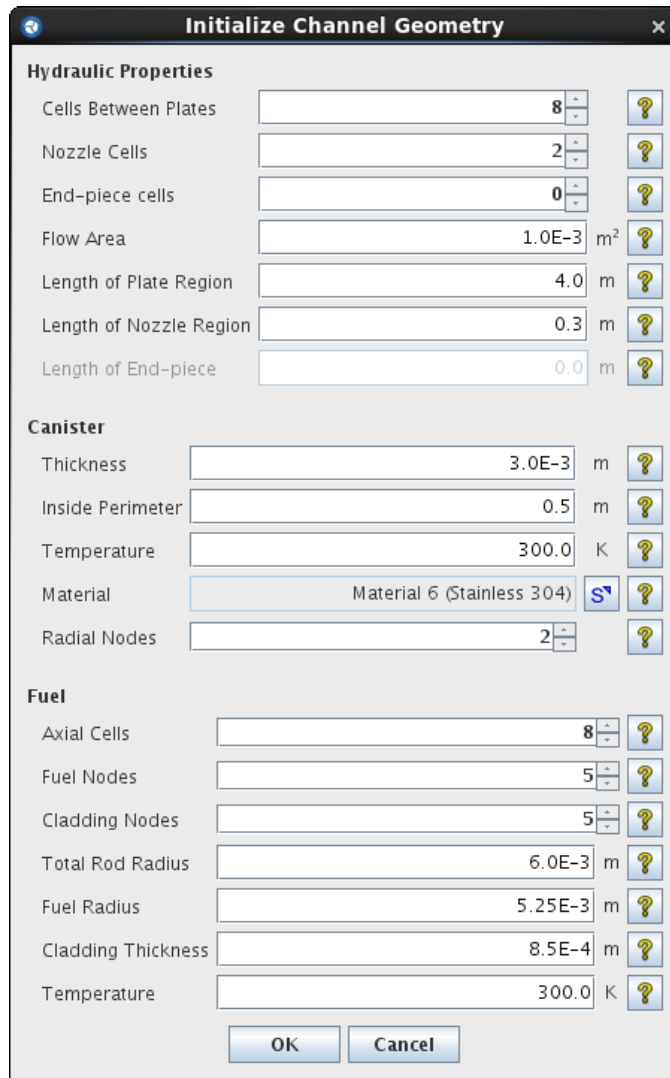


Figure 6.7. Break Hydro Input Dialog

6.2. Channel

TRACE allows the user to implement a bundled fuel channel component as a hydraulic component that contains all of the heat structure properties for the canister wall, and fuel rods. The standard hydraulic properties included in a channel are detailed in [Section 6.5, “Pipe”](#). These properties include geometry, friction, initial fluid conditions, etc. [Figure 6.8, “Channel Completion Dialog”](#) displays the completion dialog for initializing a new channel.



Initialize Channel Geometry

Hydraulic Properties

Cells Between Plates	8	?
Nozzle Cells	2	?
End-piece cells	0	?
Flow Area	1.0E-3	m ² ?
Length of Plate Region	4.0	m ?
Length of Nozzle Region	0.3	m ?
Length of End-piece	0.0	m ?

Canister

Thickness	3.0E-3	m ?
Inside Perimeter	0.5	m ?
Temperature	300.0	K ?
Material	Material 6 (Stainless 304)	S ?
Radial Nodes	2	?

Fuel

Axial Cells	8	?
Fuel Nodes	5	?
Cladding Nodes	5	?
Total Rod Radius	6.0E-3	m ?
Fuel Radius	5.25E-3	m ?
Cladding Thickness	8.5E-4	m ?
Temperature	300.0	K ?

OK Cancel

Figure 6.8. Channel Completion Dialog

6.2.1. Channel General Properties

The general channel preferences include options for defining the general channel options. [Figure 6.9, “Channel Properties”](#) displays the properties of a blank channel created in the Model Editor.

Liquid HTC	Unknown (W/m ² /K)	?
Vapor HTC	Unknown (W/m ² /K)	?
Outside Liquid Temp.	Unknown (K)	?
Outside Vapor Temp.	Unknown (K)	?
Advanced BWR Design	<input checked="" type="radio"/> True <input type="radio"/> False	?
Rod Symmetry	[0] Constant Pitch-to-Diameter	?
Surface Rays	0	?
Fuel Bundles	1	?
Rods per Row	4	?
Use Radiation Enclosure	<input type="radio"/> True <input checked="" type="radio"/> False	?
Rod Locations		E* ?
Water Rods	Valid values	E* ?
Fine Mesh Reflood	<input type="radio"/> True <input checked="" type="radio"/> False	?
Radiation Model	0	?
Anisotropic Calculation	[0] On	?

Figure 6.9. Channel Properties

6.2.2. Channel Canister Properties

The canister wall for the channel is defined inside the **Canister Wall** attribute group for the channel. The **initial temperature** array, number of meshpoints and the material for the canister wall are all defined in **Wall Material**. The canister wall has the same number of axial nodes as the channel has cells. The outside of the channel may be connected to a separate hydraulic component, through the outside component property. Figure 6.10, “Channel Canister Wall” below displays the canister wall properties of a channel.

♀ Canister Wall

Inner Radius	0.79788456 (m)	?
Wall Thickness	3.0E-3 (m)	?
Maximum Axial Nodes	100	?
Minimum Node Distance	1.0E-3 (m)	?
Liquid Level Tracking	<input type="radio"/> True <input checked="" type="radio"/> False	?
Axial Conduction	<input type="radio"/> True <input checked="" type="radio"/> False	?
Width	0.5 (m)	?
Wall Material	Valid Values	E* ?
Canister Emissivity 1	Unknown (-)	?
Canister Emissivity 2	Unknown (-)	?
Canister Emissivity 3	Unknown (-)	?
Wall Temperature	[11][2] Temperature Values	E* ?
Outside Component	Constant BCs	E* ?

Figure 6.10. Channel Canister Wall

6.2.3. Channel Fuel Properties

The fuel portion of the channel is defined in two locations. The general rod properties are defined inside the **Fuel Rod** attribute of the channel, whose properties are displayed in Figure 6.11, “Channel Fuel Rods”. The properties of non-average fuel rods are defined through the navigator, as sub-nodes. These rods may include a last axial node to implement partial-length rods in advanced rod models.

Fuel Rods

Rod Ratio	Unknown (-)	?
Pellet Radius	Unknown (m)	?
Non-Average Rods	Valid values	E?
Rod Thickness	Unknown (m)	?
Meshpoints	Invalid Material	E?
Initial Temperature	Temperature [10][9]	E?
Critical Heat Flux	[1] AECL_IPPE	?
Fuel-clad Interaction	<input type="radio"/> True <input checked="" type="radio"/> False	?
FCI Calculations	0	?
Maximum Axial Nodes	100	?
Minimum Node Distance	1.0E-3 (m)	?
Gas Gap HTC	6300.0 (W/m ² /K)	?
Rod Emissivity 1	Unknown (-)	?
Rod Emissivity 2	Unknown (-)	?
Rod Emissivity 3	Unknown (-)	?
Fuel Nodes	Invalid Values	E?
Metal Water Reaction	<input type="radio"/> True <input checked="" type="radio"/> False	?
Surface Multiplier	1.0 (-)	?

Figure 6.11. Channel Fuel Rods

6.2.4. Channel Rod Locations

The rod location table is a human readable table for placing rods inside the fuel channel. Non-average fuel rods and water rods require a specific location inside the fuel channel. These locations are specified through the rod locations dialog displayed in [Figure 6.12, “Channel Rod Location Dialog”](#). The list on the left side of the dialog displays each type of rod that is present in the current channel. The grid display shows each fuel pin grid location, and what rod type is used at that location.

Clicking in the grid display will select a fuel rod location. Holding the shift-key down when making a selection will add all of the grid locations between the last selected location and the currently selected location. Holding the control-key down when making a selection will toggle whether or not the grid location is included in the current selection.

To change what rod type is used for a given location pressing the **Assign** button with one or more rod locations selected, then immediately select the rod type in the list that you want to use for the selected locations. Water rods locations cannot be defined in this dialog. They are defined through the Location editor of a particular water rod.

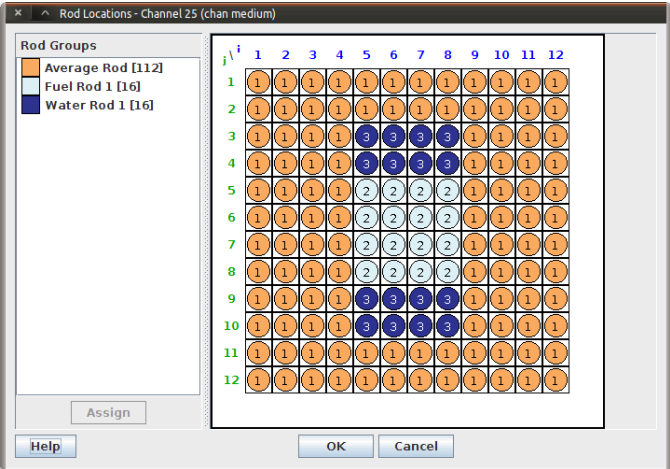


Figure 6.12. Channel Rod Location Dialog

6.2.5. Advanced Channel Features

The TRACE channel includes advanced boiling water reactor features such as partial length rods, and water rods. Partial length rods are Non-Average Rods with the **Last Axial Node** property set to a value less than the number of axial nodes in the channel. Water rods are defined inside the channel directly, in a custom dialog shown in [Figure 6.13](#), “Channel Water Rods”.

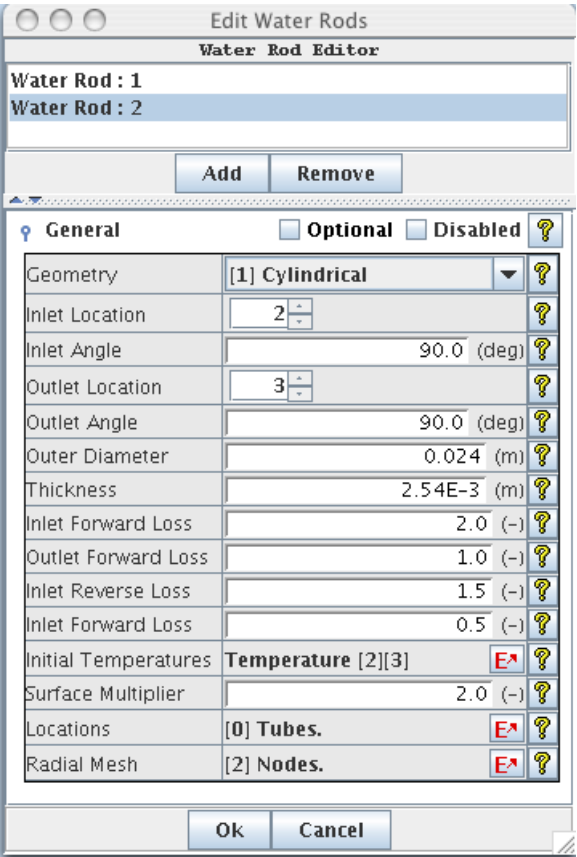


Figure 6.13. Channel Water Rods

6.3. Fill

The Fill component is a boundary component typically used as a source for fluid and vapor flows into the system. Fill components can be defined with constant fluid conditions, tabular fluid conditions, or with generalized state controllers. Figure 6.14, “Fill Properties” below displays the properties for a newly created Fill.

Fill 55

☒ General ☐ Optional ☐ Disabled

Component Name	unnamed	?
Component Number	55	?
Description	<none>	E ?
Comments	<none>	E ?
Fill Type	[1] Constant Velocity	?
Length	Unknown (m)	?
Volume	Unknown (m³)	?
Initial Gas Volume Fraction	0.0 (-)	?
Initial Liquid Temperature	372.756 (K)	?
Initial Vapor Temperature	372.756 (K)	?
Initial Pressure	1.0E5 (Pa)	?
Initial Noncondensible PP	0.0 (Pa)	?
Liquid Vel	0.0 (m/s)	?
Vapor Vel	0.0 (m/s)	?
Max Acceleration	1.0E20 (m/s²)	?
Rate Factor Table	Rows: 0 []	E ?
Couple to CONTAN Comp...	<input type="radio"/> True <input checked="" type="radio"/> False	?

Figure 6.14. Fill Properties

6.3.1. Fill Type

The *Fill Type (IFTY)* for new fills is initialized to 0. This defines a fill with constant fluid conditions. The available options for fill type and what they represent are listed below in Table 6.2, “Fill Type Values”.

Value	Description
1	Constant Velocity
2	Constant Mass Flow
3	Constant Generalized State
4	Tabular Based Velocity
5	Tabular Based Mass Flow
6	Tabular Based Generalized State
7	Trip enabled Velocity Table
8	Trip enabled Mass Flow Table
9	Trip enabled Generalized State Table
10	Control System Based Generalized State
11	Control System Based Generalized Mixture State

Table 6.2. Fill Type Values

6.3.2. Fill Table

The properties included in the Fill table vary based on the selected *IFTY* value. If the *IFTY* value is 4 or 7, the dependent variable in the table is the mixture velocity. An *IFTY* value of 5 or 8 results in a table with a mixture mass-flow for the dependent variable. If the *IFTY* value is 6 or 9, the table contains all the values of a fluid state as dependent variables. The unit type of the independent variable column is always based on a signal variable or control block selected. Figure 6.15, “Fill Generalized State Table” below displays the generalized state table for a fill.

Editing Fill Table

Independent Variable: Problem Time 1

Controller Trip: <none>

Independent Variable Form: ☐ Negative ☒ Positive ?

X	Liquid Velocity m/s	Gas Velocity m/s	Liquid Temp. K	Gas Temperature K	Gas Vol Frac -	Pressure Pa	Partial Pressure Pa
0.0	0.0	0.0	376.0	376.0	0.0	1.0E5	0.0
1.0	5.0	5.0	376.0	376.0	0.0	1.0E5	0.0
100.0	0.0	0.0	376.0	376.0	0.0	1.0E5	0.0

Add Row Remove Row

OK Cancel

Figure 6.15. Fill Generalized State Table

6.3.3. Fill Contan Coupling

A Fill component can be attached to a Contan compartment by selecting **True** for the *Couple to CONTAN* option. This enables the contan coupling data. This data is the same as that displayed for the break in Figure 6.5, “Break Contan Coupling”. A Contan component must be defined in the model when this data is enabled.

6.3.4. Fill Scale Factors

A generalized state fill allows the user to apply a scale factor to the state variables. These factors are located in the Scale Factors attribute group, as displayed below in Figure 6.16, “Fill Scale Factors”. The specified scale factor is applied to the appropriate state variable, whether the variable is defined in a table, or is specified from a control system state controller.

Scale Factors

Liquid Temperature Scale	1.0 (-)	?
Vapor Temperature Scale	1.0 (-)	?
Pressure Scale	1.0 (-)	?
Noncondensable Scale	1.0 (-)	?
Solute Scale	1.0 (-)	?

Figure 6.16. Fill Scale Factors

6.3.5. Fill State Controllers

Fills of type 10 and 11 have their fluid state defined by control system references. These state controllers can include signal variables or control blocks. The current value of the control system input is used directly as the current value for the appropriate fluid state. [Figure 6.17, “Fill State Controllers”](#) below displays the fill state controllers in the property view.

State Controllers

Vapor Volume Controller	<none>	S^	?
Solute Controller	<none>	S^	?
Liquid Flow Controller	<none>	S^	?
Vapor Flow Controller	<none>	S^	?
Noncondensable Controller	<none>	S^	?
Pressure Controller	<none>	S^	?
Liquid Temperature Controller	<none>	S^	?
Vapor Temperature Controller	<none>	S^	?

Figure 6.17. Fill State Controllers

6.3.6. Fill Control Connections

Any fill that includes a control system input has an additional connection point inside 2D views. This additional connection point shows input connections from control systems, and allows control system connections to be input using the connect tool. [Figure 6.18, “Fill Hydro Input Dialog”](#) below illustrates this connection process.

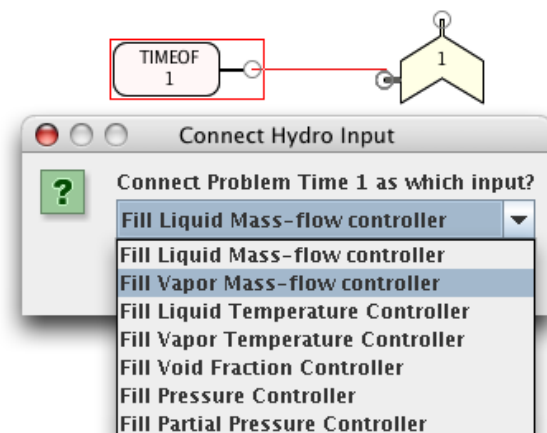


Figure 6.18. Fill Hydro Input Dialog

6.4. Feedwater Heater

Feedwater heaters are modeled in TRACE as Tee-style component. The main-tube represents the heater volume while the side-tube represents the steam input from the turbine. The standard hydraulic properties included in a heater are detailed in [Section 6.5, “Pipe”](#). These properties include geometry, friction, initial fluid conditions, etc. The Tee properties that a feedwater heater must implement are detailed in [Section 6.11.1, “Tee Properties”](#) below. [Figure 6.19, “Feedwater Heater Completion Dialog”](#) below displays the completion dialog for a new feedwater heater.

Define the Heater	
Main Tube Length	4.0 m
Side Tube Length	1.0 m
Main Tube Flow Area	1.0 m ²
Side Tube Flow Area	0.1 m ²
Number of Main Tube Cells	3
<input type="button" value="Ok"/> <input type="button" value="Cancel"/>	

Figure 6.19. Feedwater Heater Completion Dialog

6.4.1. Feedwater Heater Properties

The general properties of a heater are displayed below in [Figure 6.20, “Feedwater Heater Properties”](#). These properties control the heat transfer between the shell hydraulics in the heater component and the hydraulic component connected through a heat structure.

Vertical Heater	Horizontal	?
Tube Bank Outer Diameter	Unknown (m)	?
Baffle Distance	Unknown (m)	?
Drain Height	Unknown (m)	?
Shell Void Vs Liquid HT	Rows: 0 []	E?
Drain Void vs Liquid HT	Rows: 0 []	E?
Shell Void Vs Liquid Level	Rows: 0 []	E?

Figure 6.20. Feedwater Heater Properties

6.4.2. Feedwater Heater Valve

The feedwater heater includes a valve that controls the steam input flow from the turbine. The valve properties for a new feedwater heater are displayed in [Figure 6.21, “Feedwater Heater Valve”](#).

Valve	
Independent Source	<none> S?
Valve Interface Index	Edge 1 of 4 ?
Flow Area	Unknown (m ²) ?
Hydraulic Diameter	Unknown (m) ?
Flow Area Fraction	Unknown (-) ?

Figure 6.21. Feedwater Heater Valve

6.5. Pipe

The pipe component is used to represent one or more pipes through the system. [Figure 6.22](#), “[Pipe Completion Dialog](#)” displays the completion dialog for a new pipe created inside the ModelEditor. If the completion dialogs are not enabled, the TRACE plug-in will remember the last values entered. For a newly created pipe, the cell volumes will be evenly divided along the specified length.

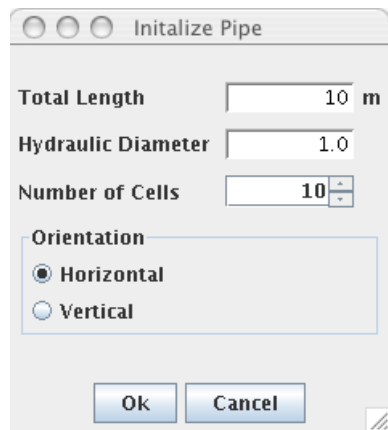


Figure 6.22. Pipe Completion Dialog

6.5.1. Pipe Properties

The majority of pipe properties are cell and edge properties. These properties can be edited using the custom editors provided for *Component Geometry*, *Initial Conditions*, *Friction*, etc. Notable exceptions to this are the pipe wall, wall power, and power to the fluid portions of the pipe. [Figure 6.23](#), “[Pipe General Properties](#)” below displays the general properties of a pipe.

Pipe 1			
General		Optional	Disabled
Component Name	unnamed		?
Component Number	1		?
Description	<none>	E^	?
Comments	<none>	E^	?
Component Geometry	Cells: 8	E^	?
Initial Conditions	[Valid Conditions]	E^	?
Friction	Fric (0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...	E^	?
Critical Heat Flux	[1] Biasi		?
Wall Roughness	0.0 (m)		?
Pipe Type	[0] No Accumulator		?
Number of Pipes	1		?
Leak Paths	[0] Leak Paths	E^	?

Figure 6.23. Pipe General Properties

6.5.2. Pipe Geometry

The geometry of a pipe is edited through the component geometry dialog. The component geometry dialog consists of a table for editing the values and a display that shows the pipe. This dialog allows the user to change the volume and length of cells, as well as the diameter, flow area, and orientation of edges within a pipe. If the *IELV* namelist variable is set to 1 the edge angles will be calculated from the specified cell elevations. [Figure 6.24, “Pipe Cell Geometry”](#) displays the cells panel of the pipe geometry dialog. Selections made in the table of the component geometry dialog will be reflected in the visual representation and vice versa. Values that cannot be edited (e.g. total volume) are shaded in grey.

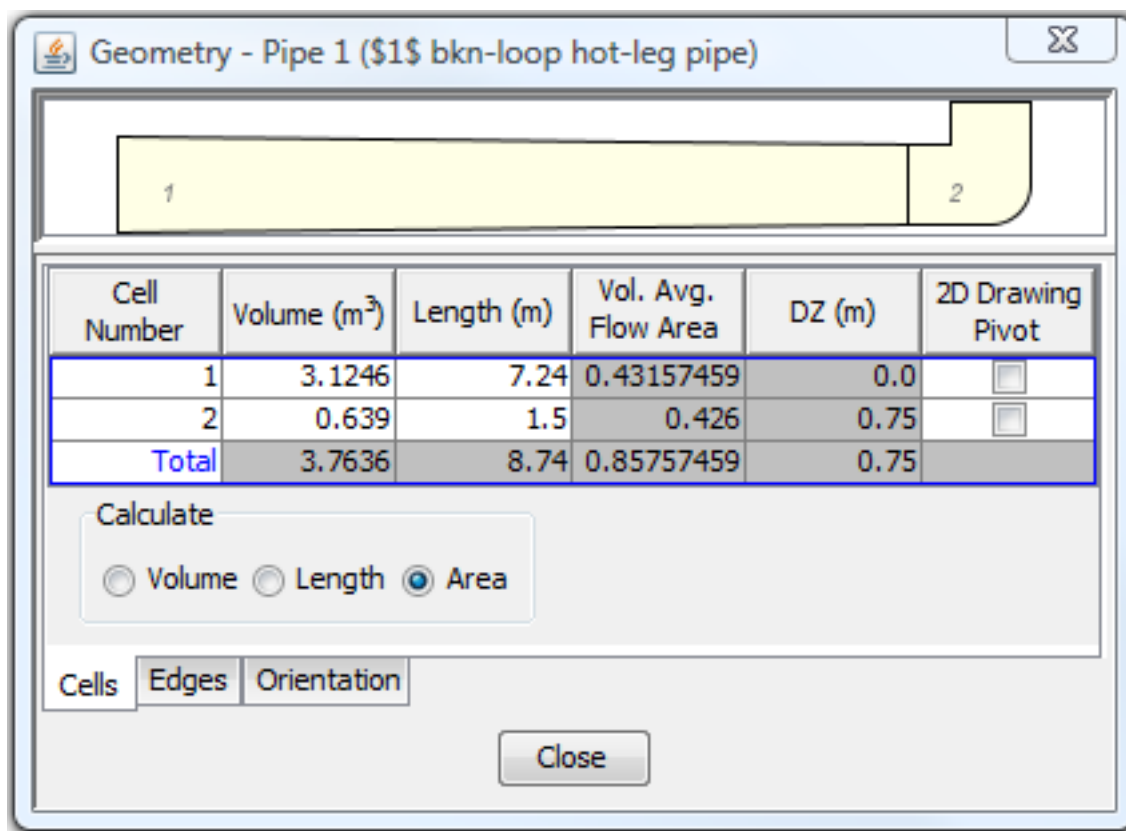


Figure 6.24. Pipe Cell Geometry

The cell geometry editor for most cell based hydraulic components provides horizontal cell center pivoting. Pivoting hydraulic cells is useful in providing a more accurate physical representation of hydraulic components, but has no bearing on the actual geometrical data. Hydraulic components which have been pivoted are displayed in the 2D view embedded in the geometry editor as well as in the primary 2D view.

In TRACE the orientation of a pipe is controlled through its edges. Each edge may be angled between +90 and -90 degrees with regard to the positive X axis. Additionally, each edge has a hydraulic diameter and flow area. The flow area is used to determine the width of each cell when rendered in a 2D view. [Figure 6.25, “Pipe Edge Geometry”](#) below displays the edge panel in the pipe geometry table.

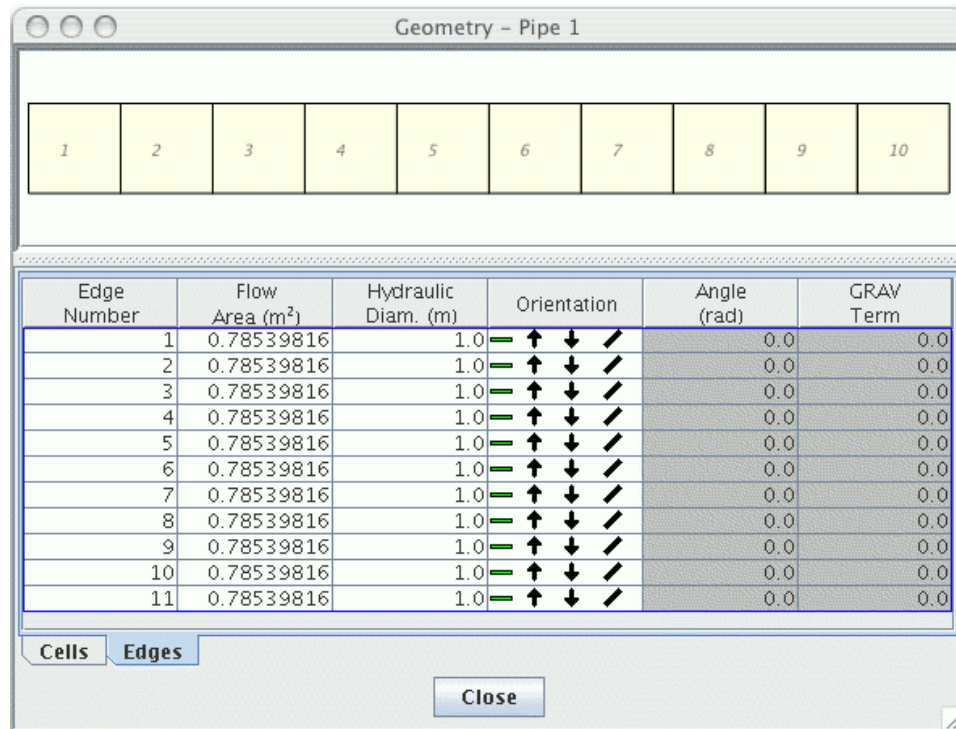


Figure 6.25. Pipe Edge Geometry

The number of cells and edges inside a pipe cannot be edited directly after creation. The 1D renodalization tool can be used to easily modify the nodalization of a pipe, and automatically update any components associated with the pipe. More information on renodalization can be found in [Chapter 16, Renodalization](#).

6.5.3. Pipe Initial Conditions

The initial conditions of the fluid inside a pipe must be defined for both the fluid cells and flow edges. The cells contain the fluid state information, while the edges detail the fluid flow data. These properties are modified in the Pipe Initial Conditions Dialog. This dialog is used to modify the initial conditions of all multi-cell 1D hydraulic components. The volume initial conditions are displayed in [Figure 6.26, “Pipe Cell Initial Conditions”](#) below.

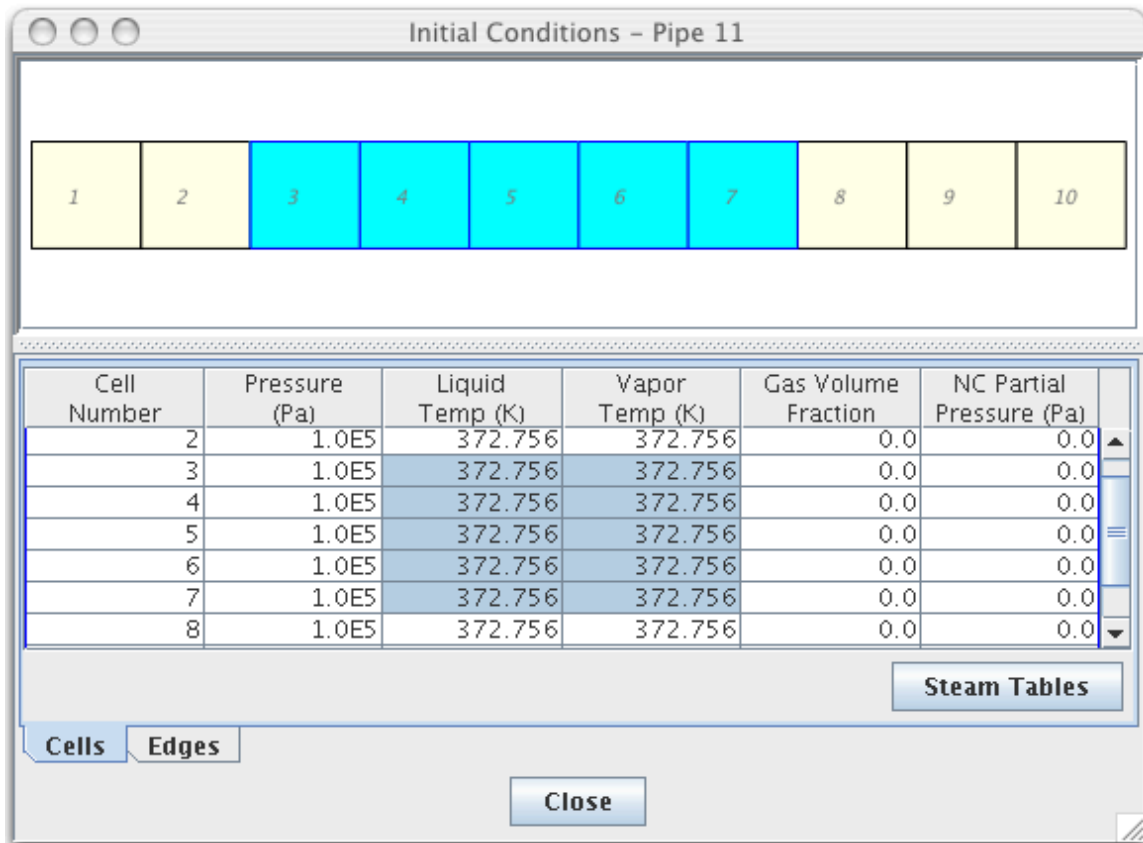


Figure 6.26. Pipe Cell Initial Conditions

The edge initial conditions define the flow velocity for each fluid phase. Positive values indicate flow from the inlet of the pipe (cell 1) towards the outlet of the pipe (cell n). Negative values indicate a reversed flow through the pipe. As with the cell panel, edge selection is reflected in both the table and the visual representation of the pipe. [Figure 6.27, “Pipe Edge Initial Conditions”](#) below displays the edge panel in the pipe initial conditions dialog.

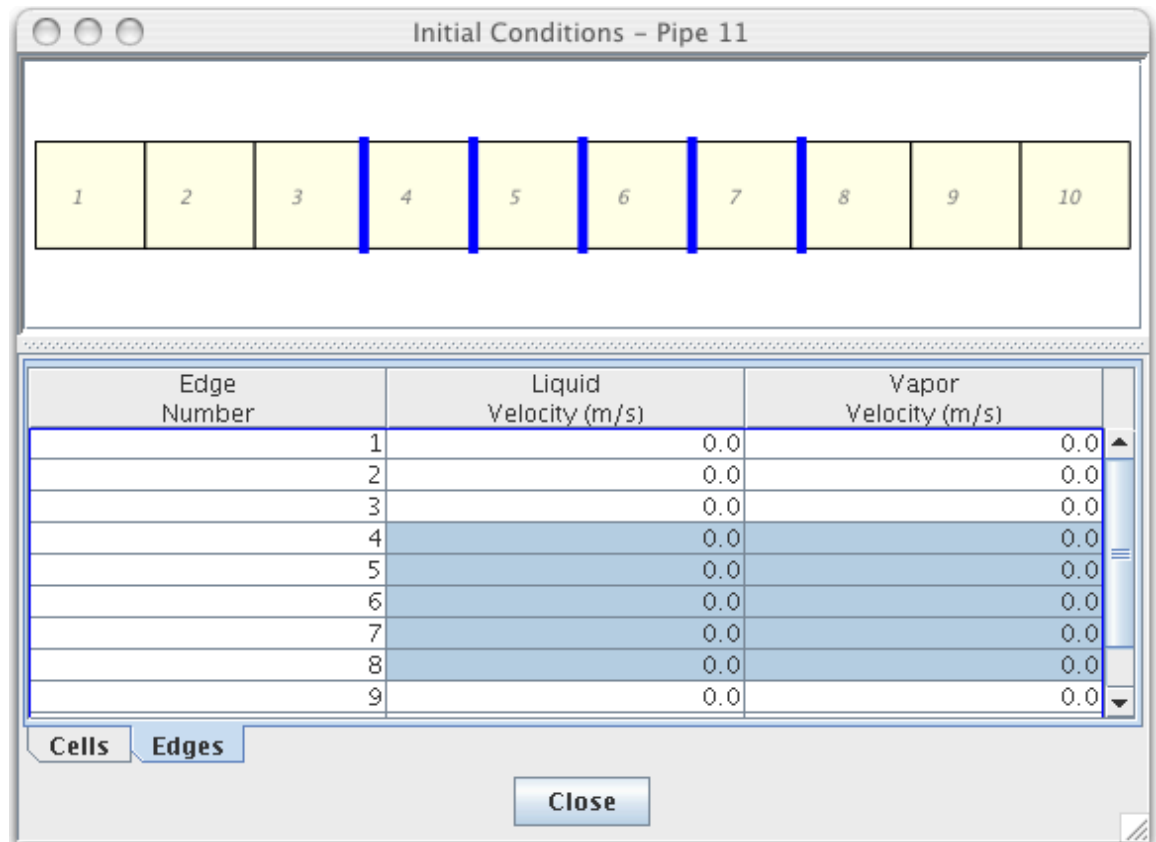


Figure 6.27. Pipe Edge Initial Conditions

6.5.4. Pipe Friction Data

The pipe friction data defines the resistance to fluid flow through each edge. Abrupt area changes in the geometry are represented here. All multi-cell 1D hydraulic components use a similar dialog to edit their friction data. [Figure 6.28, “Pipe Friction Data”](#) below displays the edge friction dialog for a pipe.

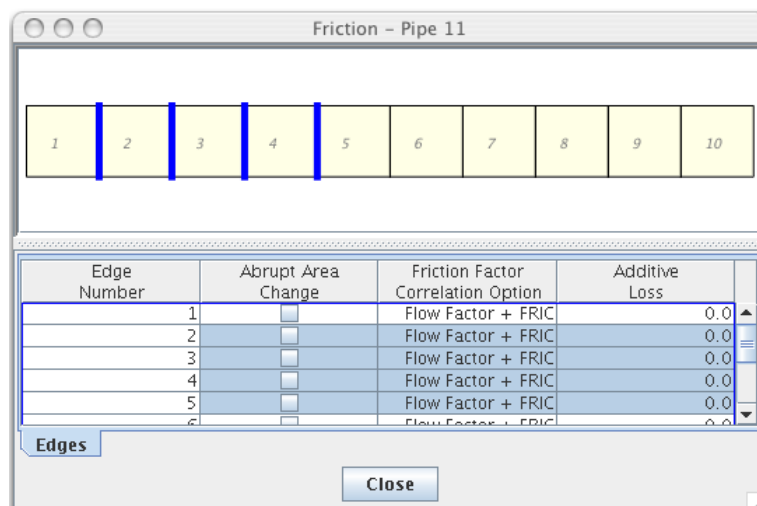


Figure 6.28. Pipe Friction Data

6.5.5. Pipe Wall Heat Structure

Pipes may have an internally defined heat structure used to represent the pipe wall. This heat structure is divided into evenly spaced radial regions defined by the wall thickness. A single material is defined over the entire radius of the pipe. The outer boundary condition of the pipe wall is controlled using constant temperature values and heat transfer coefficients for both the liquid and vapor phases. [Figure 6.29, “Pipe Wall Properties”](#) below displays the pipe wall properties for a simple pipe.

▼ Pipe Wall			
Use Pipewall	<input checked="" type="radio"/> True <input type="radio"/> False		📄 ?
Variable Radial Mesh	<input checked="" type="radio"/> True <input type="radio"/> False		📄 ?
Radial Geometry	5 Radial Nodes.	E	📄 ?
Pipewall Material	Material 6 (Stainless 304)	S	📄 ?
Wall Temperature	[20][6] Temperature Values	E	📄 ?
Outside Component	Constant BCs	E	📄 ?
Inner Radius	0.1 (m)	↔	📄 ?
Wall Thickness	0.025 (m)	↔	📄 ?
Liquid Heat Transfer	0.0 (W/m ² /K)	↔	📄 ?
Vapor Heat Transfer	0.0 (W/m ² /K)	↔	📄 ?
Outside Liquid Temp.	0.0 (K)	↔	📄 ?
Outside Vapor Temp.	0.0 (K)	↔	📄 ?

Figure 6.29. Pipe Wall Properties

6.5.5.1. Variable Mesh Pipe Walls

The TRACE Plug-in supports defining a variable radial mesh for pipewall heat structures. This property is enabled when the pipewall is turned on inside a Pipe. During ASCII export the plug-in will generate a stand-alone heat component which will be included in the ASCII model. This generated heat structure will have all of the same data as the TRACE spawned heat structure for a standard pipewall with the exception of the radial geometry. A standard pipewall has a fixed radial geometry where each node is evenly distributed. The variable mesh heat structure has a user defined radial geometry. [Figure 6.30, “Pipe Wall Radial Geometry”](#) below displays the dialog for defining the variable radial mesh.

This dialog displays the radial geometry of a variable mesh pipewall. The first column in the dialog displays the mesh interval number. The third and fourth columns display the position from the center of the pipe. The fifth and sixth columns display the relative inside and outside positions bounding the interval. These are relative to the inside surface of the pipewall. The last column displays the thickness of each mesh interval.

This dialog provides the capability of automatically generating the radial mesh. The calculation options allow for a geometric series or for equally spaced mesh intervals. The geometric series radial mesh calculation options divide the mesh into regions in such a way that each mesh interval is precisely twice the size of the previous mesh interval. The geometric interval can be calculated

starting from either the inside surface or the outside surface. The number of mesh intervals in the geometric series is defined either explicitly by defining the number of nodes, or implicitly by selecting a maximum size for the starting node of the series.

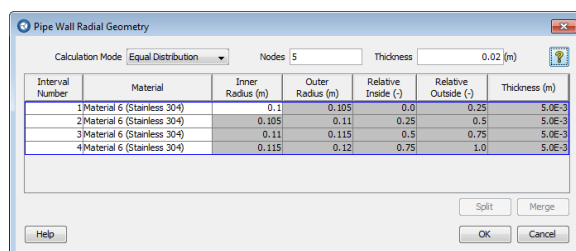


Figure 6.30. Pipe Wall Radial Geometry

6.5.6. Pipe Wall Power

Power can be applied directly to a pipe wall for any pipe in the model. This is a simple way to apply power to a system without defining more complicated heat structure components. The pipe wall must be defined before the wall power can be initialized. The power is applied using an initial value and a power table. The power is divided amongst the radial locations for each axial cell in the pipe based on the defined wall fractions. Figure 6.31, “Pipe Wall Power” below displays the Pipe Wall Power properties for a Pipe.

Wall Power

Wall Power Fractions	[10][2] Power Fractions	E	?
Initial Wall Power	0.0 (W)		?
Off Power	0.0 (W)		?
Max Wall Power Change	0.0 (W/s)		?
Wall Power Scale	1.0 (-)		?
Wall Power Table	[0] Power Table Rows	E	?
Wall Power Rate Factor	Rows: 0 []	E	?

Figure 6.31. Pipe Wall Power

6.5.7. Pipe Fluid Power

TRACE allows for power to be inserted directly into the fluid of a pipe. This is handled through the power to the fluid properties. This data is located on the fluid segment sub-node (*Main Tube*) of a pipe. To access these properties, expand the pipe in the navigator, and select the fluid segment. The fluid power is defined through an initial power value and a power table. Figure 6.32, “Pipe Fluid Power” below displays the fluid power properties for a pipe.

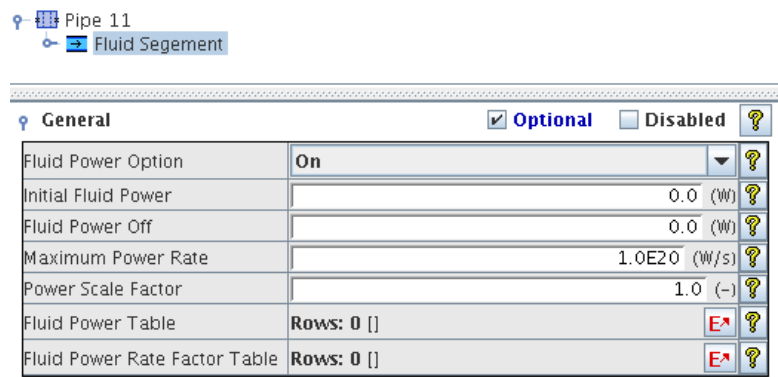


Figure 6.32. Pipe Fluid Power

6.5.8. Pipe Leak Paths

TRACE allows for special hydraulic connections originating from the side of certain hydraulic components, called a leak path. Leak paths are connections from the side of a pipe leading to any other volume in the model. One of the important differences between a leak path and a standard hydraulic connection is that a leak path can include an elevation change. [Figure 6.33, “Pipe Leak Path”](#) displays the properties of a leak path defined in a pipe.

Note Leak Paths can only be created when the "Enable Leakpath Support" TRACE plug-in preference is enabled.

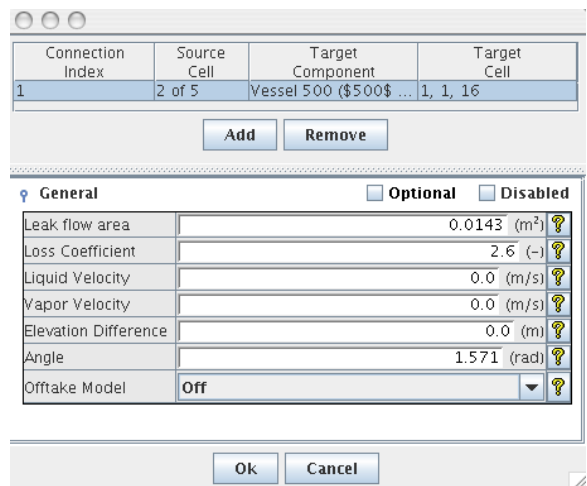


Figure 6.33. Pipe Leak Path

6.6. Plenums

A plenum component is used to connect multiple components together at a single location. Plenum components consist of a single volume with multiple junctions. Each plenum junction allows the user to specify the effective momentum length as well as the connection face on the plenum. Unlike the pipe component, the geometry and initial conditions for the plenum are modified directly in the properties list of the plenum. The only custom editing dialog for the plenum is the connections dialog displayed below in [Figure 6.34, “Plenum Junctions Dialog”](#).

6.6.1. Plenum Junctions

The Plenum Junctions Dialog allows the user to create, edit and remove junctions between the plenum and other hydraulic components. Each junction details how the plenum is connected to a different hydraulic component. A source component and edge provide the information for the remote side of the connection. The local side details where on the plenum the component is connected. The effective momentum length is used in the calculation of elevation changes in the loop check model validation test.

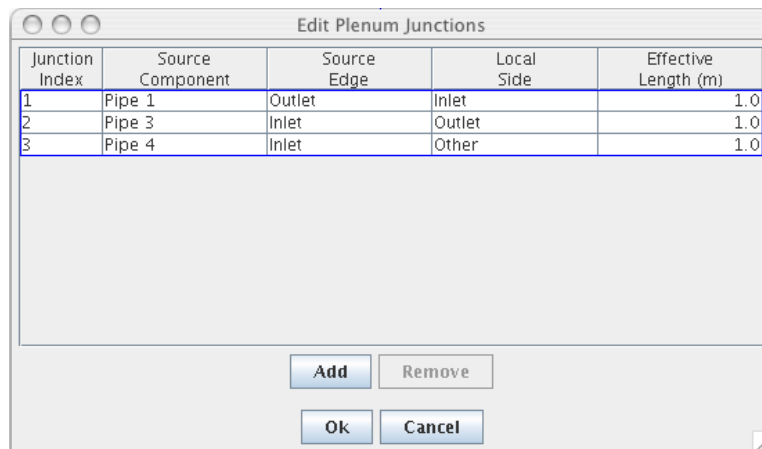


Figure 6.34. Plenum Junctions Dialog

6.7. Pumps

In TRACE, pump components are modeled as a pipe segment containing a pump cell. The pump cell is always the first cell. The standard pipe properties included in a pump are detailed in [Section 6.5, “Pipe”](#). These properties include geometry, friction, initial fluid conditions, etc.

The type of pump and its properties are modified directly in the property view. The properties needed to define a pump are based primarily on the *Pump Type (IPTY)*. The dialog that appears when creating a new pump is shown in [Figure 6.35, “Pump Completion Dialog”](#).

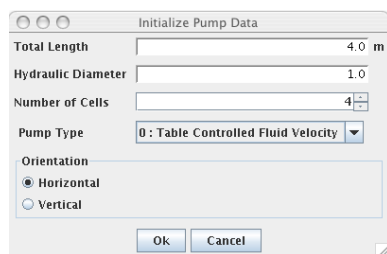


Figure 6.35. Pump Completion Dialog

6.7.1. Pump General Properties

The general pump properties are displayed below in [Figure 6.36, “General Pump Properties”](#). These properties are general pump properties that are included in most pumps. The *Pump Type (IPTY)* must also be specified in addition to the properties shown below.

Reverse Rotation	[0] No	?
Degredation Option	[1] Single and Two P...	?
Effective MOI	3460.0 (kg*m ²)	?
Use Alternative Inertia	<input type="radio"/> True <input checked="" type="radio"/> False	?
Pump Curve Option	[1] Built-in Semiscal...	?
Zero Order Efficiency	0.0 (-)	?
First Order Efficiency	0.0 (-)	?
Second Order Efficiency	0.0 (-)	?
Speed/Torque Var.	<none>	S ⁺ ?
Speed Trip	<none>	S ⁺ ?
Off Speed Controller	<none>	S ⁺ ?

Figure 6.36. General Pump Properties

6.7.2. Pump Speed Values

The speed values for the pump are located in the *Speed Values* attribute group shown below in Figure 6.37, “Pump Speed Properties”. The speed tables are not used in an electric motor pump, however the speed conditions and controllers must be entered.

Speed Values

Initial Speed	124.4 (rad/s)	?
Off Speed	0.0 (rad/s)	?
Maximum Speed Change	0.0 (rad/s)	?
Speed Scale Factor	0.0 (-)	?
Off Speed Controller	<none>	S ⁺ ?
Speed Table	Rows: 0 []	E ⁺ ?
Rate Factor Table	Rows: 0 []	E ⁺ ?

Figure 6.37. Pump Speed Properties

6.7.3. Pump Initial Fluid Conditions

The *Mass Flow Rate* property is the mass flow rate across the pump interface inlet. This property is only entered when the pump is used by a steady state controller of type 1. To enable this property, create a type 1 CSS, and select the pump as the reference component.

Pumps also include the same initial fluid conditions as a pipe component. Refer to Section 6.5, “Pipe” for information regarding pipe initial fluid conditions.

6.7.4. Pump Alternate Inertia

Pumps may use a set of alternate inertia data, enabled through the *Use Alternative Inertia* property in the general properties of a pump. The alternative inertia properties are displayed below in Figure 6.38, “Pump Alternative Inertia Properties”.

Alternate Inertia

Alternate Inertia	1.0 (kg*m ²)	?
Moment Coefficient A	2.0 (kg*m ²)	?
Moment Coefficient B	0.0 (kg*m ²)	?
Moment Coefficient C	0.0 (kg*m ²)	?
Alternate Speed Cuton	100.0 (rad/s)	?
Speed Dropped	[0] Greater Than OMTEST	?

Figure 6.38. Pump Alternative Inertia Properties

6.7.5. Pump Friction Factors

Pump components have additional friction factor data not present in other hydraulic components. This additional data represents the friction from the pumping action. The friction factors are displayed below in [Figure 6.39, “Pump Friction Factor Properties”](#).

Friction Factors

Frictional Torque 0	0.0 (Pa*m ³)	?
Frictional Torque 1	0.0 (Pa*m ³)	?
Frictional Torque 2	0.0 (Pa*m ³)	?
Frictional Torque 3	0.0 (Pa*m ³)	?
Torque Break Speed	0.0 (rad/s)	?
Low speed 0	0.0 (Pa*m ³)	?
Low speed 1	0.0 (Pa*m ³)	?
Low speed 2	0.0 (Pa*m ³)	?
Low speed 3	0.0 (Pa*m ³)	?

Figure 6.39. Pump Friction Factor Properties

6.7.6. Pump Rated Values

The rated values of a pump are the factory specifications appropriate for the make and model of the pump. These values are displayed below in [Figure 6.40, “Pump Rated Values”](#).

Rated Values

Rated Head	843.0 (m ² /s ²)	?
Rated Torque	4.285E4 (Pa*m ³)	?
Rated Volumetric Flow	5.58 (m ³ /s)	?
Rated Density	1000.0 (kg/m ³)	?
Rated Speed	124.4 (rad/s)	?

Figure 6.40. Pump Rated Values

6.7.7. Pump Homologous Curves

The *Degradation Option* and *Pump Curve Option* determine which pump-curve properties are required. The pump-curve properties are displayed below in [Figure 6.41, “Pump Curve](#)

Properties”. The pump-head and torque curves must be entered for each quadrant, while the degradation multiplier tables have a single table applied to all values.

Curves

Pump Head Multiplier	<none>		
Single-phase Head	Q1: 0, Q2: 0, Q3: 0...		
Fully-degraded Head	Q1: 0, Q2: 0, Q3: 0...		
Single-phase Torque	Q1: 0, Q2: 0, Q3: 0...		
Fully-degraded Torque	Q1: 0, Q2: 0, Q3: 0...		
Head-degradation Mult.	Rows: 0 []		
Torque-degradation Mult.	Rows: 0 []		

Figure 6.41. Pump Curve Properties

The pump-head and torque tables are modified in a custom dialog that allows the user to select the quadrant and enter the data for that quadrant as a regular table. The column headers for these tables display the unit label for the selected quadrant. The custom pump curve dialog is shown below in Figure 6.42, “Pump Curve Dialog”.

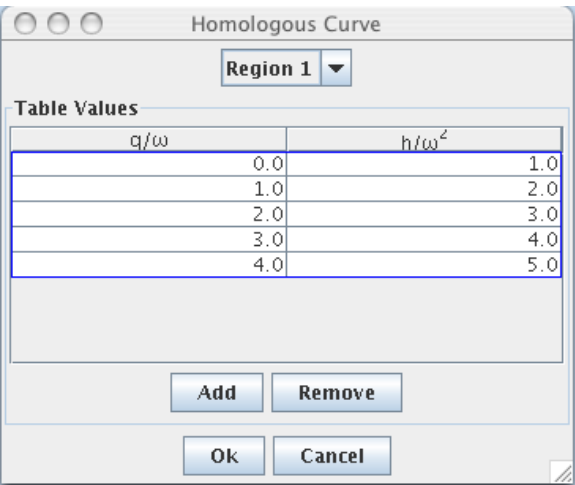


Figure 6.42. Pump Curve Dialog

6.7.8. Pump Control Connections

Control system components are used for many of the tables and properties inside pumps. These control connections may be entered directly in the property view from a component selection dialog, or they may be established using the connect tool on 2D Views. In the center of the pump node there is a connection point for control system components.

To use the connect tool to establish these connections, click on the output node of a control system component, then click on the center connection point inside the pump. A dialog will open displaying all of the available connections for the pump as shown in Figure 6.43, “Pump Control Input Connections”. After selecting a control input, pressing the **OK** button will establish the connection.

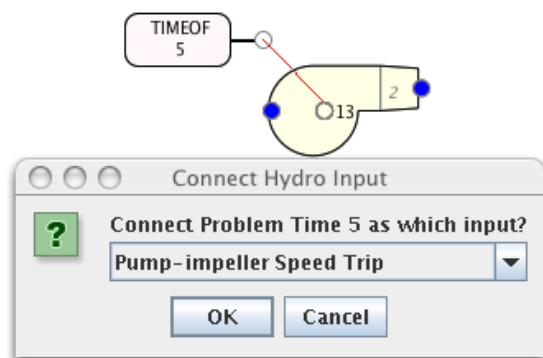


Figure 6.43. Pump Control Input Connections

6.8. Prizers

In TRACE, a PRIZER component is available for pressurizer modeling. The standard hydraulic component properties included in a PRIZER are the same as for a PIPE and are detailed in [Section 6.5, “Pipe”](#). These properties include geometry, friction, etc.

The completion dialog that appears when creating a new pressurizer inside a TRACE model is displayed below in [Figure 6.44, “Pressurizer Completion Dialog”](#). Like other completion dialogs, this dialog will not be displayed if the *Use Completion Dialogs* preference is set to *False*.

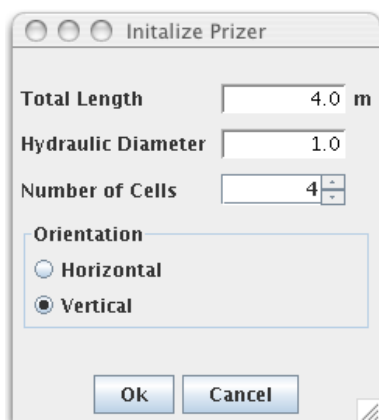


Figure 6.44. Pressurizer Completion Dialog

6.8.1. Prizer Properties

There are four additional properties for the pressurizer component. These properties are displayed below in [Figure 6.45, “Prizer Properties”](#). *Heater Power* is the total power from the built in heater. The *Pressure Setpoint* is the point at which the sprayers inside the pressurizer activate. The *Pressure Differential* is the differential at which the heater and sprayers are at maximum power. The water level that results in the heater turning off is entered in the *Heater Cutoff Level*.

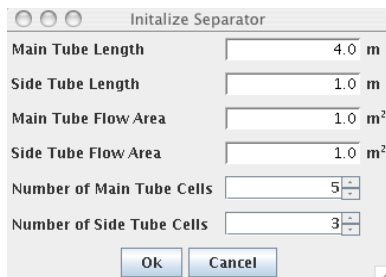
Heater Power	1.7E6 (W)	?
Pressure Setpoint	1.55E7 (Pa)	?
Pressure Differential	2.0E5 (Pa)	?
Heater Cutoff Level	3.0 (m)	?

Figure 6.45. Prizer Properties

6.9. Separator

The separator component is a volumetric component with a main and side tube with similar geometry and initial condition properties to Tee components. The main tube represents the separator itself, with the inlet supplying mixed water and steam, and the outlet releasing wet steam. The side-tube models the water drain from the separator. The standard hydraulic component properties included in a Separator are detailed in [Section 6.5, “Pipe”](#). These properties include geometry, friction, etc. The Tee properties that a separator must implement are detailed in [Section 6.11.1, “Tee Properties”](#) below.

The completion dialog that appears when creating a new Separator inside a TRACE model is displayed below in [Figure 6.46, “Separator Completion Dialog”](#). Like other completion dialogs, this dialog will not be displayed if the *Use Completion Dialogs* preference is set to *False*.



The dialog box is titled "Initialize Separator". It contains the following fields and controls:

- Main Tube Length: 4.0 m
- Side Tube Length: 1.0 m
- Main Tube Flow Area: 1.0 m²
- Side Tube Flow Area: 1.0 m²
- Number of Main Tube Cells: 5 (with up/down arrows)
- Number of Side Tube Cells: 3 (with up/down arrows)
- Buttons: Ok, Cancel

Figure 6.46. Separator Completion Dialog

6.9.1. Separator Properties

The unique properties required for defining a separator are displayed below in [Figure 6.47, “Separator Properties”](#).

Separator Count	1	?
Dryer Count	0	?
Carryover	Unknown (-)	?
Carryunder	Unknown (-)	?
Minimum Barrel Void	Unknown (-)	?
Maximum Barrel Void	Unknown (-)	?

Figure 6.47. Separator Properties

6.10. Single Junctions

Single junction components are hydraulic components with no volume. They represent a single flow path between hydraulic components with all of the flow and geometry properties of an

edge inside a pipe component. These were implemented as part of the effort to convert RELAP5 models to TRACE. Unlike other hydraulic components, the single junction does not open a completion dialog. The standard hydraulic component properties included in a Single Junction are detailed in [Section 6.5, “Pipe”](#). These properties include geometry and fluid flow. The single junction does not contain any volume properties.

6.11. Tees

In TRACE, Tee components can be used to model branching flow paths. The standard hydraulic component properties included in a Tee are detailed in [Section 6.5, “Pipe”](#). These properties include geometry, friction, etc.

The Tee component contains two fluid segments: the main tube and the side tube. The main tube represents the primary segment, while the side tube is the segment that intersects the main tube.

The completion dialog that appears when creating a new Tee inside a TRACE model is displayed below in [Figure 6.48, “Tee Completion”](#). Like other completion dialogs, this dialog will not be displayed if the *Use Completion Dialogs* preference is set to *False*.

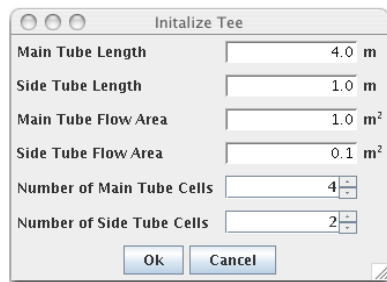


Figure 6.48. Tee Completion

6.11.1. Tee Properties

[Figure 6.49, “Tee Properties”](#) below displays the additional properties required for a Tee component. These properties detail how the side tube connects to the main tube. The *Side Tube Junction Index* defines the main tube cell where the side tube is connected. The *Zero Out Momentum* flag disables the momentum term in the motion equations. *Side Tube Junction Cosine* is the cosine of the horizontal angle between the main tube and the side tube, which is used in the momentum equations as well as the 3D display.

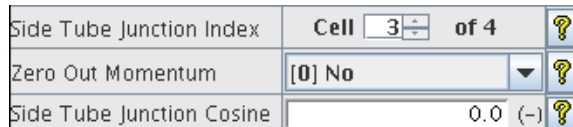


Figure 6.49. Tee Properties

6.12. Turbines

Turbines are modeled in TRACE as a volumetric component with a main and side tube with similar geometry and initial condition properties to Tee components. The main tube represents

flow path through the turbine. The side-tube models the water drain down to the feedwater heaters. The standard hydraulic component properties included in a turbine are detailed in [Section 6.5, “Pipe”](#). These properties include geometry, friction, etc. The Tee properties that must be defined for a turbine are detailed in [Section 6.11.1, “Tee Properties”](#).

The completion dialog that appears when creating a new Turbine inside a TRACE model is displayed in [Figure 6.50, “Turbine Completion Dialog”](#). Like other completion dialogs, this dialog will not be displayed if the *Use Completion Dialogs* preference is set to false.

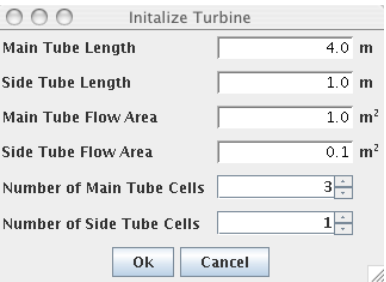


Figure 6.50. Turbine Completion Dialog

6.12.1. Turbine Properties

[Figure 6.51, “Turbine Properties”](#) below displays the additional properties required for defining a Turbine component.

Turbine Efficiency	Unknown (-)	?
Separator Efficiency	Unknown (-)	?
Initial Rotor Vel	Unknown (rad/s)	?
Rotor Inertia	Unknown (kg*m²)	?
Rated Massflow	Unknown (kg/s)	?
Sidearm Separation Model	[0] No TRAC-B Sidearm Model	?
Number of Lumped Stages	0	?
Rotor Number	0	?
Saturation Flag	0	?
Nozzel Area Option	0	?
Rated Angular Vel	Unknown (rad/s)	?
Windage Friction	Unknown (-)	?
Rated Torque	Unknown (Pa*m³)	?

Figure 6.51. Turbine Properties

6.13. Valves

In TRACE, Valve components are modeled as a pipe segment containing a valve edge. The standard pipe properties included in a Valve are detailed in [Section 6.5, “Pipe”](#). These properties include geometry, friction, initial fluid conditions, etc. Valve components are used to control flow through the model. In general these are used to simulate one or more valves inside the plant, however they can be used to activate the transient.

[Figure 6.52, “Valve Completion Dialog”](#) below displays the completion dialog for a valve. Like other completion dialogs, this dialog will not be displayed if the *Use Completion Dialogs* preference is set to *False*.

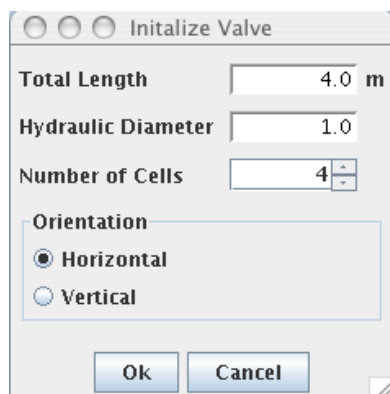


Figure 6.52. Valve Completion Dialog

6.13.1. Valve properties

The valve properties are primarily dependent on the **Valve Type (IVTY)** selected. [Figure 6.53](#), “**Valve Properties**” below displays the properties of a sample valve. The **Valve Interface Index (IVPS)** is the index of the edge inside the valve where the interface is located.

Note The Valve interface may only be an external edge if that edge is connected to a Break component.

Valve Interface Index	Edge <input type="text" value="2"/> of 3	?
Flow Area Adjustment Type	Flow Area Fraction per Second	?
Maximum Valve Rate	1.0E5 (1/s)	?
Off Adjustment Rate	0.0 (1/s)	?
Minimum Position	0.0 (-)	?
Maximum Position	0.0 (-)	?
Valve Flow Area	0.1521 (m ²)	?
Valve Hydro Diameter	0.254 (m)	?
Initial Fraction	0.0 (-)	?
Valve Stem Position	0.0 (-)	?
First Adjustment Table	Rows: 2 [0.0,0.0],[1.0E-5,1.0]	E?
Second Valve Table	Rows: 0 []	E?
Rate Factor Table	Rows: 0 []	E?
Valve Table Indep. Var.	Problem Time 1	S?
Valve Trip	Trip 11	S?
Override Trip	<none>	S?

Figure 6.53. Valve Properties

6.13.2. Valve Control Connections

Control system components are used for many of the tables and properties inside valves. These control connections may be entered directly in the property view, or they may be established using the connect tool on a 2D View. A connection point is provided on the valve stem for control system inputs.

To use the connect tool to establish these connections, click on the output node of a control system component, then click on the valve stem connection point on the valve. A dialog will

open displaying all of the available connections for the valve as shown in [Figure 6.54, “Valve Hydro Input Dialog”](#). After selecting a control input, pressing the **OK** button will establish the connection.

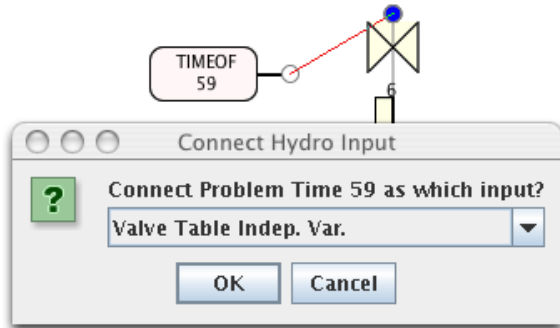


Figure 6.54. Valve Hydro Input Dialog

6.14. Jet Pump

The Jet Pump component is a volumetric component with a main and side tube with similar geometry and initial condition properties to Tee components. The main tube represents the pump volume. The side-tube models the jet source. The standard hydraulic component properties included in a Jet Pump are detailed in [Section 6.5, “Pipe”](#). These properties include geometry, friction, etc. The Tee properties that a Jet Pump must implement are detailed in [Section 6.11.1, “Tee Properties”](#) above.

The completion dialog that appears when creating a new jet pump is displayed below in [Figure 6.55, “Jet Pump Completion Dialog”](#). Like other completion dialogs, this dialog will not be displayed if the *Use Completion Dialogs* preference is set to *False*.

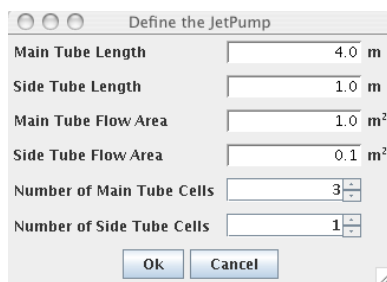


Figure 6.55. Jet Pump Completion Dialog

6.14.1. Jet Pump Properties

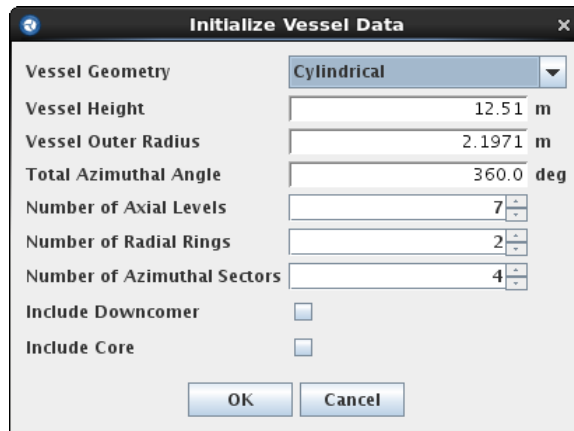
The additional properties of a jet pump that must be defined are displayed below in [Figure 6.56, “Jet Pump Properties”](#). These properties appear in the property view below all the general properties of the jet pump.

Offtake Model Option	Off	?
Number Of Jetpumps	0	?
Forward Diffuser Formloss	5.5 (-)	?
Reverse Diffuser Formloss	0.38 (-)	?
Forward Nozzle Formloss	5.5 (-)	?
Reverse Nozzle Formloss	0.38 (-)	?
Suction Formloss	0.04 (-)	?
Discharge Formloss	0.45 (-)	?

Figure 6.56. Jet Pump Properties

6.15. Vessels

The Vessel component is the 3D hydraulic component for TRACE. Vessels are either rectangular or cylindrical components broken down into axial levels. Each axial level has a planar array of cells organized first by the Y or Azimuthal axis, and then the X or Radial axis. Unlike 1D hydraulic components, Vessel cells are referred to by a compound number combining the axial level, and the planar cell index. [Figure 6.57, “Vessel Completion Dialog”](#) below displays the completion dialog for a vessel.



The dialog box is titled "Initialize Vessel Data". It contains the following fields and controls:

- Vessel Geometry:** A dropdown menu set to "Cylindrical".
- Vessel Height:** A text box containing "12.51" with a unit "m" button.
- Vessel Outer Radius:** A text box containing "2.1971" with a unit "m" button.
- Total Azimuthal Angle:** A text box containing "360.0" with a unit "deg" button.
- Number of Axial Levels:** A spin box set to "7".
- Number of Radial Rings:** A spin box set to "2".
- Number of Azimuthal Sectors:** A spin box set to "4".
- Include Downcomer:** An unchecked checkbox.
- Include Core:** An unchecked checkbox.
- Buttons:** "OK" and "Cancel" buttons at the bottom.

Figure 6.57. Vessel Completion Dialog

6.15.1. Vessel Geometry

Vessels are 3D components that may be defined in cylindrical or Cartesian coordinates. The decision of coordinate system must be made from the completion dialog, and cannot be changed after the dialog is exited. As with 1D components, the nodalization is initialized during creation and can only be modified through the renodalization wizard described in [Section 16.2, “3D Hydraulic”](#). The geometry of the existing nodalization may be modified through the Geometry and Connections dialog shown below in [Figure 6.58, “Vessel Geometry”](#).

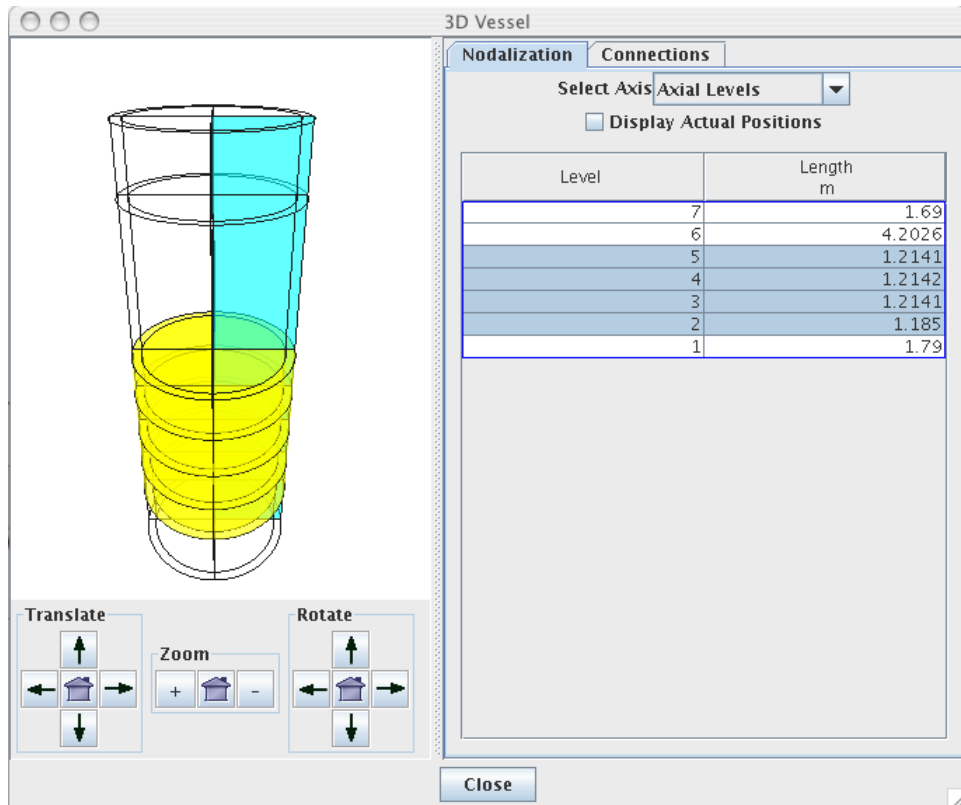


Figure 6.58. Vessel Geometry

This dialog allows the user to change the dimension of selected nodes. The selected nodes are displayed in the 3D representation of the vessel on the left hand side. The table on the right side of the dialog may display lengths in absolute length, or by relative length. Relative lengths must be displayed for User Defined Numerics to operate correctly.

6.15.2. Volumetric and Edge Data

The volume and edge properties may be modified through a single custom dialog. This dialog displays a 2D representation of the vessel from the side, and top-down. Cell and Edge selection may be done either by selecting nodes in the table, or by selecting nodes in the dialog. The volume portion of the dialog is displayed below in [Figure 6.59, "Vessel Cell Properties"](#). The editing property is selected from the list at the center of the dialog and the current units are displayed to the immediate right of the list entries.

Note *Volume Fraction* is a derived value which does not support user defined numerics.

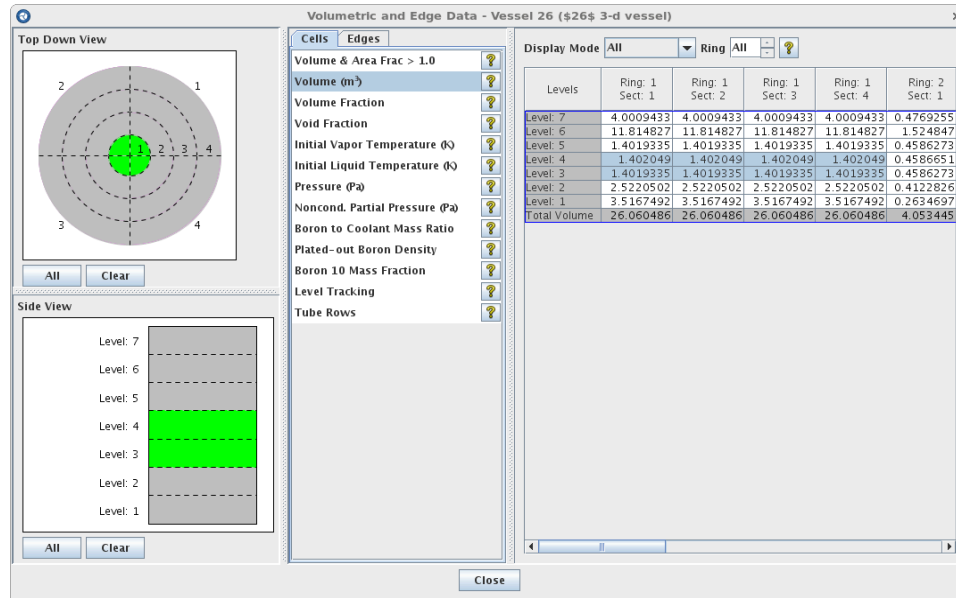


Figure 6.59. Vessel Cell Properties

The cell properties in the dialog also allow specifying which portion of the vessel to display. These display modes (when available) include the entire vessel, the downcomer, and the core region. When one of these display modes are selected, the included rows and columns in the cell table will be only those appropriate for the given selection. In addition to the global display mode table cell filtering, specific ring/x locations can be selected through the provided spinner element. When a ring or x value is specified, only the data for that ring/x value will be displayed in the table. It also should be noted that the ring/x value spinner will only include the rings/x values which are appropriate for the display mode selected (ie if the downcomer is not contained in radial level 1, radial level 1 will not be selectable from the ring selection spinner). Finally, for the cell volume property, volume totals will be displayed for each row and column in the table. If the core or downcomer is selected, the provided totals will assume values of 0.0 for the locations in the vessel which do not include the selected core/downcomer.

Figure 6.60. Vessel Cell Properties Display Filter

The edge data is defined in the same dialog by selecting the edge tab. The edge properties are also restricted by axis. An additional combo-box in the edge panel allows the user to select the current edge. The graphical representation of the vessel reflects this change by selecting the appropriate edge, as shown with the azimuthal edge selection below in [Figure 6.61, “Vessel Edge Properties”](#).

Note *Flow Area Fraction* is a calculated value which does not support user defined numerics.

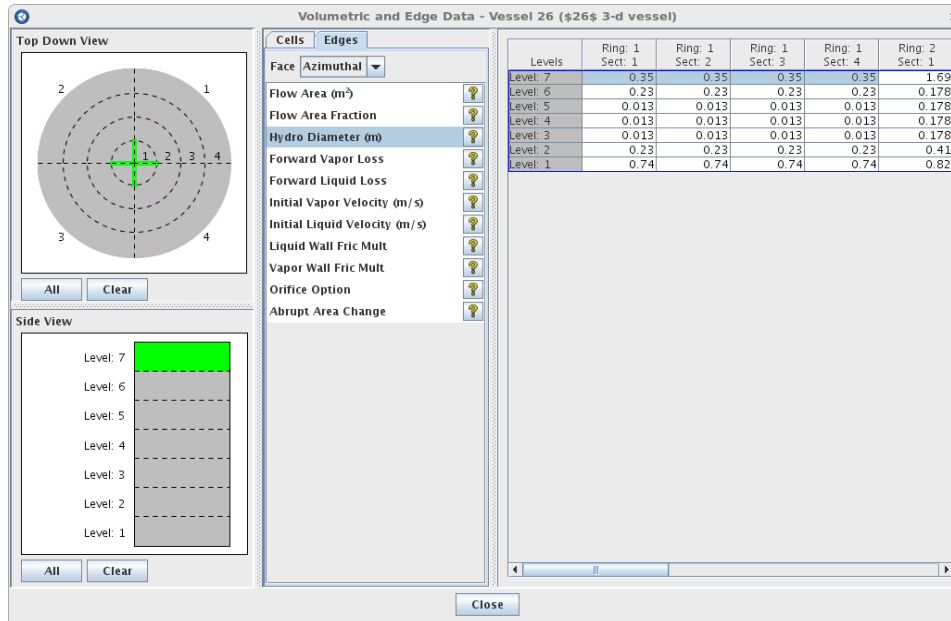


Figure 6.61. Vessel Edge Properties

6.15.3. Vessel Boundary Interfaces

The node and interface indexes for various structures inside vessels are located in the Boundary Interfaces attribute group displayed below in [Figure 6.62, “Vessel Boundary Interface Properties”](#). The layer and interface selection spinners are limited by the available nodalization of the vessel.

Boundary Interfaces

Includes Downcomer	<input checked="" type="radio"/> True <input type="radio"/> False	?
Downcomer Upper Interface	6	?
Downcomer Lower Interface	2	?
Downcomer Inner Radius	1	?
Includes Core	<input checked="" type="radio"/> True <input type="radio"/> False	?
Core Upper Interface	5	?
Core Lower Interface	2	?
Core Outer Radius	1	?
Lower Support Plate	0	?
Upper Support Plate	0	?
Upper-head Support Plate	0	?

Figure 6.62. Vessel Boundary Interface Properties

6.16. Vessel Junction Components

The vessel junction component manages the input for a set of individual single junction components connecting one vessel to another. It provides tools to facilitate creating and managing those single-junctions. It includes a user-interface that identifies the locations of the single-junctions and allows the user to specify the properties for individual single-junctions contained within the connection. A drawn representation is provided in order to be visualized

in 2D views. The component includes I/O routines for export and import of the Vessel Junction Components.

6.16.1. Initialization

The Vessel Junction component can be created like any other hydraulic component. This is done either from the **New** menu item off of the Vessel -Vessel Junction right-click pop-up menu or using the 2D view component insertion tool. When the component is first placed a setup wizard is provided. This wizard will guide you through initialization of the vessel junction component and its internal single junctions. The following dialog is provided when a Vessel Junction component is first created:

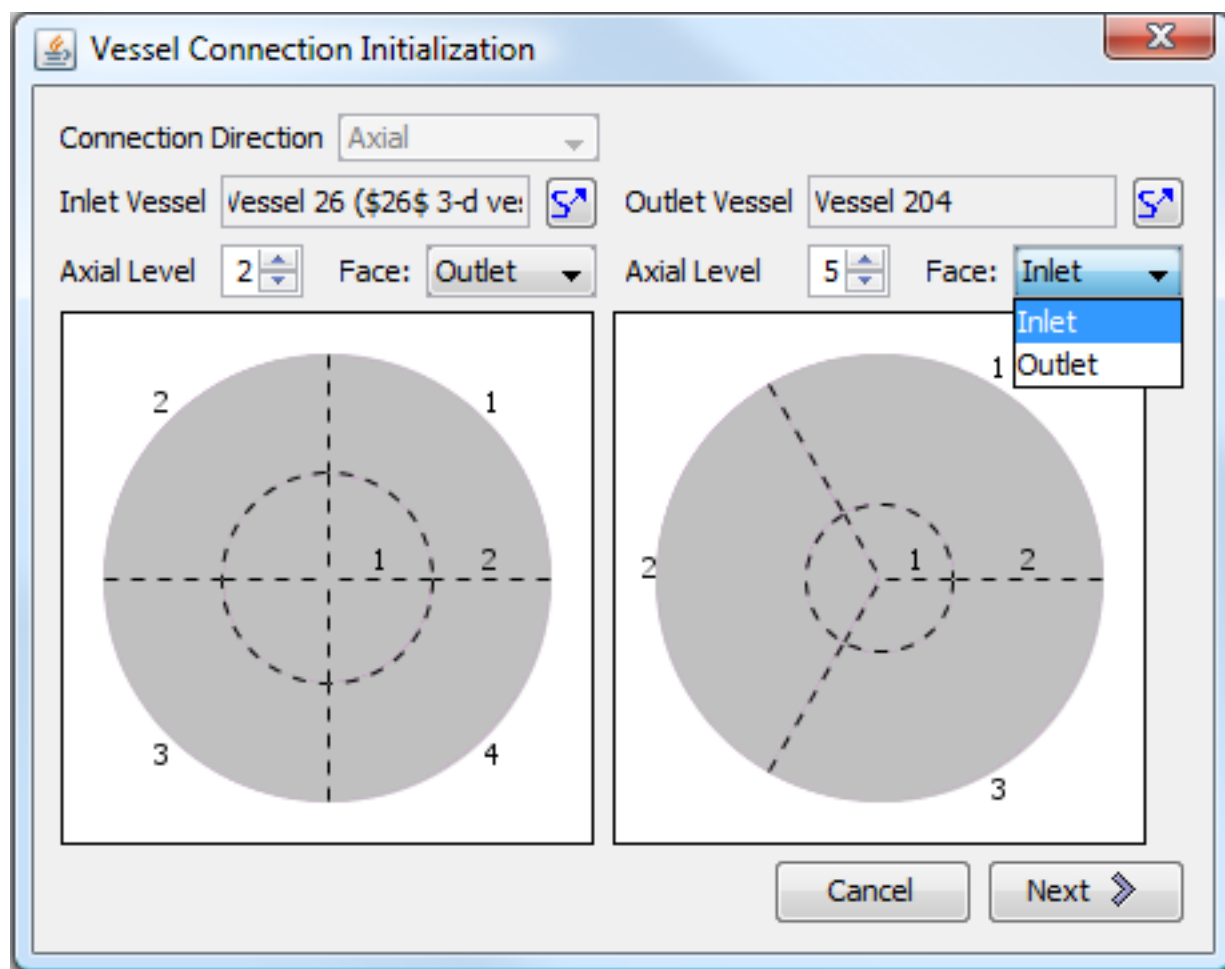


Figure 6.63. Vessel Connection Initialization Wizard

In this dialog the source vessel, target vessel, and connection direction, axial positions, and faces are specified. A drawn representation of the planar views for each selected vessel will be provided. Once the connection information is specified the **Next** button will be enabled. In the case where the selected vessels are using the same coordinate system (Cartesian to Cartesian or cylindrical to cylindrical), the wizard will generate a series of junctions for each of the intersecting areas between the vessels (except on those locations where zero volume exists). The flows areas for the resulting junctions will be the flow area of the volume formed by the intersecting cells multiplied by the minimum volume fraction of the two adjoining cells. No

internal junctions will be created when vessels are selected which have different coordinate systems.

A vessel initialization configuration dialog is provided once the vessels have been selected and the internal junctions are initialized. This dialog provides a display of each of the vessels involved in the connection, add and remove junction buttons, and mapping functionality.

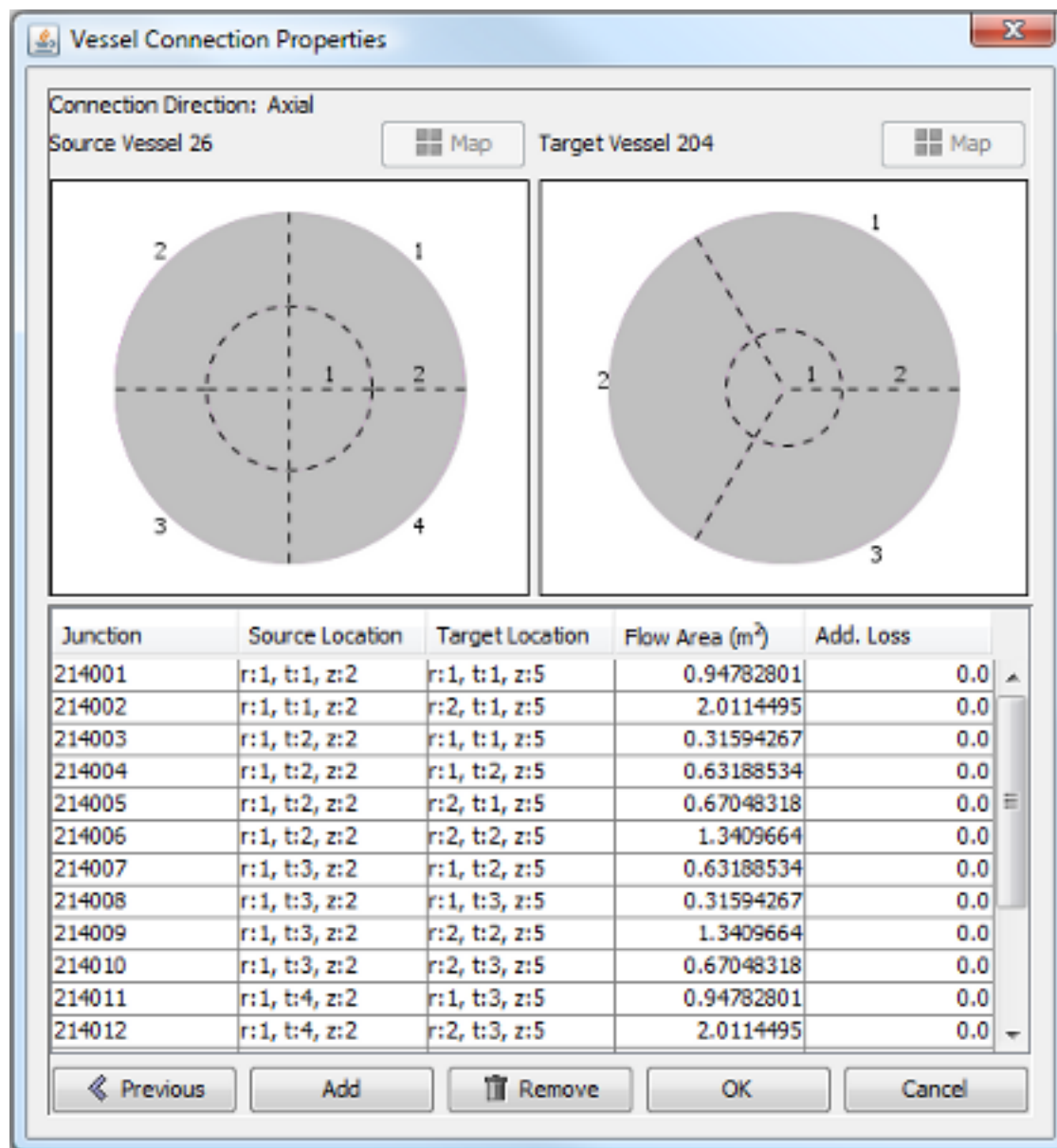


Figure 6.64. Vessel Connection Initialization Configuration Dialog

When a junction in the table is selected, the planar views will update to identify the cells the junction connects. When a cell on one of the planar views is selected the opposite planar view will identify the locations of junctions made to that vessel. Multiple cells can be selected by clicking and dragging across the planar views. Press and drag will add cells to the current selection. Control click will add or remove the selected cell from the selection. Junctions can be manually added to or removed from the list of junctions by selecting the add or remove button. Any newly

added junctions will be set to the first planar locations. Junction connection locations can be changed by selecting on one of the location values in the table and choosing the desired position from the provided planar cell selector.

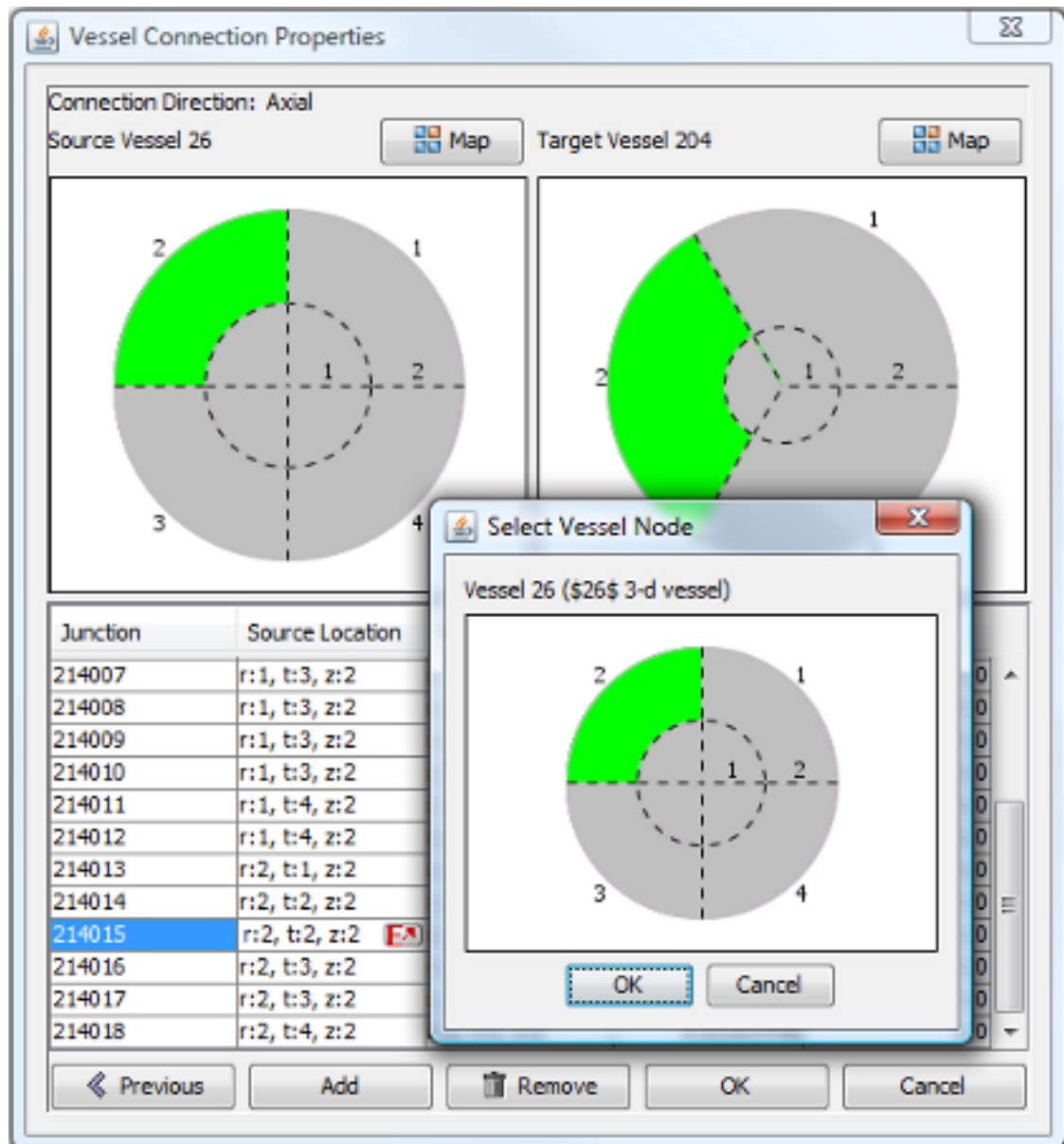


Figure 6.65. Vessel Planar Location Selector

A special mapping capability is provided to assist assigning internal junctions at the intersecting areas for vessels which share the same geometry. Selecting on a location(s) on one of the planar views will enable a mapping button. This button will display a planar mapping dialog. The planar mapping dialog provides a list of each of the locations on the opposite vessel which intersect at the selected vessel planar location(s). Any of the junctions which are already specified will be identified in the included column with a check mark. Checking values in the include column toggle whether or not the junction will be included in the main junction list. The selected junctions are added or removed from the main list with calculated flow areas once the mapping dialog is closed.

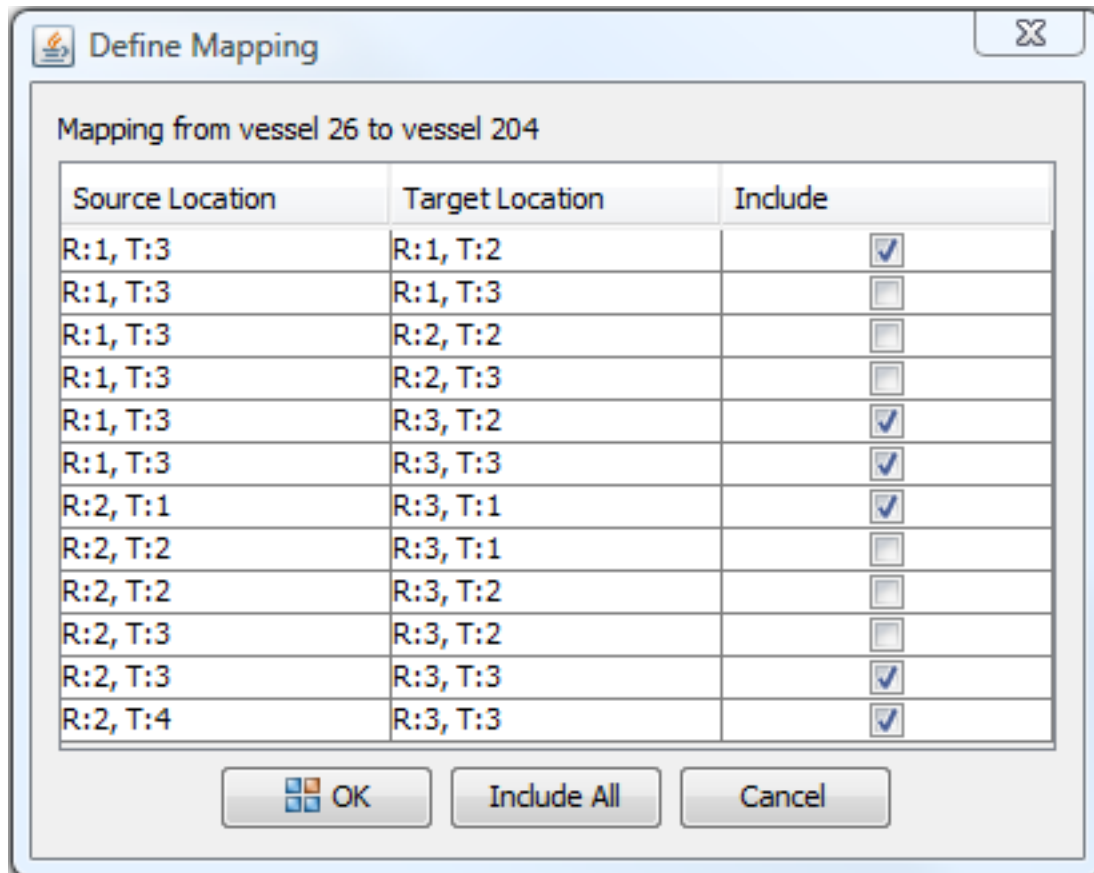


Figure 6.66. Vessel Mapping Planar Location Dialog

The junction initialization configuration table allows specification of some junction properties. Flow areas and loss coefficients are provided for each of the junctions. These values can be edited much like any other real value in the TRACE plug-in. K-factors or friction factors will be provided depending on the model IKFAC option. A reverse loss column will be provided when reverse loss coefficients are needed. Pressing the OK button on the dialog will add the new Vessel Junction Component to the model.

6.16.2. Editing

Properties for the vessel junction component can be specified through the vessel junction property editor. The property view for the vessel junction component includes fields for name, description, number, and edge data. The vessel junction component number should be a number unique to the existing hydraulic component numbers. The **Connection Edge Data** property provides the vessel junction property editor. This dialog allows a user to edit each of the individual junction properties for geometry, friction, and initial conditions. The layout of the dialog is very similar to the vessel initialization configuration dialog. Refer to the previous section for information on adding, removing, and mapping junctions. An edit mode combo box is provided at the bottom of the dialog. This includes options for editing the different property types. When a property type is selected, the table will refresh to include the properties which are associated with that type. Table cell editors are provided for editing each of the individual property values. Pressing the OK button will set the specified values on the vessel junction component and close the dialog.

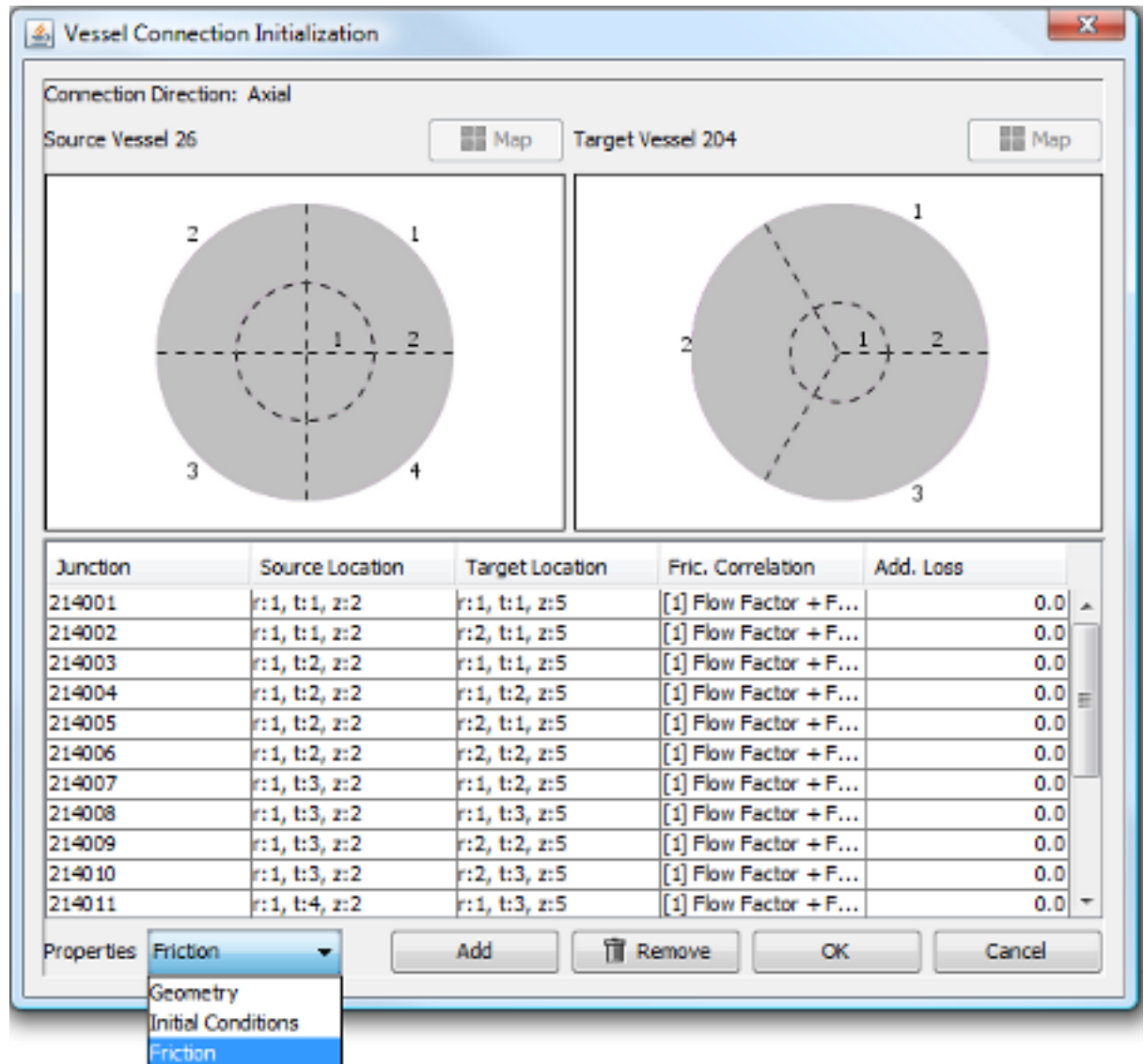


Figure 6.67. Vessel Junction Property Editor

6.16.3. Drawn Representation

A 2D drawn representation is provided for vessel junction components. It includes two connection points for connecting to a source and target vessel. The connection tool can be used to create connections from the drawn vessel junction to the target vessels. If one of the connections to a vessel is deleted or disconnected, the locations for the internal junctions on the disconnected side are cleared. If the vessel junction component is reconnected to a different vessel, the locations will need to be respecified to the proper planar locations.

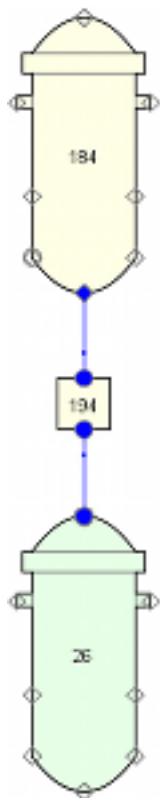


Figure 6.68. Drawn Vessel Junction Component

A location specification dialog is provided when the vessel junction connections are made using the connect tool. This dialog allows the specification of the axial location and face to connect to on the selected vessel. A connection point will be placed at the desired location when the vessels are drawn in the nodalized format.

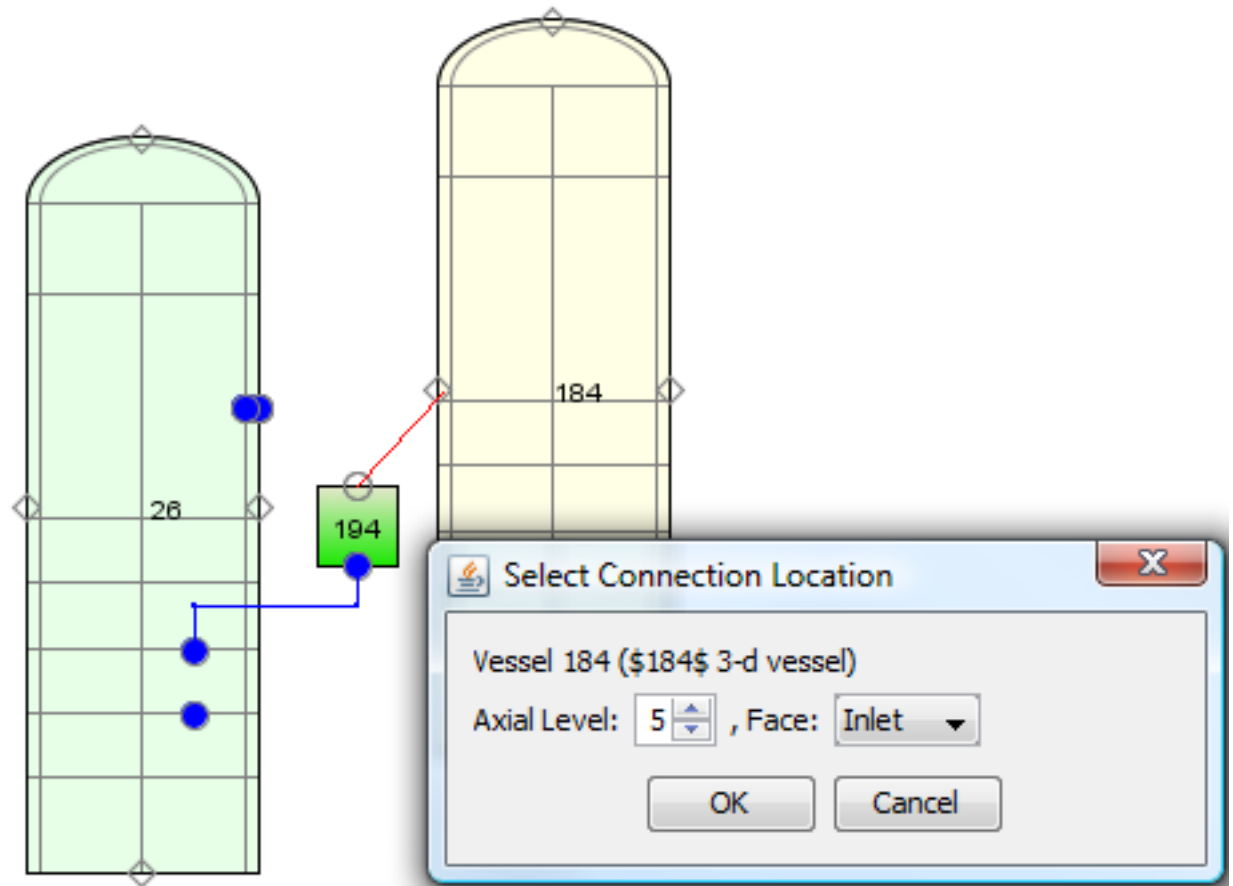


Figure 6.69. Vessel Connection Location Dialog

Chapter 7. Control Systems

The control system of model is used to perform calculations and provide system inputs to allow feedback controls and user interaction to the model. The control system consists of signal variables, control blocks, and trips. The signal variables serve as inputs from the model. Control blocks provide a value based on manipulation of input data. Trips activate when an input value enters or exits a specified range.

7.1. Signal Variables

Signal variables serve as input obtained from elsewhere in the model. This may be a global property, such as calculation time, or a probed value from a specific location in a component such as the pressure in a cell of a pipe. The signal variables are further divided by the source of the property.

Note A TIME general signal variable is created automatically for all new TRACE models.

Signal variables have a common property called the *Parameter Type* or *ISVN*. This is the enumeration that defines what type of signal is to be returned. The remaining signal variable properties are enabled or disabled based on the selected type.

The *Behavior Mode* property defines how the value should be obtained from the selected input. [Table 7.1, “Signal Variable Behavior Modes”](#) lists the available behaviors. This property is encoded in the sign or value of the ISVN, ICN1 and ICN2. The encoding is provided automatically when the behavior is selected.

Encoding	Description
ISVN > 0, ICN1 > 0, ICN2 = 0	The exact value at the specified location.
ISVN < 0, ICN1 > 0, ICN2 = 0	The time difference of the value at the specified location.
ISVN < 0, ICN1 > 0, ICN2 > 0	The difference between the values at the specified locations.
ISVN > 0, ICN1 < 0, ICN2 < 0	The minimum of the values at the specified locations.
ISVN > 0, ICN1 > 0, ICN2 > 0	The maximum of the values at the specified locations.
ISVN > 0, ICN1 > 0, ICN2 < 0	The volume average of the values at the specified locations.

Table 7.1. Signal Variable Behavior Modes

7.1.1. General Signal Variables

General signal variables obtain a value from the global properties of the simulation. These variables have no *Signal* reference. Most general signal variable allow only the *Exact Value Behavior Mode*.

7.1.2. Component Signal Variable

Component signal variables obtain a specific value from a component, such as valve stem position, pump hydraulic torque or heat structure power reactivity. These signal variables require

a component reference but have no specific location. Since they have no ICN1 or ICN2 values they typically only allow the *Exact* or *Time Difference* behavior modes. The properties of a Component signal variable are displayed below in [Figure 7.1, “Component Signal Variable”](#).

The screenshot shows a dialog box titled "Pump Rotational Speed 5". It has a "General" tab and two checkboxes: "Optional" (checked) and "Disabled" (unchecked). Below these are several fields:

Component Name	unnamed	?
Signal-variable ID	5	?
Description	<none>	E* ?
Comments	<none>	E* ?
Parameter Type	[41] Pump Rotational Speed	?
Signal	Pump 4 (\$4\$ bkn-loop pump)	S* ?
Behavior Mode	Exact Value	?

Figure 7.1. Component Signal Variable

7.1.3. Volume Signal Variable

Volume signal variables are used to obtain a value from either a single cell or a range of cells inside a hydraulic component. These signal variables require a hydraulic component reference and one or two locations. The dialog for selecting the cells in a hydraulic component is displayed below in [Figure 7.2, “Volume Selection Dialog”](#). Pressing the [S*](#) button will open component selection dialog for all hydraulic components in the model that contain hydraulic cells. The editor will automatically determine whether one or two locations are required and will limit the selection to the number cells in the component.

The screenshot shows a dialog box titled "Signal Variable". It has a "Signal Source" section with a "Hydraulic" button and a text field containing "Tee 15 (\$15\$ int-loop c-leg & hpis/lpis)". To the right of the text field is a [S*](#) button. Below this are two sections for location selection:

First Location
Cell: 3 of 4

Second Location
Cell: 1 of 4

At the bottom are "Ok" and "Cancel" buttons.

Figure 7.2. Volume Selection Dialog

7.1.4. Edge Signal Variables

Edge signal variables provide access to the information across the edges inside a hydraulic component. Much like the Volume signal variables, the user must select a hydraulic component. Instead of specifying a volume the user must select an edge inside the component. The dialog for selecting the hydraulic component and edges is displayed below in [Figure 7.3, “Edge Selection Dialog”](#). Pressing the [S*](#) button opens the component selection dialog for all hydraulic components in the model that contain edges. The editor will automatically determine whether one or two locations are required and will limit the selection to the number edges in the component.

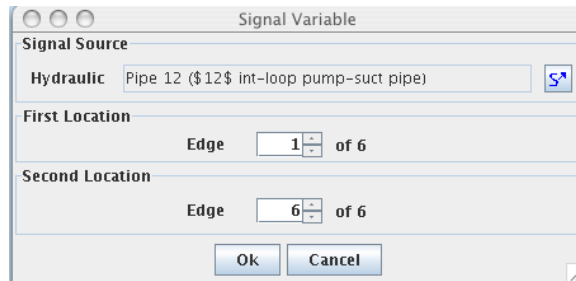


Figure 7.3. Edge Selection Dialog

7.1.5. Heat Signal Variables

Heat signal variables provide access to heat structure values. These values may be on individual rods, surfaces, or nodes inside the heat structure. The editing dialog displayed below in [Figure 7.4, “Heat Selection Dialog”](#), provides means for selecting a heatstructure and the locations on that heatstructure. Any hydraulic component that may have a built-in pipewall may be selected as a source for a heat signal variable. This includes pipes, tees, etc. The selection spinners allow the user to select a radial location, rod, and axial level.

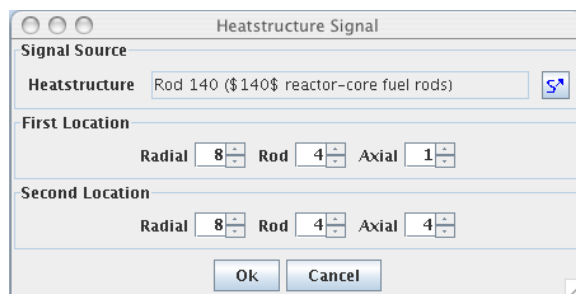


Figure 7.4. Heat Selection Dialog

7.1.6. Control Signal Variables

Signal variables can be used to report the current value of a control block, or trip component through the Control signal. The properties of a control signal variable are displayed below in [Figure 7.5, “Control Signal Variable Properties”](#).

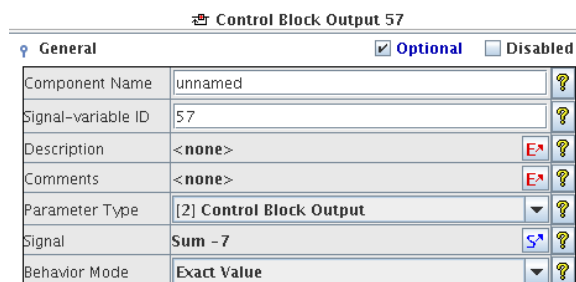


Figure 7.5. Control Signal Variable Properties

7.1.7. Power Signal Variables

Power signal variables are used to obtain power related values from either a heat structure or a Power component. These signal variables allow either the exact or time difference behaviour modes.

7.2. Control Blocks

A control block is a control system component that manipulates input data. Control blocks take input from signal variables and other control blocks, and output a floating point value. Input connections to the control blocks can be established by using the connect tool inside a 2D View. Simply use the connect tool to connect the output of the control system to the input of the control block.

The properties of a simple control block are displayed below in [Figure 7.6, “Control Block Properties”](#). The output units for a control block are specified through the units for the maximum and minimum values. **Constants One** and **Two** are used when needed to provide additional input for the control data. In general **Constant Two** serves as the initial value for the block.

Exponential - 1		
<input checked="" type="checkbox"/> General <input type="checkbox"/> Optional <input type="checkbox"/> Disabled		
Component Name	unnamed	?
Control Block ID	-1	?
Type	[60] Exponential	?
Description	<none>	E ?
Comments	<none>	E ?
Gain	1.0 (-)	?
Minimum	-1.0E20 (-)	?
Maximum	1.0E20 (-)	?
Constant 1	0.0 (-)	?
Constant 2	0.0 (-)	?

Figure 7.6. Control Block Properties

The control blocks are broken down into different sections depending on the operation of the control block. The tables below display the different ICBN values specified for each type of control block.

7.2.1. Arithmetic Blocks

Arithmetic control blocks handle basic mathematical functions. These blocks also handle functions with one, two or three independent variables.

Type (ICBN)	Description
9	A constant value.
14	Divides the first input by the second.
44	Performs a positive difference on the two input values.
47	Calculates a random number.
52	Determines the square root of the selected input.
54	Subtracts the second input from the first.
60	Raises the first input to a power.
102	A function of 1, 2 or 3 independent variables.
103	Adds any number of inputs together.
104	Multiplies any number of inputs together.

Table 7.2. Arithmetic Block Types

7.2.2. Calculus Blocks

Calculus control blocks handle more advance mathematical functions.

Type (ICBN)	Description
12	Calculates the derivative.
13	A limited double integrator.
17	Exponentiate the input.
23	A simple integration.
24	Integrate with Mode Control.
31	Limited Integrator
32	The Natural Log of the input.

Table 7.3. Calculus Block Types

7.2.3. Controller Blocks

Controller blocks contain control blocks that are typically used to control specific properties.

Type (ICBN)	Description
-9	Interactive Variable
22	Input switch
51	Second Order Transfer
200	PI Controller.
201	PID Controller.
202	Level Controller
203	Flow Controller
204	Pressure Controller

Table 7.4. Controller Block Types

7.2.4. Logical Control Blocks

Logical blocks represent simple logical comparisons.

Type (ICBN)	Description
5	Logical AND
15	Logical Exclusive Or
16	Logical Equivalence
18	Logical Flip-Flop
19	Gate
20	Greater Than or Equal To
21	Greater Than
25	Logical Inclusive Or
28	Logical Counter
33	Less Than or Equal To
34	Less Than
40	Logical Not And
41	Logical Not Equal
42	Logical Not Or
43	Logical Not or Negation

Table 7.5. Logic Block Types

7.2.5. Manipulation Blocks

Manipulation blocks are used for manipulating numerical values.

Type (ICBN)	Description
1	Absolute Value
4	Intergerizer
11	Dead Band, Zone, Space
29	Logical Input Switch
35	Maximum Value
36	Maximum During Transient
37	Minimum Value
38	Minimum During Transient
45	Quantizer
48	Sign Function
50	Sign Inversion
58	Variable Limiter
63	Trip Time
64	Minimum of Multiple Inputs
65	Maximum of Multiple Inputs
76	Pass Through

Table 7.6. Manipulation Block Types

7.2.6. Time Blocks

Time control blocks include all the control blocks that have to do with calculation time.

Type (ICBN)	Description
26	Lag
27	Logical Delay
30	Lead-Lag
46	Ramp
53	Step
61	0 Order Hold
62	Trip
77	Time of Change
100	Time Delay

Table 7.7. Time Block Types

7.2.7. Trigonometry Blocks

Trigonometry blocks perform trigonometric functions on their inputs.

Type (ICBN)	Description
2	Arcosine
6	Arcsine
7	Arctangent
8	Arctangent 2
10	Cosine
49	Sine
55	Tangent

Table 7.8. Trigonometry Block Types

7.2.8. Non-Standard Control Blocks

In general, there is a one-to-one ratio between control block types in SNAP, and TRACE control block numbers (ICBN). There are three exceptions however. TRACE provides multiple control block types for the Sum (5 types), Multiply (2 types) and Function (2 types) operations. These TRACE control block type numbers represented in the TRACE plug-in as a single set of Sum, Multiply and Function blocks. The following sections detail the specific differences for each type.

Sum Block

There are five different types of Sum blocks allowed by TRACE: Simple Sum (3), Constant Sum (56), Sum of Three (57), Weighted Sum (59), and Multiple Summer (103). Instead of providing these choices to the user, SNAP provides one Sum block, and then determines which type of Sum block is appropriate on export. For example if a Sum block has three input connections each with a weight of 1.0 such as the sample above, the export will report a Sum of Three (57). However if the user then enters a weight other than 1.0 for one of the input connections, the ASCII output would automatically convert the sum to a Multiple Summer (103).

Multiply Block

Similar to the Sum block, the Multiply block has a Multiple Multiply option (104) in addition to the Simple Multiplier (39). The TRACE plug-in only has one Multiply arithmetic block, but will export a Simple Multiplier (39) if the block has two or less input connections. Otherwise the block will be exported as a Multiple Multiplier (104).

Function Block

Function blocks lump together TRACE Function of One variable (101) and Function of Two or Three Variables (102). The difference between these two functions is the number of independent variables. When creating a new function block, the input connections must be created before the table data may be entered. Adding and removing connections after the table has been created will insert or remove the appropriate dependent variable value locations.

7.3. Trips

Trips are special control blocks that are used to indicate when an input value reaches a certain level or meets a specific criteria. Each trip is defined by single input signal, and logic that

determines what range of values correspond to "ON" from the signal source. The logic for determining the trips value is based on one or more Set Points, defined inside the *Setpoint Data* property. Each setpoint can have a setpoint factor table defined for it.

7.3.1. Trip Input Source

The Trip input source defines the control system component that supplies the input for the trip. This can be a control block, signal variable, trip controller, or trip signal expression as defined by the signal type. The input source can be defined through the property view *Input Source* editor, or by using the connect tool inside a 2D view.

7.3.2. Trip Signal Type

Trips are limited to specific input types. Once an input source has been defined, the *Signal Type* can only be changed if the new type supports the current input source. This means that a type 1 trip could be changed to a type 4 trip, but not to a type 2 trip. The signal types are listed below in [Table 7.9, "Trip Signal Types"](#).

Value	Description
1	Signal Variable, or Control Block
2	Trip Signal Expression
3	Trip Signal Controller
4	Simple Trip: Signal Variables or Control Blocks

Table 7.9. Trip Signal Types

7.3.3. Trip Setpoint Data

The setpoint data inside the Trip component is defined through the Setpoint Data dialog displayed below in [Figure , "Trip Setpoint Data"](#). The number of rows in this dialog is controlled by the drop-down list in the middle of the dialog. This list displays the singal range type (isrt) values available for the trip component. This dialog is where the setpoint values, units, delay time and setpoint factor tables are all defined for a Trip.

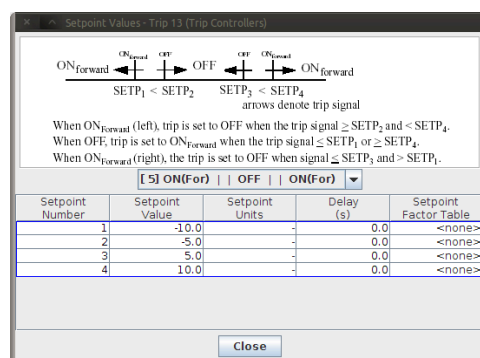


Figure . Trip Setpoint Data

7.3.4. Trip Signal Expressions

Trip signal expressions provide a shortcut for providing input for the Trip. Trip signal expressions behave as control components. The expression is made up of smaller sub-expressions and up to 5 constants. The properties of a signal expression are shown below in [Figure 7.8, “Trip Signal Expression Properties”](#).

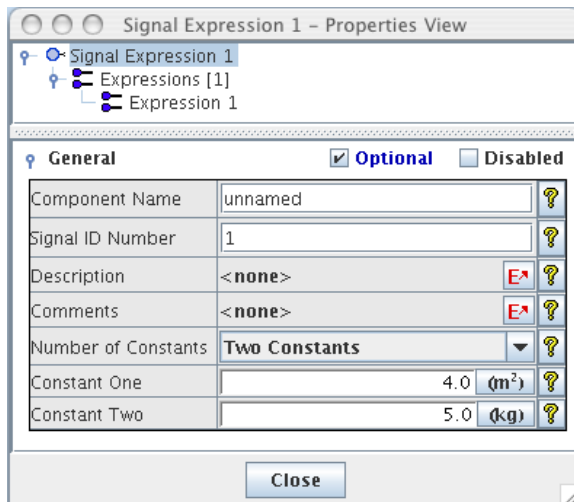


Figure 7.8. Trip Signal Expression Properties

The sub-expressions perform a simple arithmetic operation on two input variables. The input variables may be signal variables, control blocks, trips, constants or previously defined sub-expressions in the same signal expression. Sub-expressions can be found by expanding the expression in the navigator. The Sub expressions may be added or removed through the right-click pop-up menu shown below in [Figure 7.9, “Trip Signal Expression Pop-up Menu”](#).

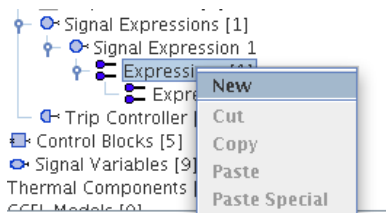


Figure 7.9. Trip Signal Expression Pop-up Menu

7.4. Control System Display Annotations

Control Systems added to a view may optionally display several annotations describing important properties. This behavior is toggled through the **Display Annotations** check box in the display element's right-click pop-up. The default value for new display components may be toggled in the TRACE plug-in preferences (see [Section 3.1, “TRACE Plug-in Preferences”](#)).

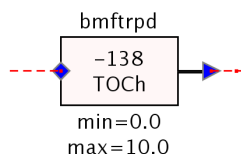


Figure 7.10. Display Annotations

Chapter 8. Thermal Components

Heat structures, custom user materials, and radiation enclosures are organized into the Thermal Components portion of TRACE. The Thermal category inside the navigator for a sample model is displayed in [Figure 8.1, “Thermal Components”](#). Heat structures provide thermal sources and sinks, and can be used to represent pipe-walls, reactor shrouds, fuel rods, etc. The user materials in TRACE allow the user to define custom thermal properties for heat transfer. Radiation enclosures model the rod-to-rod radiation of heat inside fuel channels.

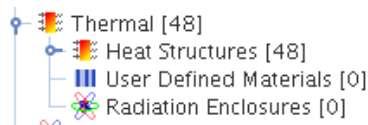


Figure 8.1. Thermal Components

8.1. Heat Structures

Heat structures represent rigid structures that can absorb, transfer, or radiate heat within the system. In TRACE, these heat structures are modeled with either a cylindrical, rectangular or spherical geometry. They can be used to represent pipe-walls, support structures, vessel containment, or fuel rods.

A heat structure consists of:

- radial data that remains constant along the axial length of the structure
- axial data to indicate surface area for heat transfer
- inner and outer radial boundary conditions
- the fuel information, when modeling fuel rods

8.1.1. Creating a Heatstructure

New heat structures may be created in three different ways. Heat structures may be created through the navigator in a similar fashion to other components. Simply right-click on the *Rod Heat Structures* or *Slab Heat Structures* nodes in the navigator and select the *New* option. Heat structures may also be inserted into a 2D view using the insert tool. When a new heat structure is created in either of these ways the dialog shown in [Figure 8.2, “Heat Structure Completion Dialog”](#) will be displayed. This dialog allows the user to initialize the heat structure's geometry prior to being added to the model.

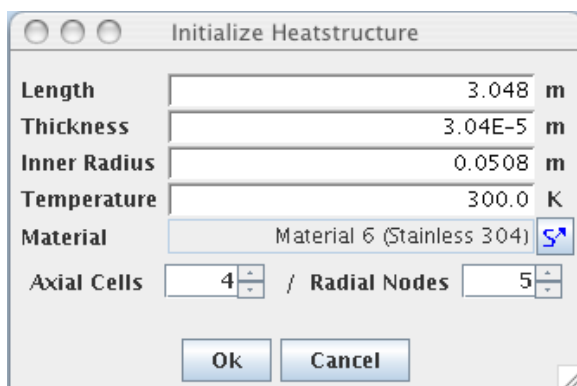


Figure 8.2. Heat Structure Completion Dialog

Rod or slab heat structures may also be created from the right-click pop-up menu of a 1D hydraulic component. This is a shortcut for producing pipe-wall heat structures. [Figure 8.3, “Pipe-wall Heat Structure Completion Dialog”](#) below displays the completion dialog for a new heat structure. The axial geometry for the heat structure nodes is derived from the connected pipe cells.

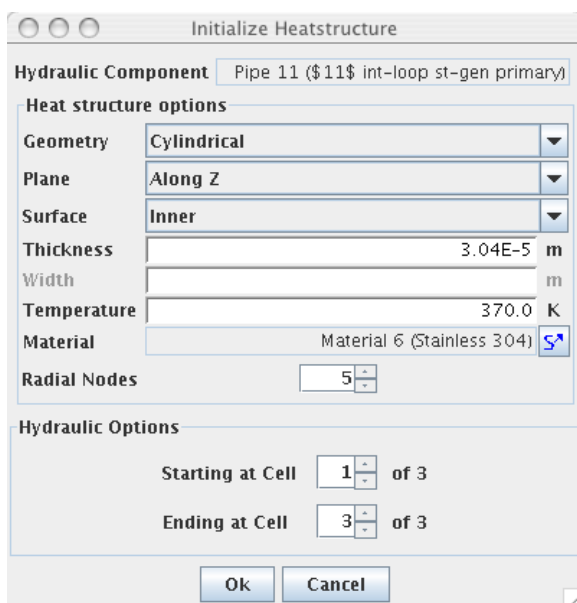


Figure 8.3. Pipe-wall Heat Structure Completion Dialog

8.1.2. Connecting A Heat Structure

Heatstructures are connected to hydraulic components at their surfaces. Once a heatstructure has been created, it can be connected to hydraulic components in three ways. First, the right-click pop-up menu for the heatstructure contains a **Attach To Hydro** option. Second, each surface can be modified individually through the *Axial Nodes / Surface BCs* heatstructure property. Finally, the connect tool can be used to connect a heatstructure to a hydraulic component inside a 2D View.

Connecting a heatstructure using the **Attach to Hydro** option or the connect tool will open the Connect to Hydraulic Component dialog shown in [Figure 8.4, “Heat Structure Attach To](#)

Hydraulic Dialog". This dialog allows the user to specify the hydraulic component connections to the heat structure. Checking the Renodalize option will adjust the axial nodalization of the heatstructure to match the selected components. Any existing connections will be preserved.

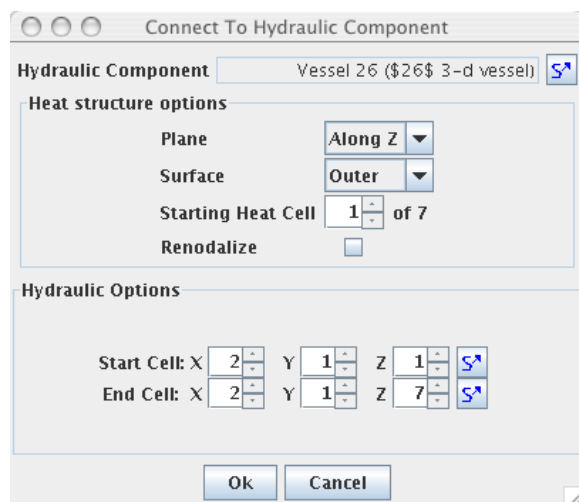


Figure 8.4. Heat Structure Attach To Hydraulic Dialog

Selecting the *Axial Nodes / Surface BCs* property opens the Surfaces dialog shown in [Figure 8.5, "Surfaces Dialog"](#). The combo-box at the top of this dialog can be used to select the average rod to edit. The selected rod's cells will be displayed in the table at the top of the dialog. Selecting cells in the first column of the table will display their properties in the bottom of the dialog. These properties include **Axial Length**, **Fine Mesh Count**, *Elevation*, and **Cell Orientation**. Selecting cells in the second or third column of the table will display the connections for those cells.

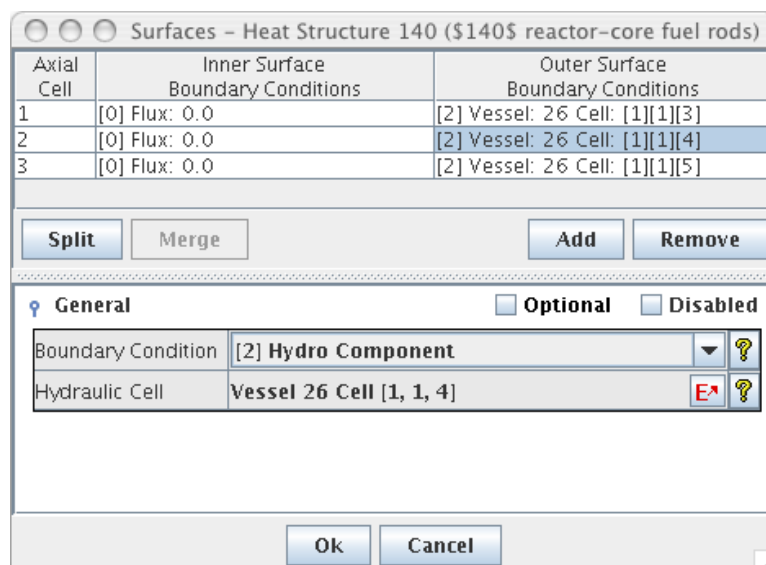


Figure 8.5. Surfaces Dialog

8.1.3. Radial Geometry

A heat structure's radial geometry is used to determine heat conduction between the inside and outside surfaces. The radial geometry may be modified in the Meshpoint dialog shown below in

Figure 8.6, “Heat Structure Radial Geometry Dialog”. There are three options for defining the radial mesh of heat structures: Standard Mesh, Finite Element, and Lumped Parameter. Both the standard mesh and finite element modes organize the thickness by regions of common material. Each region has a thickness and is further divided into sets of material intervals. The node points between material intervals are temperature nodes inside the heat structure. The finite-element conduction model defines a temperature node in the center of any non-gap material interval. Using the Lumped Parameter defines the heat structure with a single temperature node. Only one material region is allowed for lumped parameters.

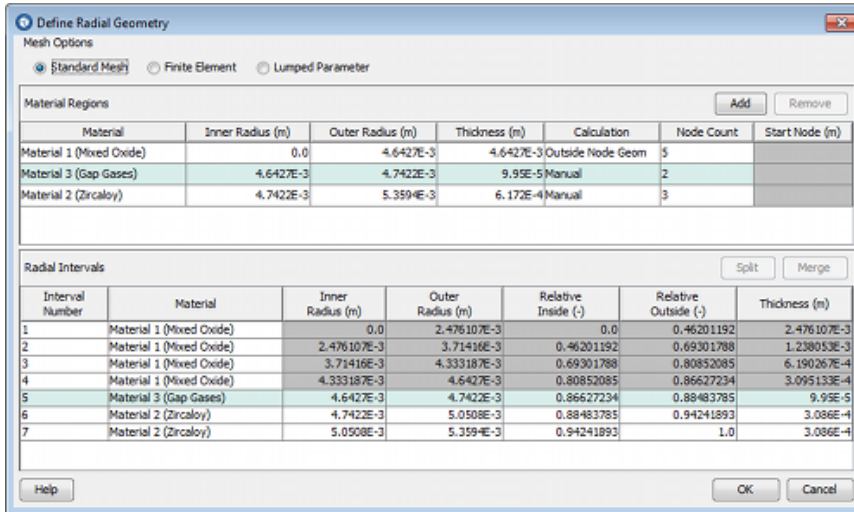


Figure 8.6. Heat Structure Radial Geometry Dialog

The top table in this dialog displays the material regions. Each region is displayed with an inside and outside radius, and a thickness value. Setting the inside radius of the first material region defines the inner radius of the heat structure. Every other row in the material region table is highlighted to ease readability. The Calculation column defines options for automatically calculating a material regions internal mesh intervals. The mesh intervals can be calculated as equally spaced, or a geometric series.

The geometric series calculation of the radial mesh defines the intervals in such a way that each interval is twice the thickness of the preceeding interval. The number of nodes inside the geometric series can be explicitly defined by selecting the Geometric Series By Nodes option. The Geometric Series By Size option calculates the number of intervals by finding the geometric series where the initial node is less than or equal to a user defined size.

The lower table in this dialog displays the different radial mesh intervals of all of the material regions combined. The highlighted rows in this table correspond to mesh intervals for material regions that are highlighted in the top table. This allows an easy way to identify which mesh intervals come from which material regions. Additionally the second column in the bottom table specifies the material for that interval.

Manually defined mesh intervals may be split into multiple intervals or joined together using the **Split** and **Merge** buttons. Only mesh intervals inside the same material region can be merged together. This dialog may result in a renodalization of the heat structure, further described in [Section 16.3.1, “Radial Renodalization”](#). Manually defined mesh intervals allow the user to change the size and location of mesh intervals by specifying the inside and outside radius, or the inner and outer relative positions or finally by specifying the thickness of a region.

Specifying the radius or relative position of a material region will modify the adjacent regions to preserve the thickness of the heat structure. Changing the thickness of a mesh interval will only modify that specific interval.

8.1.4. Built-in Materials

TRACE includes several commonly used materials that have well defined behaviours over a large range of temperatures. These are listed below, along with their corresponding material numbers in [Table 8.1, “Built-in Materials”](#).

Material Number	Name
1	Mixed Oxide
2	Zircaloy
3	Gap Gases
4	Boron-Nitride
5	Constantan/Nichrome
6	Stainless 304
7	Stainless 316
8	Stainless 347
9	Carbon A508
10	Inconel 718
11	Zircaloy Dioxide
12	Inconel 600

Table 8.1. Built-in Materials

8.2. User-Defined Materials

User Defined Materials may be used to specify a material that differs from the built-in materials. User defined materials have four properties that must be defined over a range of temperature: *Density*, *Specific Heat*, *Thermal Conductivity* and *Emissivity*. These properties may either be defined in a table, or as a functional fit table. Any property defined using the functional fit tables is not included in the primary material table. User defined materials may be used as the radial material for any hydraulic component's pipe-wall or for radial regions within a heatstructure.

8.3. Radiation Enclosures

Radiation between heat structures is handled by implementing a Radiation Enclosure component. The enclosure component consists of a list of heat structures, organized by parallel axial levels, and tables defining the view factor and beam lengths between heat structure cells. This data is very similar to the enclosure data embedded inside Chan components.

Chapter 9. Power Components

There are two types of components that insert power into the model. The Power component handles the kinetics and neutronics calculations for a modeling the reactor. Fluid power components are used to inject power directly into the fluid of hydraulic components.

9.1. Power

All of the kinetics and neutronics information inside a TRACE model is handled by the Power component. This component allows the user to support reactor feedback to multiple hydraulic and heat components in one location. A TRACE model may have more than one power component that operate independently to provide power for the system.

9.1.1. Power Properties

When defining a power component inside a TRACE model, the **IPWTY** flag indicates the type of power being defined. The values for this flag are listed below in [Table 9.1, “Power Types”](#).

Power Type Value	Name
1	Constant REACT Pogrammed Reactivity
2	Table Lookup of Programmed Reactivity
3	Initial Zero Programmed Reactivity
4	Initial constant RECT programmed Reactivity
5	Constant Reactor Core Power
6	Table look-up of Reactor Core Power
7	Trip-Initiated Table lookup of Reactor Core Power

Table 9.1. Power Types

Independent of type, the power is inserted either into heat structure nodes, or the fuel rods of a CHAN component. The selection is based off the **Target Component Type** property. Selecting which components are powered is done through the Powered Components property. The dialog that opens when selecting powered components is shown below in [Figure 9.1, “Powered Components Dialog”](#). This dialog allows the user to specify the fraction of power that goes to each component. The axial power shape described in determines how the power will be destributed in the powered components.

Powered Components – Power 174 (Power Comp for old ht str 140)			
Unpowered Component	Radial Nodes [NODES]	Axial Levels [NZHTSTR]	
Heat Structure 126 (\$126\$ bkn-loop st-gen tubes)	3	2	▲
Heat Structure 127 (\$127\$ bkn-loop st-gen tubes)	3	2	■
Heat Structure 128 (\$128\$ int-loop st-gen tubes)	3	2	
Heat Structure 129 (\$129\$ int-loop st-gen tubes)	3	2	
Heat Structure 130 (\$130\$ vessel slabs (R1,L1))	2	1	
Heat Structure 131 (\$131\$ vessel slabs (R2,L1))	2	1	▼
Unpowered Components		Powered Components	
Powered Component	Axial Levels [NZHTSTR]	Power Fraction [CPOWER]	
Heat Structure 140 (\$140\$ reactor-core fuel rods)	3	1.0	
Heat Structure 171 (\$140\$ reactor-core fuel rods)	3	1.0	
Heat Structure 172 (\$140\$ reactor-core fuel rods)	3	1.0	
Heat Structure 173 (\$140\$ reactor-core fuel rods)	3	1.0	
<div>Close</div> <div>Normalize</div>			

Figure 9.1. Powered Components Dialog

9.1.2. Power Shape

The Power Shape is a table of ratios used to distribute the total power to regions within a heat structure. Figure 9.2, “Power Shape Data” below displays the *Power Shape* properties of a fuel rod heat structure. The radial power shape can either be entered as a constant shape if the *Power Shape* option is set to 0 (*1-D axial table*) or as part of the Power Shape Table if the *Power Shape* option is set to 1 (*2-D axial-r or axial-x table*).

Power Shape

Power Shape	[0] 1-D axial table (default)	▼	?
Power Shape Dependence	[0] Defined by IZPWZ (default)	▼	?
Axial Integration Option	[0] Histogram, Step Betwn Locs	▼	?
Use User Defined Shapes	[0] None	▼	?
Axial Initial Shape Value	0.0 (-)		?
Axial Off Shape Value	0.0 (-)		?
Axial Max Change Rate	0.0 (1/s)		?
Axial Shape Location Source	Locs. Defined Heat Structure	▼	?
Axial Shape Locations	Rows: 3 [1.2141],[2.4283],[3.6...	E▲	?
Radial Power Shape	Rows: 8 [0.0,1.2109],[2.0E-3,1...	E▲	?
Power Shape Table	1 shape of with 5x1 values	E▲	?
Axial Power Rate Factor	Rows: 0 []	E▲	?

Figure 9.2. Power Shape Data

The *User Defined Shapes* option allows constant radial or axial shape arrays to be specified for the feedback portion of the power calculation. The *Axial Shape Location Source* and *Radial Shape Location Source* options define whether the power shape should reflect the connected components, or if they should be based on a user defined table.

The Power Shape Table dialog shown below in [Figure 9.3, “Power Shape Table Dialog”](#), allows the user to modify the power shape table for specific entries of an input variable. This table contains a row for each axial level in the heat structure. If the *2-D axial-r or axial-x table* option is specified the table will include a column for each radial node. The left-most column displays the inside of the heat structure, and the right-most column corresponds to the outside of the heat structure.

	Radial	1	2	3	4
Axial	Location(m)	0.0	2.0E-3	3.0E-3	4.0E-3
1	0.0	0.64285	0.64285	0.64285	0.64285
2	1.2141	1.2321	1.2321	1.2321	1.2321
3	2.4283	1.1786	1.1786	1.1786	1.1786
4	3.6424	0.53571	0.53571	0.53571	0.53571

Figure 9.3. Power Shape Table Dialog

If the *2-D axial-r or axial-x table* option is specified the table will include a column for each radial node. The left-most column displays the inside of the heat structure, and the right-most column the outside of the heat structure.

Any number of shapes may be input. Each shape corresponds to a value for the selected *Abscissa* variable. Shapes can be added with the **Add Shape** button or removed with the **Remove Shape** button. The **Displayed Power Shape** spinner can be used to select the currently displayed shape.

9.2. Fluid Power

The Fluid Power components are used to inject heat directly into the fluid portion of hydraulic components. The power is either based on a table, or a constant value, and is distributed across volumes. [Figure 9.4, “Fluid Power Properties”](#) below displays the general properties of a fluid power component.

Fluid Power 124 (\$124\$ Target Pwr.)

General

☐ Optional ☐ Disabled

Component Name	\$124\$ Target Pwr.	?
Component Number	124	?
Description	< none >	E ?
Comments	< none >	E ?
In-beam Option	[6] Table Lookup	?
Decay Option	[5] Constant Power	?
In-Beam Table	Rows: 7 [0.0,0.0],[1000.0,5.0E6],[3600....	E ?
Initial In-Beam	0.0 (W)	?
Initial Decay	0.0 (W)	?
In-beam Cell Fractions	Powered Components [4]	E ?
Decay Cells	Powered Components [1]	E ?

Figure 9.4. Fluid Power Properties

9.2.1. Powered Components

The dialog for selecting volumes in the model for a fluid power is displayed below in [Figure 9.5, “Fluid Power Selected Volumes”](#). The top half of the dialog is a table that allows selecting hydraulic components inside the model that have fluid volumes. Selecting a volume in the top portion of the dialog displays the nodes inside the component that are currently powered. A fraction must be defined for each node to determine the fraction of current power applied to the fluid.

Define Powered Component Cells

Hydraulic Components

Hydraulic Component

Pipe 210 (\$210\$ core section)

Pipe 212 (\$212\$ core section)

Pipe 213 (\$213\$ core section)

Pipe 214 (\$214\$ core section)

Add

Remove

Powered Cells

Cell Number	Fraction
Cell 2 of 6	0.0625
Cell 3 of 6	0.0625
Cell 4 of 6	0.0625
Cell 5 of 6	0.0625

Add

Remove


Ok

Cancel

Figure 9.5. Fluid Power Selected Volumes

Chapter 10. Countercurrent Flow Limitation

Countercurrent flow limitation (CCFL) models are independent shared components in TRACE. They may be selected for any number of edges for 1D components or for cells in the Vessel component. [Figure 10.1, “CCFL Model Properties”](#) below shows the properties of a CCFL model.

 **CCFL Model 3**

General ☐ Optional ☐ Disabled

Component Name	unnamed	?
Component Number	3	?
Description	<none>	E? ?
Comments	<none>	E? ?
Bankoff Interpolation	-1.0 (-)	?
Slope	0.0 (-)	?
Correlation Constant	0.0 (-)	?
Number of Holes	9	?
Plate Thickness	0.0762 (m)	?
Flow Area Ratio	0.4869 (-)	?
Hole Diameter	0.1693 (m)	?
Bond Number Limit	0.0 (-)	?

Figure 10.1. CCFL Model Properties

Chapter 11. Contain Components

TRACE may be coupled with the CONTAIN analysis code for doing large-scale accident analysis. The CONTAIN component embedded inside a TRACE model serves as a connection point for containment boundary data from either the embedded CONTAIN libraries in TRACE, or a concurrently executing CONTAIN model. The individual portions of a CONTAIN component are represented as sub-nodes inside the navigator. [Figure 11.1, “CONTAN Sub-Nodes”](#) displays the sub-nodes inside a CONTAIN component in a sample model.

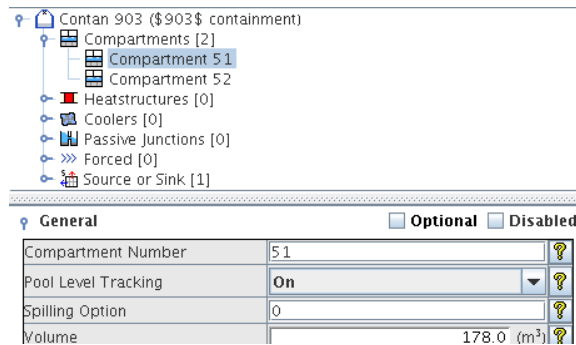


Figure 11.1. CONTAN Sub-Nodes

Chapter 12. Extern Components

TRACE allows for distributed runs where a single model is divided into portions and run simultaneously on separate processes or machines. The EXTERN component represents a component that is currently defined in a different portion of the model. EXTERN components can be referred to for hydraulic connections or for signal variables, or for boundary conditions. The properties of a simple exterior component are displayed in [Figure 12.1, “Exterior Component Properties”](#).

Exterior 1

☒ General ☐ Optional ☐ Disabled

Component Name	unnamed	?
Component Number	1	?
Description	<none>	E ?
Comments	<none>	E ?
Component Type	[1] Fluid Component	?
Dimensions	[1] X or Radial Axis Only.	?
X Cells	10	?

Figure 12.1. Exterior Component Properties

Chapter 13. ASCII I/O

TRACE models in the ModelEditor can be exported to ASCII files. The ASCII export options may be found on the right-click pop-up menu from the model node in the navigator as shown below in [Figure 13.1, “TRACE Export Options”](#) and in the **Export** sub-menu of the **File** menu.

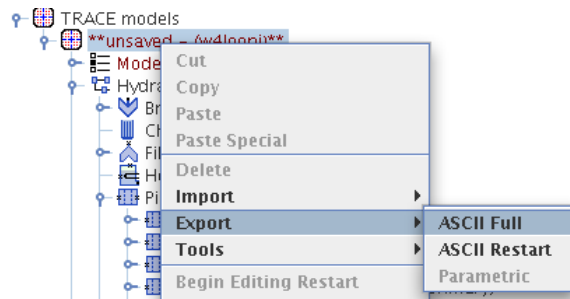


Figure 13.1. TRACE Export Options

A TRACE "input deck" is an ASCII file containing all the data required by the TRACE analysis code to execute a calculation. Entire TRACE models can be imported from TRACE input decks by using the **File->Import->TRACE** menu item. TRACE models can be exported as TRACE compliant input decks using the menu item found in the **Export** menu shown in [Figure 13.1, “TRACE Export Options”](#) shown above.

Chapter 14. Retrieve Initial Conditions

When restarting a TRACE calculation, any resupplied hydraulic components should have initial conditions that correspond those present in the calculation at the time of restart. The TRACE plug-in has the capability to import these values from a completed job located on a Calculation Server, or from a local TPR file.

Note It is advised that this is done before beginning the restart edit. If restart editing has been enabled, every hydraulic component, control block and heatstructure in the model will be marked as modified after the initial conditions have been imported.

To import the initial conditions into a TRACE model, right-click on the model node in the navigator and select the **Retrieve Initial Conditions** option as shown below in [Figure 14.1](#), “Retrieve Initial Conditions Pop-up Menu Item”.

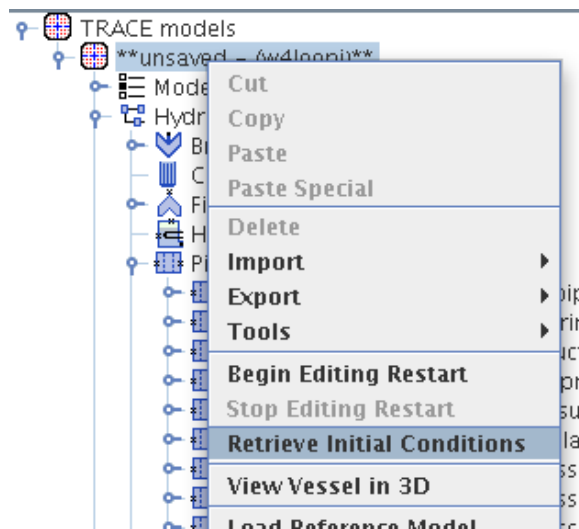


Figure 14.1. Retrieve Initial Conditions Pop-up Menu Item

The initial conditions dialog displayed below in [Figure Retrieve Initial Conditions](#), “Retrieve Initial Conditions Dialog” requires the user to either select a job from a calculation server, or a TPR file. Once the job is selected the specific restart location must be selected from the available list. When the dialog is closed the model will be updated to match the selected conditions.

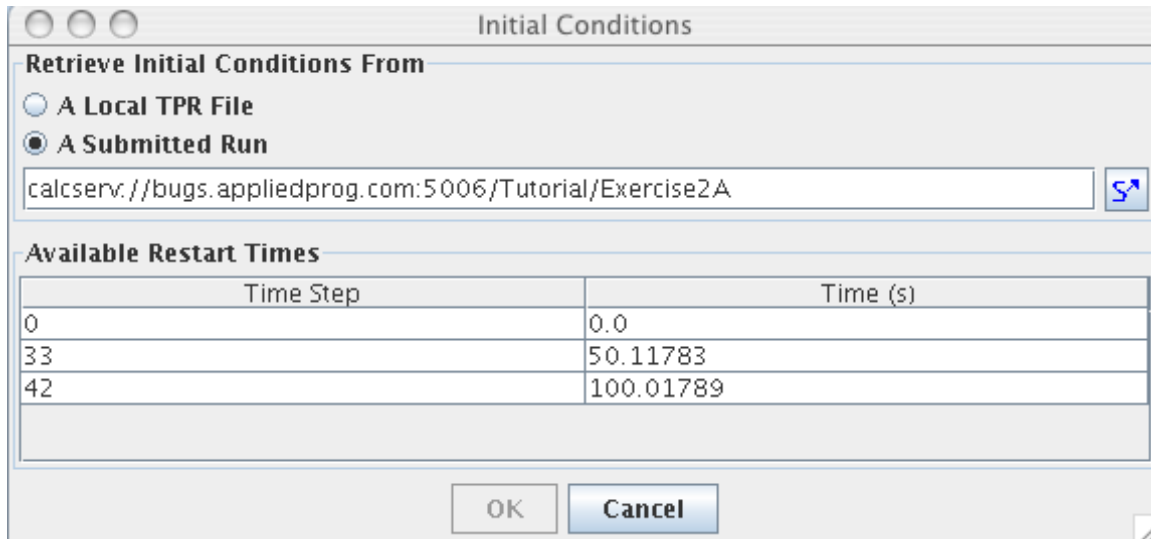


Figure Retrieve Initial Conditions. Retrieve Initial Conditions Dialog

14.1. Managing Initial Conditions

In addition to being able to retrieve initial conditions, the TRACE plug-in allows a user to store and load existing initial conditions. Each of the stored initial condition sets will be saved and loaded with the model. The editor for managing model initial conditions can be accessed from either the right-click pop-up menu item of the TRACE model node or in the Model Options property view next to the Initial Conditions property. The editor is displayed below in [Figure Manage Initial Conditions, “Manage Initial Conditions Editor”](#).

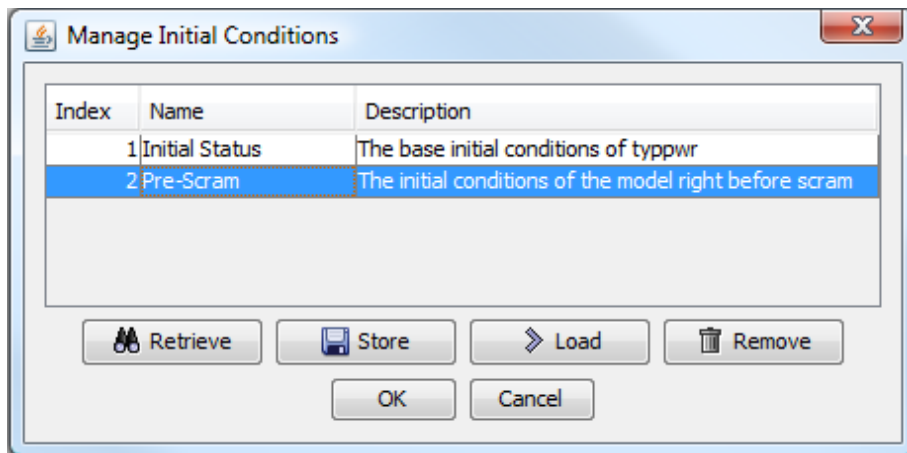


Figure Manage Initial Conditions. Manage Initial Conditions Editor

Notice that the editor provides a few options for managing initial conditions. Selecting the **Retrieve** button will cause the dialog to display the retrieve initial conditions dialog shown in [Figure Retrieve Initial Conditions, “Retrieve Initial Conditions Dialog”](#). The dialog provides a way to set the initial conditions by importing them from an existing run. Selecting the **Store** button will store the current model initial conditions. This can be used in conjunction with the **Retrieve** button to save sets of retrieved initial conditions. Once a set of conditions has been stored, a name and label can be applied to identify those conditions. Selecting the **Load** button

will load the conditions of the selected set. A stored set of conditions will be deleted when the **Remove** button is selected.

Note Once a set of conditions is retrieved from an existing run and stored, the conditions can be loaded without having the existing run available.

Chapter 15. Model Validation Tests

The TRACE plug-in performs optional tests to validate the data inside the model. These tests occur before a model is exported to ASCII, submitted to a calculation server or when the **Check Model**, ✓, button is pressed. The validation tests are displayed below in [Figure 15.1, “Model Validation Tests”](#). All validation tests include a verbosity option for controlling the amount of detail provided by the test. Only tests that are *Enabled* will be included during model validation.

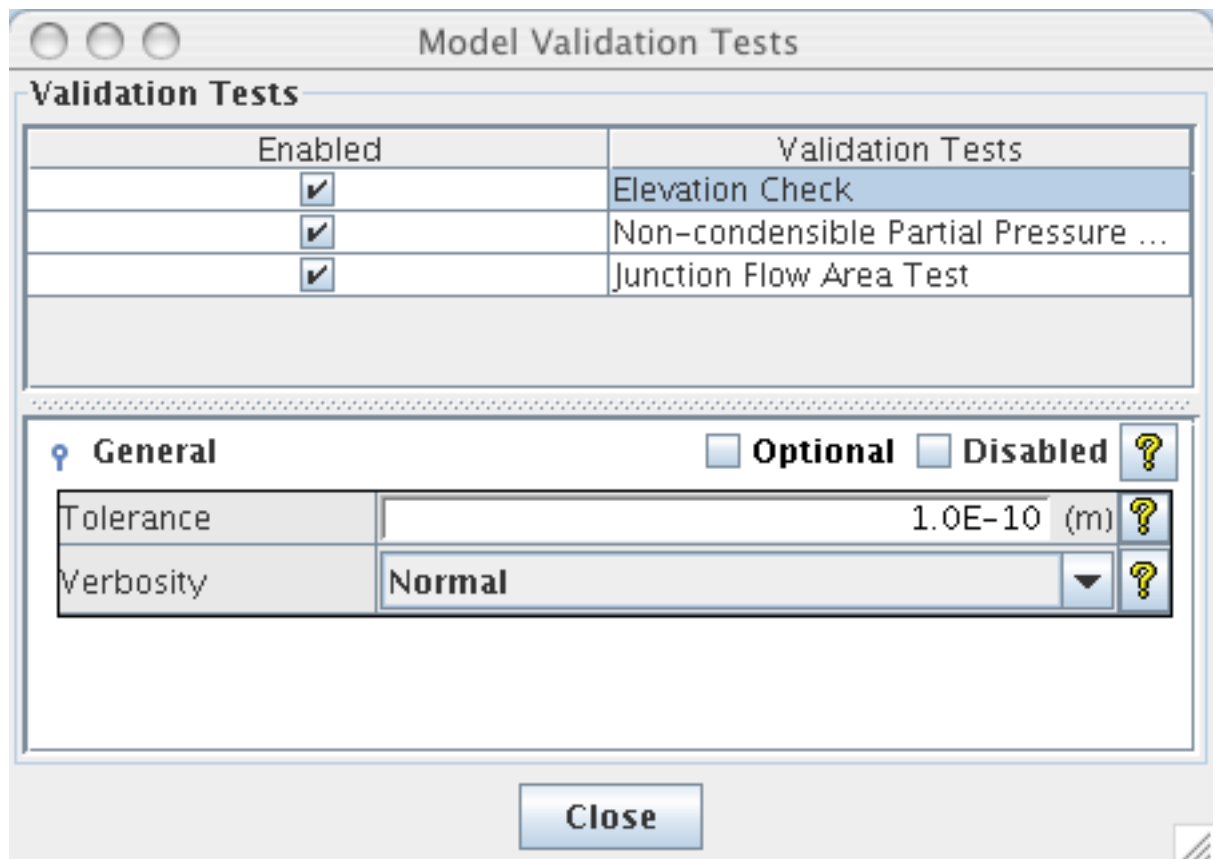


Figure 15.1. Model Validation Tests

15.1. Loop Check

This test verifies the specified position of components inside the model. Also known as the Loop Checker, this verifies that all hydraulic component elevation data is consistent between connected components. If **grav** terms are defined, then the loops will be checked to ensure that they have consistent lengths and **grav** terms to prevent gravity pumping. The *Tolerance* setting allows the user to define the maximum allowed distance between cell edges.

15.2. Non-Condensible Partial Pressure Test

This validation test provides an error message when any hydraulic volume inside the model contains non-condensables that when the *NC Pressure Calc (NOAIR)* namelist variable is On (0). When the *NOAIR* namelist variable is set to Off (1), the model is required to contain no

non-condensables. This test runs through all of the hydraulic volumes in the model and reports which components need to be modified.

15.3. Junction Flow Area Test

The junction flow area test ensures that adjacent hydraulic volumes have consistent geometry. Each edge inside the model is compared with a calculated flow area for the volumes on either side. An error is reported if the edge's flow area is more than 10% larger than the smaller of the two calculated flow areas. This includes edges between two components. Vessels, Plenums, Fills and Breaks are excluded from this test.

15.4. Abrupt Area Change Test

The abrupt area change test reports edges that should have a friction, or the abrupt area change flag set. The calculated area of adjacent volumes are compared. If the volumes differ by more than a user-modifiable ratio, the volumes are determined to involve an area change. This test will then report an error if the intervening edge between the two volumes does not have either friction defined, or the abrupt area change model enabled. This test optionally compares the edge's flow area with the adjacent volumes area as well.

15.5. Edge Hydro-Diameter Test

This validation test compares the hydraulic diameter specified for edges inside 1D hydraulic components with the calculated hydraulic diameter of the adjacent volumes. An error is reported if the edge is not within the tolerance of one of the calculated diameters. Lumped models often contain hydraulic diameters for components that are much smaller than a calculated hydraulic diameter, so this test ignores edge hydraulic diameters that are smaller than the calculated adjacent volume diameters.

15.6. PUMPFRICTQ Consistency Test

The PUMPFRICTQ validation test verifies that power provided by enabling the PUMPFRICTQ namelist option does not conflict with power added using a FLPOW component. A detailed warning message is provided through the message window during the event where the PUMPFRICTQ namelist option is enabled and a fluid power component is being used.

15.7. Vessel Connection Test

The Vessel Connection test verifies the consistency of Hydraulic connections to the vessel. The component flow direction is compared target vessel cell face to ensure a consistent flow between components. Additionally this test verifies that the hydraulic connections to the vessel do not occur inside vessel nodes that have a volume fraction of 0.0.

15.8. Adjacent Cell Volume Test

The Adjacent cell volume test verifies adjacent cell volumes do not differ by a pre-defined ratio. As a default, volumes are considered to be in error if one is more than 10 times the volume of the other. This ratio can be specified on a model-by-model basis.

15.9. Channel Surface Test

The Channel surface test ensures that the canister surface connections between a channel and a vessel are consistent. The the axial length of the channel nodes are compared with the height of the connected vessel levels. If the channel nodes, taken together, are longer than the vessel level an error is reported. This test is not run in models that do not contain channels.

Chapter 16. Renodalization

The size and shape of a hydraulic component can be modified through built-in editors. However, a special tool is required to change the number of hydraulic cells while preserving the overall geometry and external connections. Renodalization tools are provided separately for 1D and 3D hydraulic components. These tools are described in the following sections.

Note All 1D and 3D renodalization can be safely undone and redone.

16.1. 1D Hydraulic

All multi-cell 1D hydraulic components may be renodalized. These include Pipe, Pump, Prizers, Tees, and Valves. To begin the process, select the **Renodalize** option from the right-click pop-up menu of the component in either the navigator or the 2D view as shown in [Figure 16.1, “1D Hydraulic Pop-up Menu”](#).

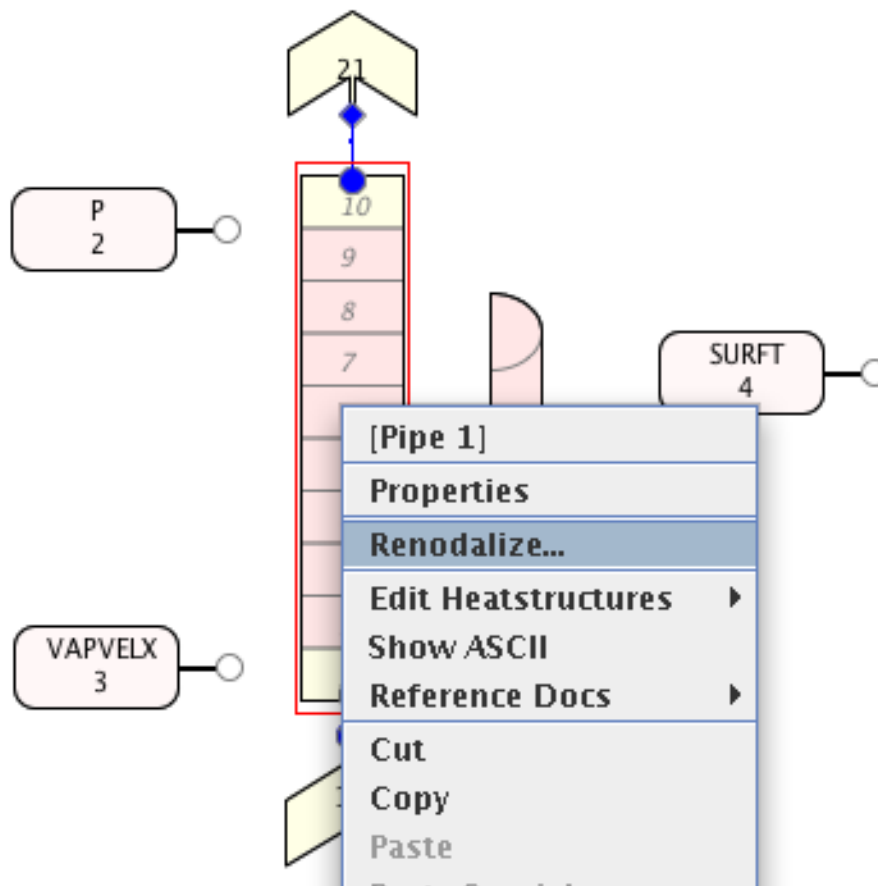


Figure 16.1. 1D Hydraulic Pop-up Menu

The **Renodalize** option opens the Renodalization dialog. This dialog contains a 2D view of the component at the top and a table of either axial node lengths (*Nodes*) or total elevation (*Elevation*)

at the bottom. [Figure 16.2, “1D Hydraulic Renodalization Dialog”](#) displays the 1D renodalization dialog for the pipe shown in [Figure 16.1, “1D Hydraulic Pop-up Menu”](#). This dialog allows the component to be renodalized by splitting and merging selected cells. Cells can be selected either in the top or bottom of the dialog and split or merged by using **Split**, **Split Uniform** and **Merge** buttons. The top and bottom of the dialog will update to display the results of each operation. Splits and merges can be undone and redone using the forward and back buttons provided. When the **Next** button is pressed, the Renodalization Options Dialog will open. The [Section 16.1.7, “Renodalization Options Panel”](#) allows the user to customize how affected components reconnect themselves to the renodalized component.

The **Revert** button can be used to set the nodalization of the current component to that of a component in the reference model. Revert is discussed in more detail in [Section 16.4, “Reference Model”](#).

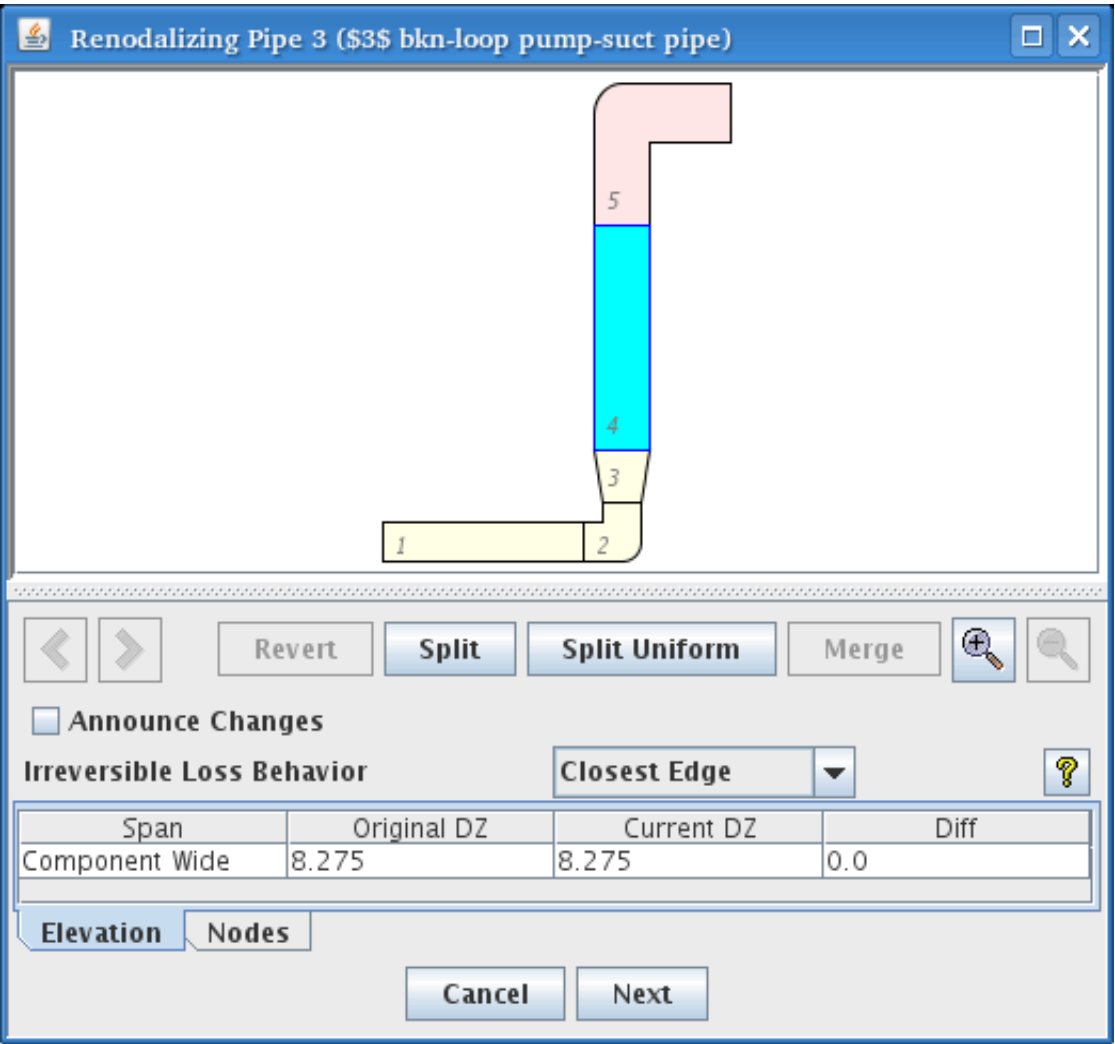


Figure 16.2. 1D Hydraulic Renodalization Dialog

16.1.1. Split

The **Split** button is enabled when one or more cells are selected. This button opens the Split Cells Dialog shown in [Figure 16.3, “Split Cell Dialog”](#). This dialog allows the user to specify how

new cells will be created from each of the selected cells. The Split Cells Dialog contains a table of the new cells that will be created and the fraction of the original cell they will represent. If the fractions entered do not sum to 1.0, the user will be prompted to normalize the ratios using the **Normalize** button before continuing.

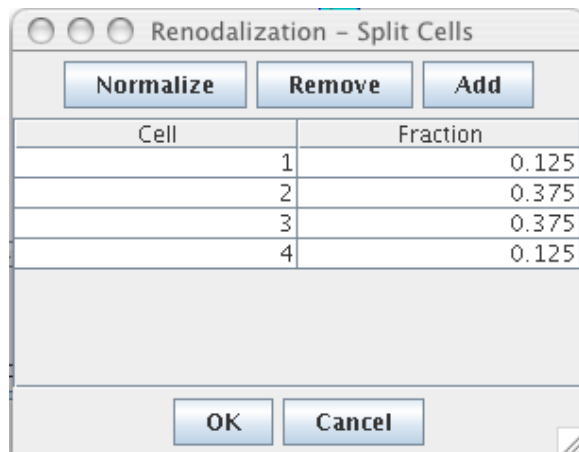


Figure 16.3. Split Cell Dialog

Splitting a cell results in multiple smaller cells that occupy the same total volume as the original. The cell volume is examined and compared with the flow area's on either side of the cell. If there is an abrupt area change, then all internal edges are based on the area corresponding to the cell's volume. If the volume does not correspond to the flow area on either side of the cell, the cell is assumed to be a conical section, and the internal edge flow areas are calculated appropriately. These same conditions are used to determine the volume of each of the smaller cells.

16.1.2. Split Uniform

The **Split Uniform** button allows the user to split each of the selected cells into a user specified number of evenly sized cells. Properties for each edge are determined in the same manner as described in [Section 16.1.1, “Split”](#).

Note To avoid creating elevation changes, split cells into an odd number of nodes when the cell is an elbow, or has a crossflow connection.

16.1.3. Merge

The **Merge** button joins the selected cells together into a single cell. This button is enabled when 2 or more cells are selected.

Note The cells on either side of a valve interface may not be merged together.

16.1.4. Elevation Change

The *Elevation* tab contains a table that displays the *Original DZ*, the *Current DZ* and the *Difference* between the two. This table is used to determine if the elevation change of a

component has been affected by the current renodalization. Renodalization can drastically change the absolute elevation change across a component. The most common way to inadvertently change the elevation is to split an curved cell into an even number of nodes, or merge a curved cell with a neighboring cell.

16.1.5. Announce Changes

Activating the **Announce Changes** option will provide additional information about the renodalization process when the **OK** button is pressed. Any component that is affected by the resulting renodalization will display a message to the Message Window. These messages allow the user to track which components have been changed.

16.1.6. Irreversible Loss Behavior

The irreversible loss option determines how friction losses are preserved when cells are merged together. The available options include merging with the inlet edge, merging with the outlet edge, and merging with the closest edge. The losses are all converted to K-Factors, to preserve the pressure drop across the whole pipe during the renodalization.

16.1.7. Renodalization Options Panel

After the hydraulic component renodalization is complete, the Renodalization Options Panel is displayed. This panel allows the user to customize how components associated with the 1D hydraulic component are modified in response to the renodalization.

A tree containing each component affected by the renodalized component is displayed on the left hand side of the dialog. The root level of the tree contains all the components directly affected by the original component renodalization. Each affected component may require additional component changes. For example, a heat structure may change its axial nodalization after a connected pipe is renodalized. Any signal variables on this heat structure will need to be modified to reflect the new axial nodalization of the heat structure. These signal variables would appear as leaf nodes under the heat structure.

The top of the dialog is a rendering panel that displays a scaled representation of a component without elevation changes or volume, just axial length. The displayed component reflects the cause of the selected component's adjustment. In the [Figure 16.4, "Renodalization Options Panel"](#) below, the selected signal variable component was modified by Pipe 3's renodalization. The rendering panel displays Pipe 3's original and modified nodalization, and then highlights the reference cell selected by the signal variable.

To the right of the component tree is a property view that displays the options that may be modified for the selected component. In the example below, the user can adjust the signal source reference cell by using the spinner provided. If changing an option results in a difference in component nodalization, the child nodes will be reset.

Below the property view is a report panel that displays details about results of the renodalization on the currently selected node.

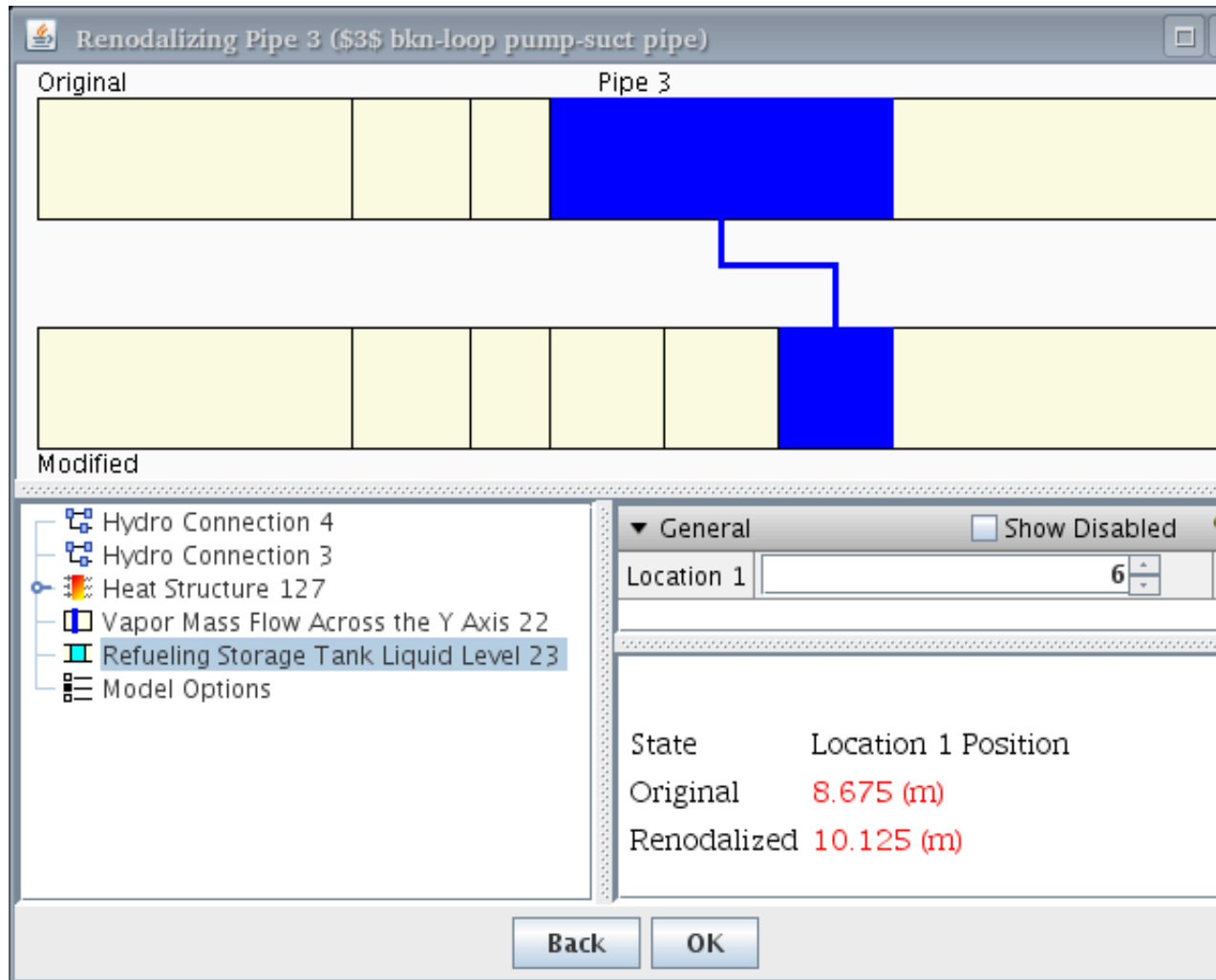


Figure 16.4. Renodalization Options Panel

When the **OK** button is pressed, the changes are finalized and applied to the model. The Renodalization Report Dialog is displayed after the renodalization is complete. This dialog, shown in [Figure 16.5, “Renodalization Report”](#), contains detailed information regarding the components modified by the renodalization. Changes which may affect the result of the calculation, such as elevation changes, are reported as warnings. The report can be exported to a local HTML file by using the **Export** button.

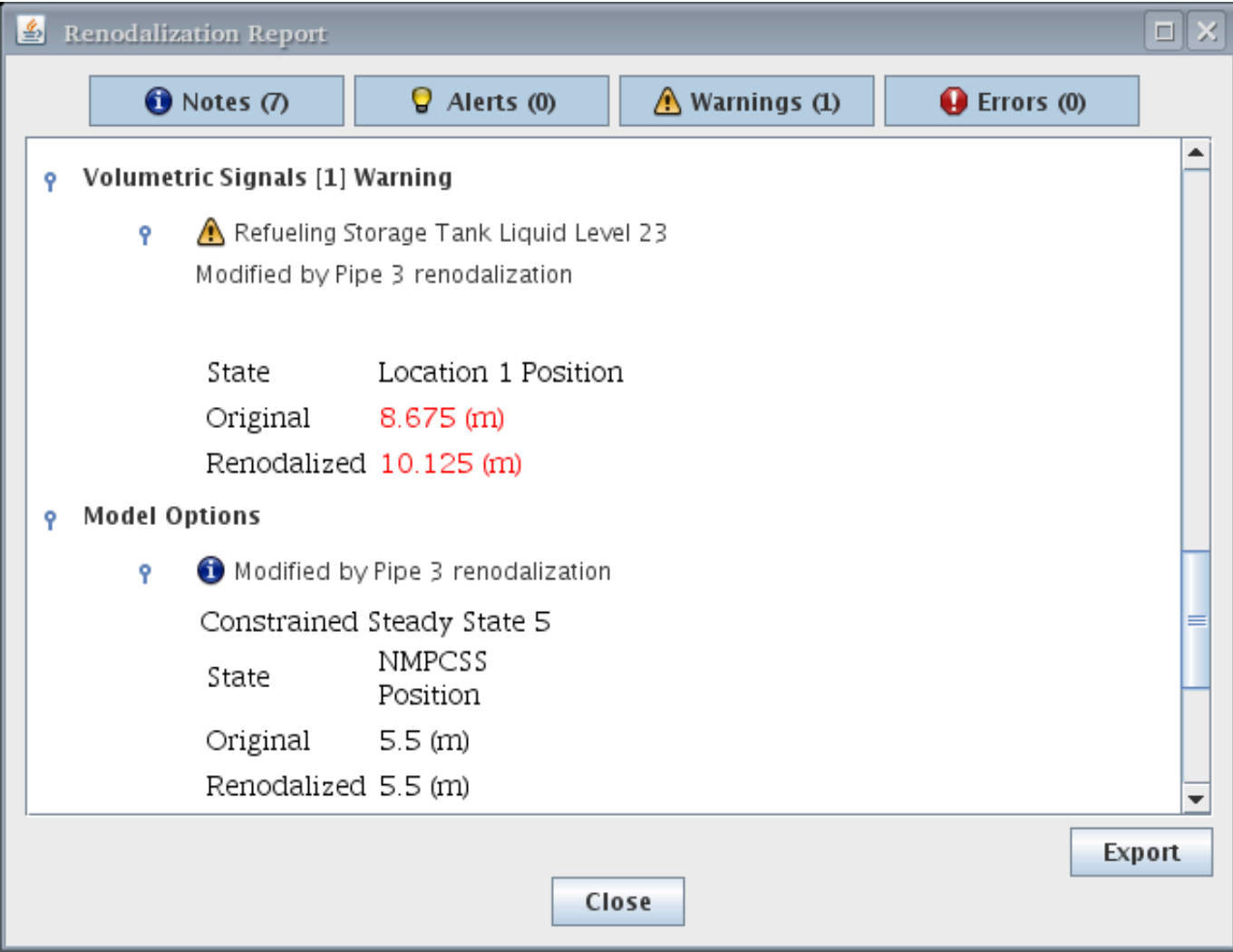


Figure 16.5. Renodalization Report

16.2. 3D Hydraulic

Vessel components inside a TRACE model may be renodalized along each axis (axial, radial, azimuthal). [Figure 16.6, “Vessel Pop-up Menu”](#) below displays the right-click pop-up menu that appears for a Vessel component.

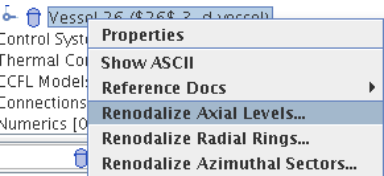


Figure 16.6. Vessel Pop-up Menu

16.2.1. Vessel Axial Renodalization

When renodalizing the axial levels of a vessel, the total height of the vessel is preserved. [Figure 16.7, “Vessel Axial Renodalization Dialog”](#) below displays the axial renodalization dialog for a TRACE vessel. Axial renodalization is very similar to the 1D hydraulic renodalization.

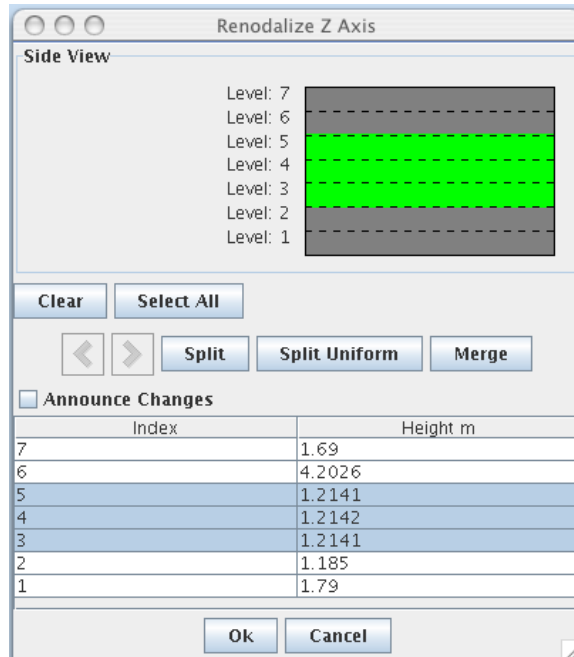


Figure 16.7. Vessel Axial Renodalization Dialog

Heat structures that are connected to the vessel are modified in the same manner as those connected to a pipe that is renodalized. Axial nodes inside the heat structure will be split to match split vessel levels. This will result in a heat structure renodalization as described in [Section 16.3, “Heat Structure”](#) below.

Hydraulic connections to axial levels within the vessel will be moved to the closest matching edge location. Connections to radial or azimuthal faces will match the closest cell center to the original cell center. Axial connections will connect to the closest edge to the original edge.

Note Adjusting Axial Levels may modify the elevation of connections to a vessel. Run the loop-check calculation after renodalizing a vessel to check for changes.

Edge and Volume signal variables will be updated to refer to the closest location in the renodalized vessel to their original location. The volume variables will find the closest cell center location, and edge variables will find the closest edge position.

Note Signal variables that refer to renodalized components should be checked following renodalization to ensure that their usage as control system inputs is appropriate. For example, a mass flow signal variable across an edge that is split into thirds will report only a third of the flow following renodalization. Modification of the control system would be required to obtain the total flow across the three edges.

Channel components require special consideration during vessel renodalization. The channel nodalization is important and will not be changed when the vessel nodalization increases. Instead when the levels of the vessel that a channel connects to are modified, the channel surface is connected to the resulting channel based on the location in the vessel of the channel inlet. Channel cells will be connected vertically from the channel inlet based on channel cell length. The outlet hydraulic connection will limit the vessel level where channel nodes can be connected, so in the case where the channel is longer than the associated vessel nodes, the excess channel length will be added to the highest vessel node.

16.2.2. Vessel Radial Renodalization

The number of rings, or nodes along the X axis of a vessel can be modified by selecting the **Renodalize Radial Rings** (Cylindrical) or **Renodalize X Axis** (Cartesian) option from the right-click pop-up menu on a vessel. The renodalization dialog is shown in [Figure 16.8, “Vessel Radial Renodalization Dialog”](#). This dialog allows the user to split or merge rings in a similar fashion to the to levels in the axial renodalization dialog. The primary difference between axial and radial renodalizations is how heat structures are modified.

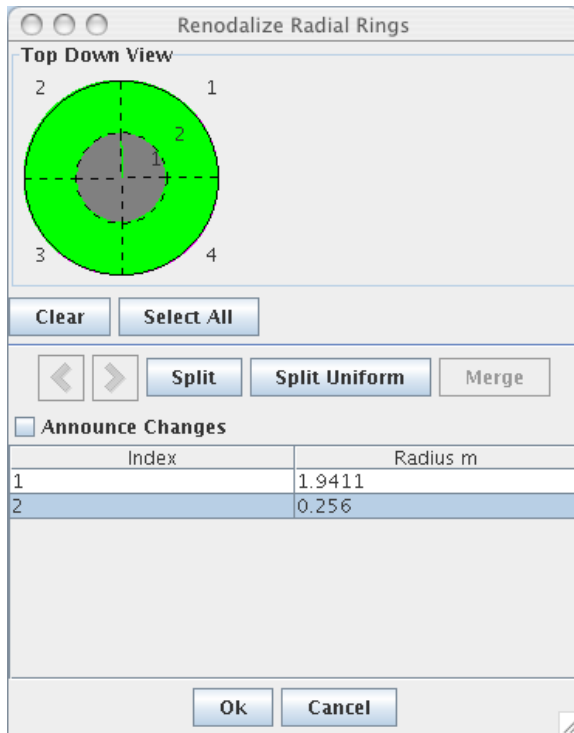


Figure 16.8. Vessel Radial Renodalization Dialog

Heat structures that connect to vessel nodes will be moved to the radial ring location that contains the volume to which they originally connected. Vessel wall heat structures that connect different rings inside the vessel will be updated to connect the same volumes. This may result in a heat structure that has the same cells on both the outside and inside.

Hydraulic connections are moved to the location in the renodalized vessel that most closely matches the original connection location. Axial and azimuthal connections are moved to the ring whose center radius most closely matches the radius of the original ring. Radial connections are moved to the ring whose edge most closely matches the original location.

Edge and Volume signal variables will be moved to the closest location in the renodalized vessel to their original location. Volume variables will be moved to the closest cell center location, and edge variables to the closest edge position.

16.2.3. Vessel Azimuthal Renodalization

The number of azimuthal sectors nodes along the Y axis of a vessel is modified by selecting the **Renodalize Azimuthal Sectors** or **Renodalize Y Axis** option from the right-click pop-

up menu on a vessel. The renodalization dialog is shown in [Figure 16.9, “Vessel Azimuthal Renodalization”](#). This dialog allows the user to split or merge sectors in a similar fashion to the levels in the axial renodalization dialog. The primary difference between axial and azimuthal renodalizations is how heat structures are modified.

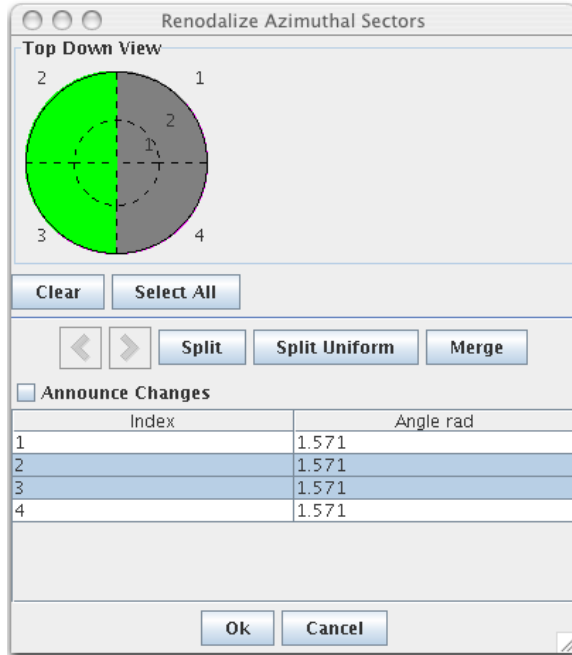


Figure 16.9. Vessel Azimuthal Renodalization

Heat structures that connect to vessel nodes will be moved to the azimuthal location that contains the node to which they were originally connected. Vessel wall heat structures that connect different sectors inside the vessel will be modified to connect the same volumes. This may result in a heat structure that has the same cells on the outside as the inside.

Hydraulic connections are moved to the location in the renodalized vessel that most closely matches the original connection location. Axial and radial connections are moved to the sector whose center angle most closely matches the angle of the original sector. Azimuthal connections are moved to the sector whose edge most closely matches the original location.

Edge and Volume signal variables will be moved to the closest location in the renodalized vessel to their original location. Volume variables will be moved to the closest cell center location and edge variables to the closest edge location.

Note Some volume variables may produce different results when the volume of a connected cell is modified. Pay attention to the signal variables that report a change

16.3. Heat Structure

Heat structures may be renodalized radially and axially. Radial renodalization occurs when the meshpoint array defining the radial geometry of a heat structure is modified. Axial renodalization occurs either when changing the hydraulic component connected to the heat structure or when the user modifies the axial nodalization directly.

16.3.1. Radial Renodalization

The radial geometry of a heat structure may be modified through the radial geometry dialog shown in [Figure 8.6, “Heat Structure Radial Geometry Dialog”](#). When the dialog is closed, the heat structures will interpolate all internal tables (such as fuel rod temperature) based on the radial dimension. Any heat structure signal variable reference that includes a radial location will automatically be updated to refer to the radial location closest to the original radial location selected.

16.3.2. Axial Renodalization

The most frequent cause of an axial renodalization for a heat structure is in response to renodalization of a connected hydraulic component. The heat structure will attempt to produce the most compact nodalization that will provide equivalent heat transfer. Essentially, whenever a hydraulic cell connected to a heat structure is split, the heat cell will be split in the same locations. Heat cells will merge together as well, but only if the surface boundary conditions on both sides of the heat cells are identical at each level. That is, if the left surface BCs are the same, and the right surface BCs are the same, the two cells will automatically coalesce into a single cell.

The user may create a new heat structure nodalization by splitting or merging axial levels directly through the *Axial Node / Surface BC's* editor. [Figure 16.10, “Heatstructure Axial Editor”](#) shows each of the axial nodes along the heat structure.

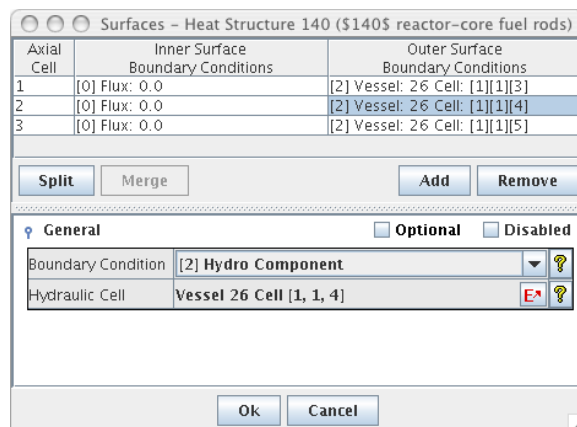


Figure 16.10. Heatstructure Axial Editor

16.4. Reference Model

The TRACE plug-in supports reverting the nodalization of hydraulic components to a previous state stored in a separate TRACE MED file. This file is referred to as a Reference Model. To use the reference model, first select the external MED file using the **Load Reference Model** option from the right-click pop-up menu of the model in the navigator. This menu item is displayed below in [Figure 16.11, “Reference Model Menu Item”](#).

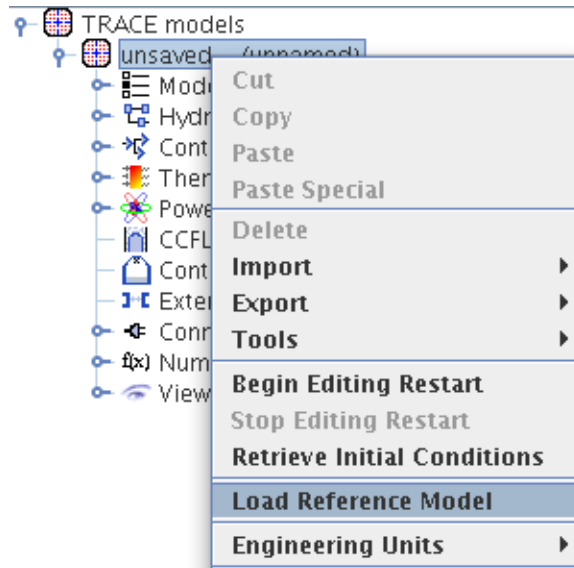


Figure 16.11. Reference Model Menu Item

Once the Reference Model has been loaded, hydraulic components can be reverted back to the nodalization and properties contained inside the reference model. Pressing **Revert** button in the renodalization dialog will open a dialog to select the reference component. The reference component's nodalization will be copied over to the current component.

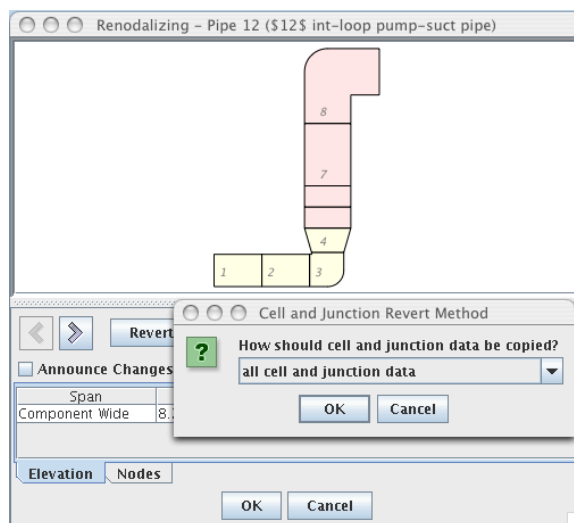


Figure 16.12. Revert Component

After selecting a component, the user has the option to copy over *all cell and junction data* or *only geometric data* as shown in Figure 16.12, “Revert Component”. Like a other renodalization operations, the changes will not be made until the user presses the **OK** button. The related components, such as heat structures and hydraulic connections, will update as though the user had modified the nodalization directly.

Chapter 17. Loop Closure

The loop closure tool inside the TRACE plug-in allows the user to examine the exact elevations of nodes inside the model. This test examines the hydraulic components inside the model and determines the separate hydraulic systems. These systems are then listed in a dialog shown in [Figure 17.1, “Hydraulic Systems Selection”](#), which allows the user to select a root component, and to specify an inlet elevation for that component. If the **IELV** namelist variable is set true, the inlet elevations are initialized by the cell-centered elevations specified for fluid components.

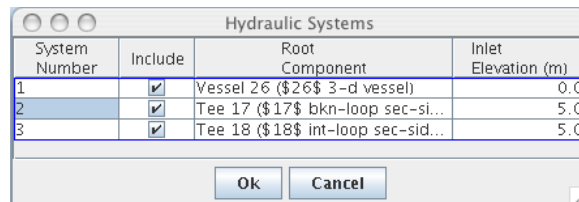


Figure 17.1. Hydraulic Systems Selection

Once the initial elevation is selected, pressing the **Next** button opens the report dialog. This dialog, shown in [Figure 17.2, “Loop Closure Dialog”](#), displays the elevations calculated for each connection for a selected system. A label at the top indicates if the current systems closure is within tolerance. If the loop has failed the tolerance check, the list of errors is displayed in the lower part of the dialog. Selecting the verbose report displays the cell-centered elevation for each node in a standard hydraulic component, and each level in a vessel component, as well as the connection elevations. The data may be exported to a comma-separated values (.CSV) formatted text file by pressing the **Export** button.

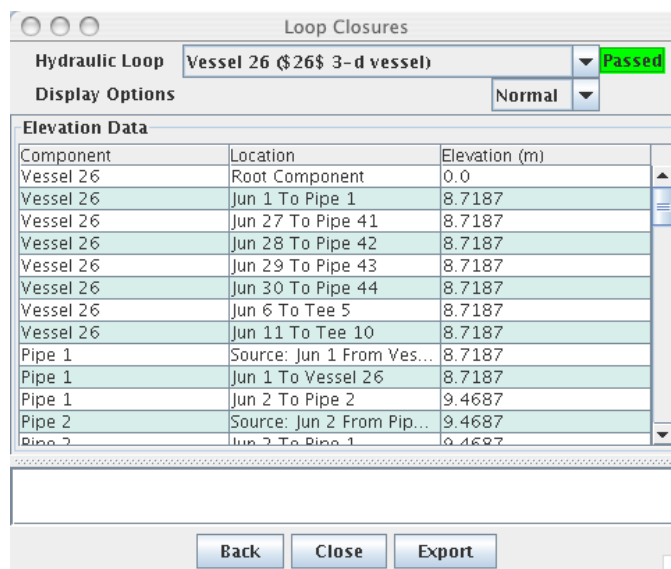


Figure 17.2. Loop Closure Dialog

Chapter 18. Model Notebooks

The TRACE plug-in allows generating model-wide reports as a single annotated document, called a model notebook. Information such as calculations, export data, model status, attribute descriptions, etc. are all included.

To export a model notebook, first open either the **File** menu or right-click on the model node in the Navigator. In the resulting menu, select **Model Notebook** from the **Export** sub-menu. An *Export Model Notebook* dialog will appear, as shown in Figure 18.1, “Export Model Notebook Dialog”. Document-related settings are all placed within the **General** and **Sub-Systems** tabs, explained below. The **Export** button at the bottom of the dialog will initiate notebook generation with the current configuration.

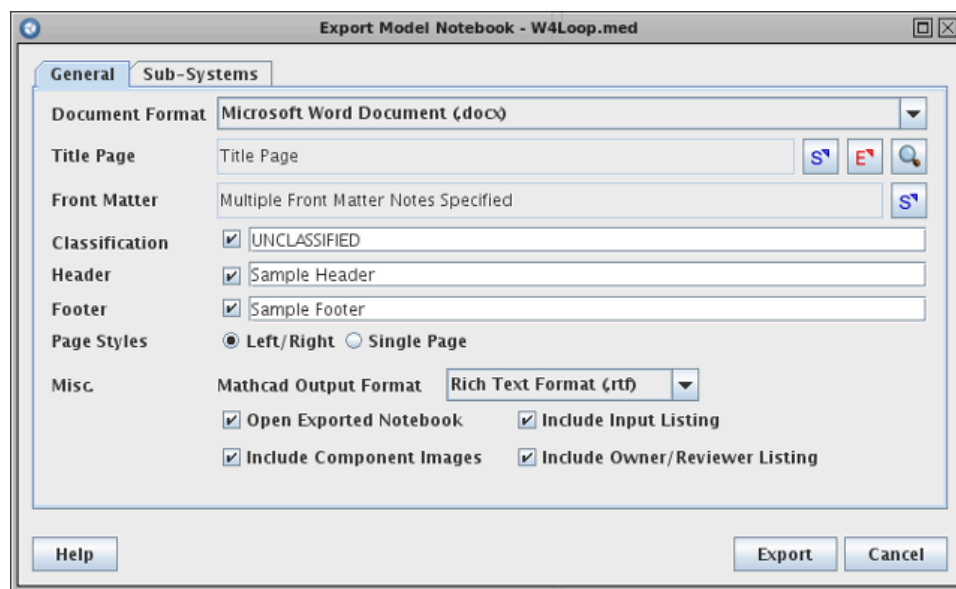


Figure 18.1. Export Model Notebook Dialog

General Settings

The majority of document configuration occurs in the **General** tab. The available options are described below.

Title Page allows selecting or creating a model note to use as the document title page. To the right of the field are three buttons related to the reference: **Select**, **Edit**, and **Preview**. **Select** brings up a selection dialog to specify which note is referenced. **Edit** opens a dialog for modifying the selected note. Finally, **Preview** will export the note as an ODF to a user-specified location and open it in the system-specific default document viewer.

The *Classification*, *Header*, and *Footer* fields are optional values placed into the document header and footer. *Header*, when specified, is placed at the top of every page in the document; likewise for *Footer* and the bottom of the document. If *Classification* is specified, the text is placed in large type in both the header and footer.

Page Styles determines whether the document is generated as pages that alternate between "left" and "right" pages, suitable for printing in book form, or as single pages more suited to an

electronic document. This setting mostly relates to page margins and the organization of elements in the header and footer, such as which side of the footer includes the page number.

Misc. contains the remaining options.

- **Mathcad Output Format.** When user numerics are included in the notebook and one or more are Mathcad functions, this setting determines the format of the function output created by Mathcad and linked in the document.
- **Include Input Listing.** When selected, the ASCII output for each component is listed in a code section after the annotated information.
- **Include Owner/Reviewer Listing.** Determines whether owner and reviewer names and timestamps are included in the annotated component information.
- **Open Exported Notebook.** When selected, the generated document will be opened in the system-specific document viewer.
- **Include Component Images.** Determines whether components with non-trivial View graphics include the component image in the document.

Sub-System Settings

Exported model notebooks may opt to organize model components by sub-systems. This functionality is defined in the **Sub-Systems** tab, as shown in [Figure 18.2, “Export Model Notebook Sub-Systems Tab”](#).

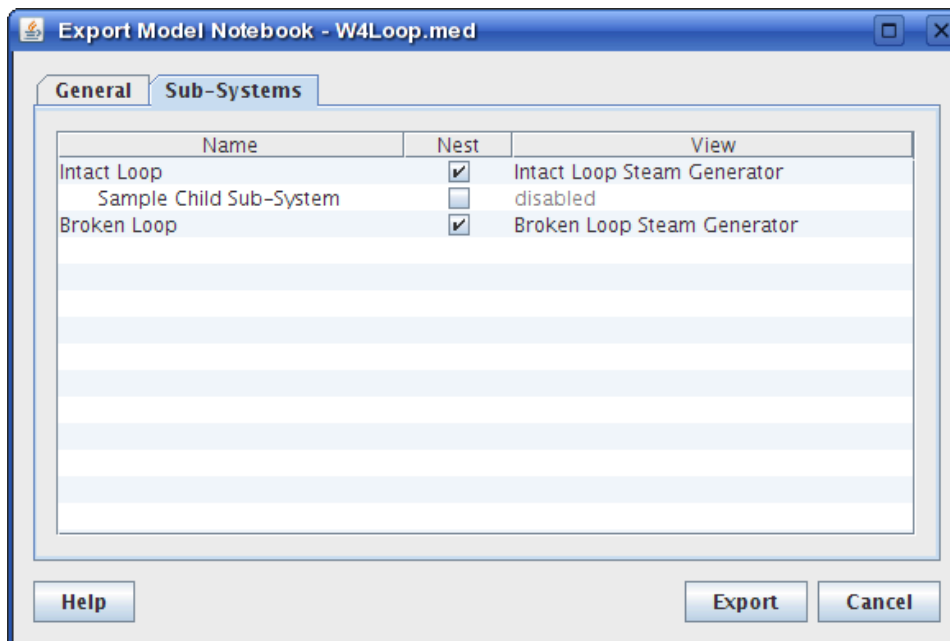


Figure 18.2. Export Model Notebook Sub-Systems Tab

The list of sub-systems is limited to top-level, "parent" sub-systems (those not found within another system) and any "child" sub-systems they contain directly (child status is indicated by an indented name). The *Nest* flag indicates whether the components in the sub-system are organized

into a distinct section in the document (nested) or folded into their parent section (non-nested). In the case of non-nested top-level systems, the system components are folded into an implicit model-wide section. That same model-wide section is also the home of all model components not organized into a sub-system. If either the model is devoid of sub-systems or no systems are nested, then system-level sectioning is not performed at all.

Note Third-level sub-systems and beyond are implicitly folded into their second-level parents.

The *View* column allows setting a view used to represent the sub-system. When a view reference is made, an image of the view will be placed at the top of the system section. View references are disabled for non-nested systems, as they have no distinct section to house the image.

Chapter 19. Batch Commands

The TRACE plug-in adds support for the following ModelEditor batch commands.

TRACE IMPORT [file type] <Mn> [filename]

This command imports data from an input file. If the file contains a complete model a new model will be created inside the ModelEditor. Otherwise, the data must be imported on top of an existing model.

- **[file type]** - One of the following import file types:

ASCII_FULL - An ASCII input file compliant with TRACE.

ASCII_RESTART - An ASCII restart input file to import on top of the current model

TPR - A platform independent binary restart file.

- **<Mn>** - An optional argument used to identify the model. Valid model labels are **M0-M9**.
- **[filename]** - The file name which should include the full path to the file in quotes.

TRACE EXPORT [file type] <Mn> [filename]

This command exports a file of the specified type to the given file name. Each of the available file types is described below.

- **[file type]** - One of the following export file types:

ASCII_FULL - Exports the current model in the TRACE ASCII input format.

ORDERED_ASCII - Exports the current model in the TRACE ASCII input format, in the same component order as the original ASCII file.

ASCII_RESTART - A TRACE ASCII restart input file including the components modified after the previous **BEGIN_RESTART** command.

METRICS_SPEC – The model specification file for the Test Suite Analyzer (TSA).

METRICS – The plug-in metrics data file for Test Suite Analyzer (TSA).

NOTEBOOK_ODT – A Model Notebook as an OpenDocument Format (ODF) file.

NOTEBOOK_DOCX – A Model Notebook as an Office Open XML file.

- **<Mn>** - An optional argument used to identify the model. Valid model labels are **M0-M9**.
- **[filename]** - The file name which should include the full path to the file in quotes.

TRACE BEGIN_RESTART

This begins a set of restart changes for the current model. This command identical to the model pop-up menu item **Begin Editing Restart**. Any subsequent changes made to the model before the STOP_RESTART command will be considered restart changes and will be included in any restart deck exports.

TRACE STOP_RESTART

This ends a set of restart changes and returns the model to a non-restart state. This is identical to the model pop-up menu item **Stop Editing Restart**.

TRACE SET_NAME <Mn> [name]

This command sets the name of either the current model or the model designated by the Mn parameter.

- <Mn> - An optional argument used to identify the model. Valid model labels are **M0-M9**.
- [name] - The new name of the model in quotes.

TRACE CREATE_CONSTANT <Mn> [constant name]

This command is used to create a new user defined constant with the specified name. An error will result if a constant with the given name already exists. The new user defined constant will be created in either the current model or the model designated by the Mn parameter.

- <Mn> - An optional argument used to identify the model. Valid model labels are **M0-M9**.
- [constant name] - The name of the user defined constant to be created in quotes.

TRACE CREATE_VIEW <Mn> [category name]

This command is used to create a new view containing all the components in the specified category. Acceptable categories can be found as the names of category nodes in the Navigator. The new view will be created in either the current model or the model designated by the Mn parameter.

- <Mn> - An optional argument used to identify the model. Valid model labels are **M0-M9**.
- [category name] - The name of the component category in quotes.

TRACE SETVALUE <Mn> [variable] [component] [indexes] [value]

This command sets a numerical value on a property inside the model. The property must be specified by a combination of label, component and node index as appropriate.

- <Mn> - An optional argument used to identify the model. Valid model labels are **M0-M9**.
- [variable]- The name of the variable (such as **rftn**) to modify.
- [component] - The id number of the component to modify. The type of component is automatically determined from the variable argument.
- [indexes] - The indexes argument can include from zero to four integers depending on the variable being modified. A complete listing of the variables available for modification and their parameters can be found in [Section 19.1, “TRACE Variable Indexes”](#) below. For

example, to set the void fraction of cell 3 in pipe 101 to 0.773, the following batch command would be used:

```
TRACE SETVALUE alp 101 3 0.773
```

Setting this same property in vessel 26, axial cell 2, radial ring 3, azimuthal sector 4 would use this command:

```
TRACE SETVALUE alp 26 2 3 4 0.773
```

Note the change in the number of integer parameters ('3' vs '2 3 4'). This distinction between 1D and 3D components is enforced by the batch interpreter and error messages will be issued if too many or too few parameters are given.

TEE based component indexing does not include the phantom cell index. So for example, a tee containing 3 main tube cells and 2 side tube cells will have 5 axial cell indexes, and 7 axial edge indexes. The following command will set the pressure of the first side tube cell of component 101 to 1000.0 psi:

```
TRACE SETVALUE p 101 4 1000.0
```

To set the liquid velocity at the inlet side of the side tube to 10 ft/s, enter the following command.

```
TRACE SETVALUE vl 101 5 10.0
```

- **[value]** - Either the floating point value to assign to the variable, or an *Inline File* argument. Data values must be input in the models current units, SI or British.

An *Inline File* is intended as a way to place a larger, more strictly formatted set of data as a single value argument. For the TRACE plug-in this capability is used by the control block table CBFTAB. For CBFTAB the Inline File is the same load format input required by the ASCII input deck for this table. For example, the following command would be used to replace the current CBFTAB in control block -3 (type 102) with a 2x3x4 table:

```
TRACE SETVALUE cbftab -3 2 3 4 <
* cbtblx1* 1.0 2.0s
* cbtblx2* 1.0 2.0 3.0s
* cbtblx3* 1.0 2.0 3.0 4.0s
* cbtbly * 8.0 8.0 8.0 8.0 8.0s
* cbtbly * 8.0 8.0 8.0 8.0 8.0s
* cbtbly * 8.0 8.0 8.0 8.0 8.0s
* cbtbly * 8.0 8.0 8.0 8.0 8.0s
* cbtbly * 8.0 8.0 8.0 8.0e
.
```

**TRACE SETNUMERIC <Mn> [numeric name] [variable]
[component] [indexes]**

This command is used to retrieve a variable value from a component and place it in a user defined constant or variable in the same model. The syntax is identical to that of SETVALUE with the exception of the numeric name argument, which contains the name of a user defined constant or variable.

- **<Mn>** - An optional argument used to identify the model. Valid model labels are **M0-M9**.

-
- **[numeric name]** - The name of a user defined constant or variable. Note that if the name includes spaces that it must be quoted in this batch command.
 - **[component]** - The id number of the component to modify. The type of component is automatically determined from the variable argument.
 - **[indexes]** - The indexes argument can include from zero to four integers depending on the variable being modified. A complete listing of the variables available for modification and their parameters can be found in [Section 19.1, "TRACE Variable Indexes"](#).

For example, to set the user defined constant named "VoidFrac" to the void fraction of cell 3 in pipe 101, the following batch command would be used:

```
TRACE SETNUMERIC "VoidFrac" alp 101 3
```

Setting the constant to the same property in vessel 26, axial cell 2, radial ring 3, azimuthal sector 4, would use this command:

```
TRACE SETNUMERIC "VoidFrac" alp 26 2 3 4
```

TRACE SETFRICS

This command provides the ability to convert a TRACE model's loss coefficients to be friction factors for both 1D and 3D hydraulic components. The conversion is performed for internal and external hydraulic edges as well as crossflow and leakage path connections. If the model is currently in FRIC format, the input will be left alone. The TRACE namelist variable IKFAC will be set to FRICS.

TRACE SETKFACS

This command behaves much like the above SETFRICS batch command except that all loss coefficients will be converted to K-Factors. The TRACE namelist variable IKFAC will be set to K-Factors.

TRACE CONVERT LEAKPATHS

This command will automatically convert all leak path connections to appropriate Single Junction equivalents.

19.1. TRACE Variable Indexes

Component Type	Variable Type	Index 1	Index 2	Index 3	Index 4
Hydraulic (1D/3D)	alp	Axial	<Radial>	<Azimuthal>	-
Hydraulic (1D/3D)	tl	Axial	<Radial>	<Azimuthal>	-
Hydraulic (1D/3D)	tv	Axial	<Radial>	<Azimuthal>	-
Hydraulic (1D/3D)	vl	Axial	<Radial>	<Azimuthal>	<Face>
Hydraulic (1D/3D)	vv	Axial	<Radial>	<Azimuthal>	<Face>
Hydraulic (1D/3D)	p	Axial	<Radial>	<Azimuthal>	-
Hydraulic (1D/3D)	pa	Axial	<Radial>	<Azimuthal>	-
Hydraulic (1D/3D)	tw	Axial	<Radial>	-	-
Hydraulic (0D)	pin	-	-	-	-
Hydraulic (0D)	tin	-	-	-	-
Hydraulic (0D)	alpin	-	-	-	-
Hydraulic (0D)	vlin	-	-	-	-
Hydraulic (0D)	tlin	-	-	-	-
Hydraulic (0D)	pain	-	-	-	-
Hydraulic (0D)	flowin	-	-	-	-
Hydraulic (0D)	vvin	-	-	-	-
Hydraulic (0D)	tvin	-	-	-	-
Valve	favlve	-	-	-	-
Pump	omegan	-	-	-	-

Table 19.1. TRACE Hydraulic Variable Indexes

Component Type	Variable Type	Index 1	Index 2	Index 3	Index 4
Heat Structure	rftn	Axial	Radial	Rod/Slab	-
Heat Structure	rpowri	-	-	-	-

Table 19.2. TRACE Heat Structure Variable Indexes

Component Type	Variable Type	Index 1	Index 2	Index 3	Index 4
Control Block	cbgain	-	-	-	-
Control Block	cbxmin	-	-	-	-
Control Block	cbxmax	-	-	-	-
Control Block	cbcon1	-	-	-	-
Control Block	cbcon2	-	-	-	-
Control Block	cbftab	Number of X1 Values	<Number of X2 Values>	<Number of X3 Values>	Inline File
Trip	setp	SETP (1-4)	-	-	-
Trip	dtsp	DTSP (1-4)	-	-	-

Table 19.3. TRACE Control System Variable Indexes

Note Indexes enclosed in "<",">" are optional

Chapter 20. Resource File Import/Export

The TRACE plug-in provides the ability to export an ASCII model from the Model Editor with each of its associated resources so that the model can be quickly tweaked and read back in without losing these resources. The resources can be any combination of views, numerics, or model notes. The exported model contains a numerics map which is written to the top of the file which identifies numerics that have been assigned within the model. Numerics written to the ASCII deck are renamed and appear in the exported model in a unix substitution format. The resource map will identify the actual names of the numerics and the unix substitution names they are mapped to.

Resource exports can be made by selecting the **Include Resource Map** option in the ASCII Export menu. This is displayed in below in [Figure 20.1, “ASCII Resource Export”](#).

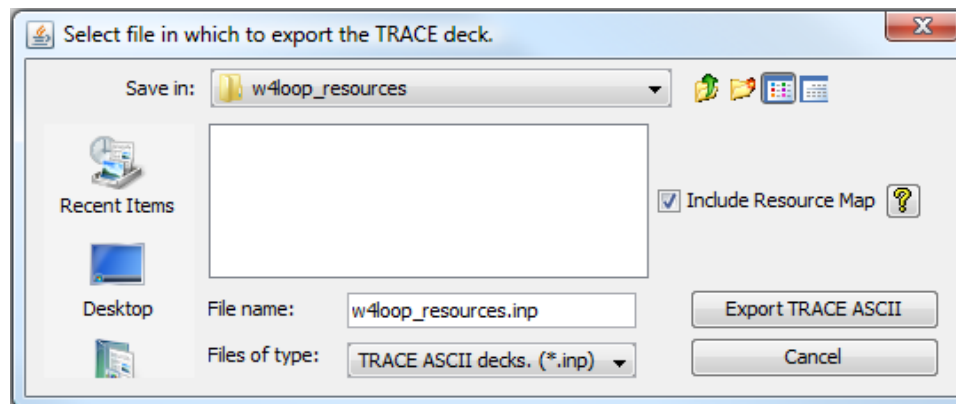


Figure 20.1. ASCII Resource Export

The plug-in will prompt the user to save the .med file for the model being exported if it has not yet been saved. The ASCII export model refers to the saved .med file for its resource information.

Once the exported model has been updated through the use of a text editor, it can be imported back into the Model Editor using the normal TRACE import routine. The plug-in will identify the ASCII model as a resource model and attempt to locate the resources for that model. The following dialog is provided when a resource model is imported:

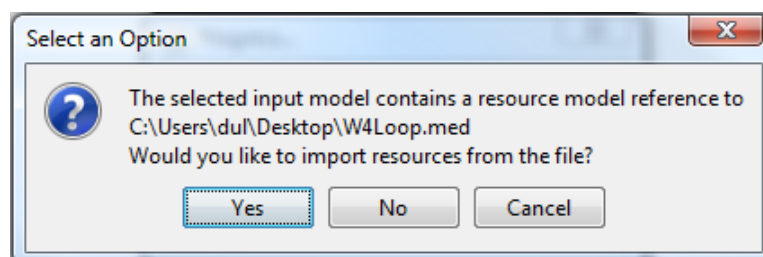


Figure 20.2. Resource Model Import

Once the resource file has been located the TRACE plug-in will gather all the information required to import the resources into the new model. The user has the option to decide which

of the resources will be included in the new model once the import has completed. The desired resources can be chosen from the provided configuration dialog . This dialog can be seen in [Figure 20.3, “Resource Options”](#).

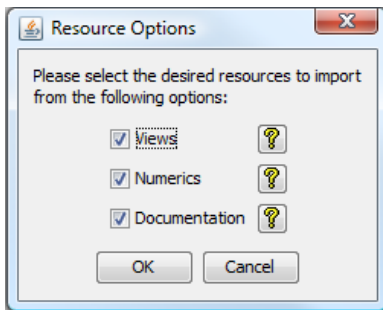


Figure 20.3. Resource Options

Once all of the desired resources are selected, the import will commence and create the new model. The model will contain any modifications made with the text editor during the resource export and will include each of the selected resources.

Chapter 21. PARCS Mapping

The PARCS Mapping component is a TRACE component which provides a user-interface and IO routines for creating maptab files. These files are used to provide a mapping of kinetics from the PARCS geometry to TRACE components.

21.1. Initial Map Creation

A PARC Mapping component can be created much like any other TRACE component. Selecting the New item from the right-click pop-up menu on the PARCS Mapping category node provides a initialization dialog which can be used to create a new mapping or import an existing mapping. The component's right-click pop-up menu Show ASCII item provides an ASCII view of the mapping data as it is currently configured.

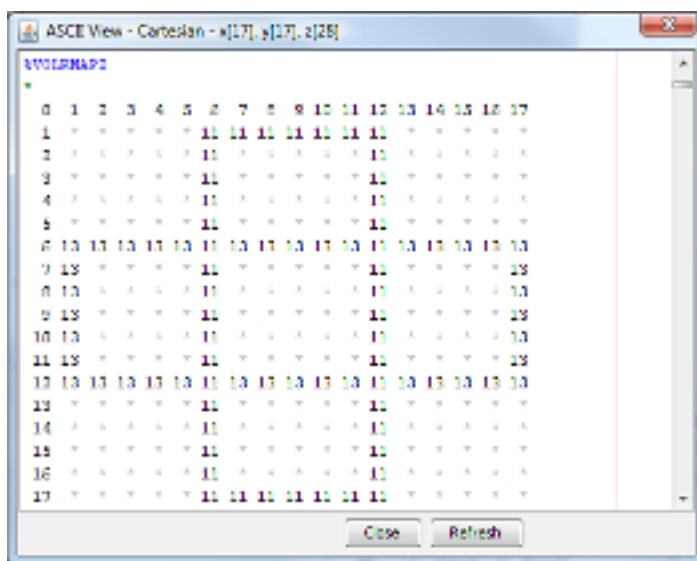


Figure 21.1. ASCII View

Creating a new PARCS mapping component provides the user with an initial map creation dialog. This dialog contains three separate ways to instantiate a mapping: 2D Hydraulic Mapping, Existing map Import, and Vessel Mapping. Each of these three modes require that a PARCS model be selected. The PARCS model selection editor will allow a PARCS model to be selected from within the Model Editor or from an external .med file.

2D Hydraulic Mapping will create a mapping file which couples PARCS to TRACE hydraulic components. The desired hydraulic components to couple should then be selected as well as their initial core locations. The hydraulic components will be defined at the selected planar locations across each of the axial levels. Each level can be reconfigured in the Map Coupling Editor for the selected map.

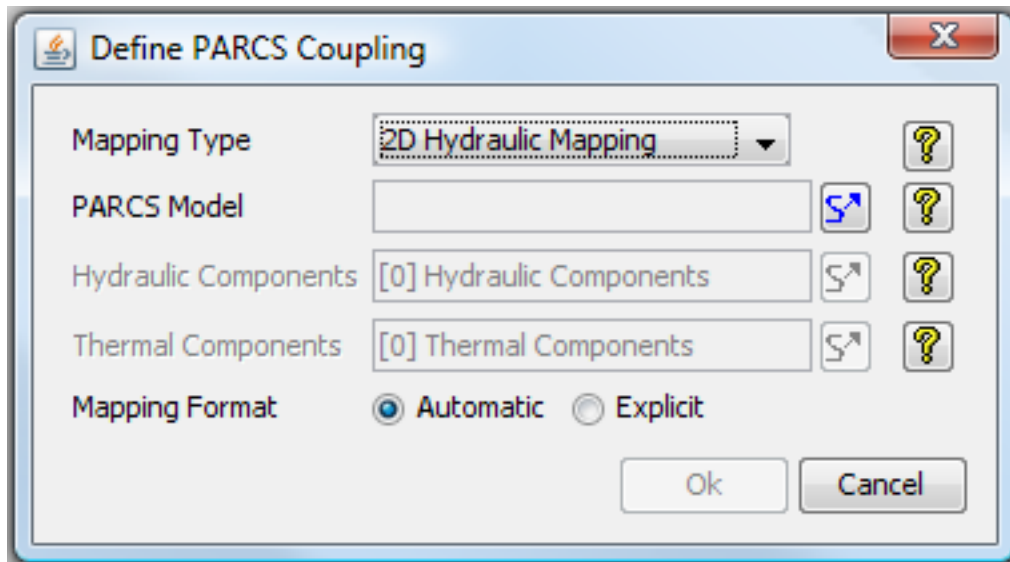


Figure 21.2. 2D Hydraulic Map Creation

The Vessel Mapping type provides the ability to define couplings from PARCS to TRACE vessel geometries. Once the vessel to map to has been selected, the "Core Bottom" and "Core Radius" properties should be defined. The Core Bottom property will determine from which axial level the coupling will connect to. The Core Radius property defines the highest radial node to connect to. These properties provide the ability to restrict the coupling to specific segments of the entire core.

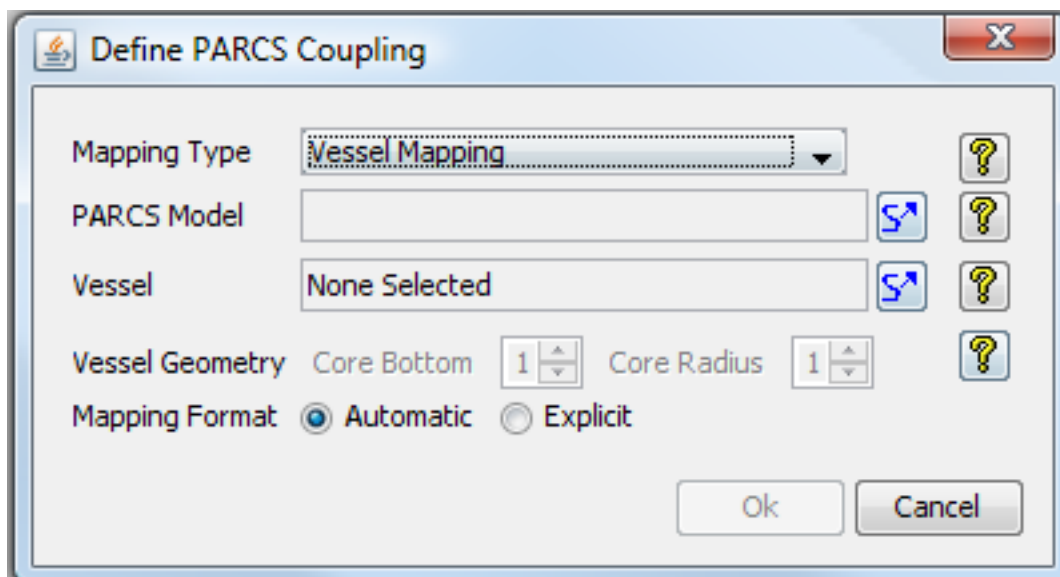


Figure 21.3. Vessel Map Creation

The Import Mapping type will import an existing coupling file. The mapping format should be specified according to the type of map which is to be imported. Currently the Automatic and Explicit coupling formats are supported.

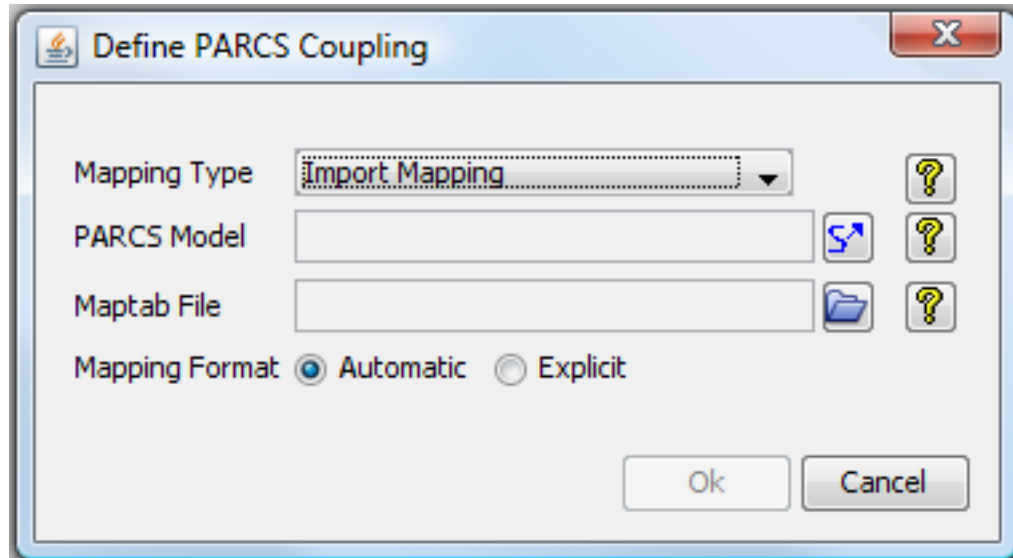


Figure 21.4. Import Mapping

The "OK" button will become available once all required information is provided. The new mapping will appear in the Navigator and additional editing can then be made from the components property view.

21.2. Mapping Configuration

After the initial map creation stage has been completed the map should then be configured. The property view will display the properties for the specific type of map created.

The Vessel mapping provides a series of options for defining the coupling between the TRACE and PARCS components. The Vessel Mapping editor can be accessed by selecting the "Hydraulic Components" property. The Vessel Mapping editor displays a planar nodal diagram for selecting specific regions in the core and the specific component mapping where values are assigned. When a planar region of the core is selected in the 2D Vessel Planar View, the 2D mapping view will highlight each of the locations where that planar index is set. Similarly, when the mapping nodes are selected, the 2D vessel planar view will highlight the planar locations of the selected mapping nodes. The dialog provides two ways to assign planar indices to the mapping nodes. The first way to assign a vessel planar index is to select on the mapping nodes you wish to set, select the assign button, and then use the crosshair to select the region from the 2D vessel planar view. The other way to specify a mapping node vessel planar index is to select on the desired nodes in the 2D mapping view and specify the numerical index in the provided table. This table also allows the specification of the mapping fraction. Selecting the clear button will remove the assigned planar index at the selected mapping node. The highlight option will cause the mapping view to highlight the current mapping selection.

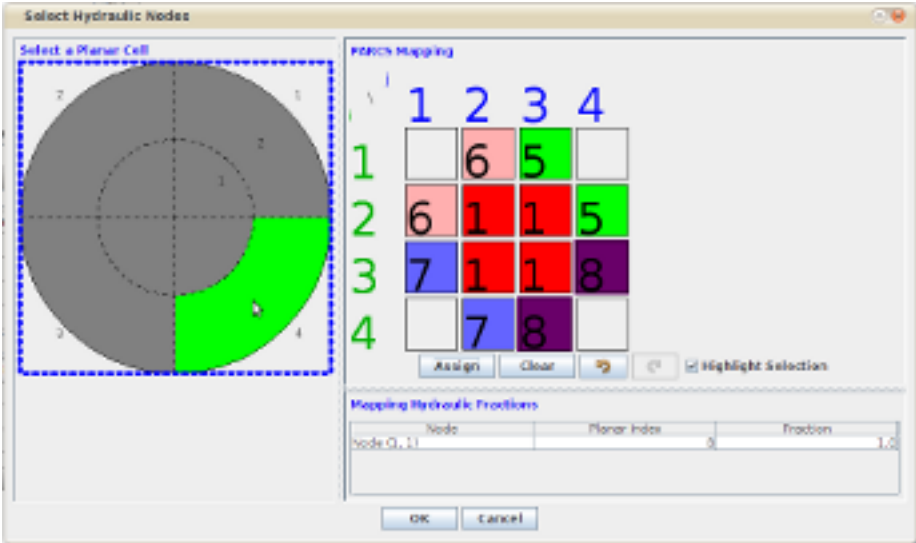


Figure 21.5. Vessel Mapping Editor

The Hydraulic mapping also provides a series of editors for defining map properties. Selecting the "Hydraulic Components" field will enable the 2D hydraulic mapping editor. The hydraulic editor is similar to the Vessel editor as it allows specific node assignment of components. The hydraulic editor includes Axial Levels and a list of available hydraulic components to set on the core planar nodes. Selecting on a hydraulic component in the provided list will cause the 2D mapping display to select each of the nodes which are assigned to the selected hydraulic component. Selecting on a 2D mapping node selects the hydraulic components in the list which are mapped to the selected nodes. The assignment of the hydraulic components to the 2D mapping view is similar to the vessel mapping dialog. Selecting the Assign button will set the cursor to a crosshair which can be used to select the hydraulic component which is to mapped to the selected nodes.

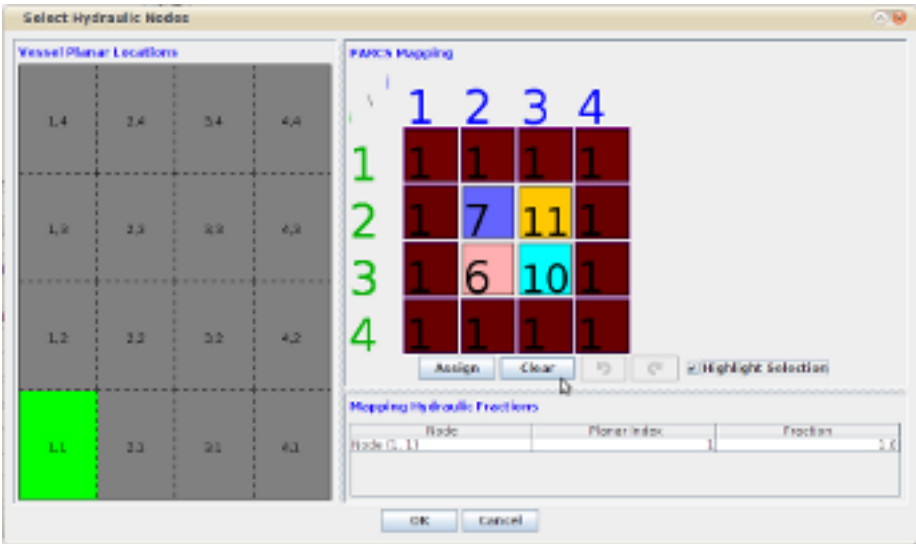


Figure 21.6. Hydraulic Mapping Editor

Both the Hydraulic Mapping editor and Vessel Mapping editor provide undo/redo capability. An undo or redo operation can be made by clicking on the provided buttons or pressing the control+z (undo)/control+y(redo) key combinations. The undo/redo state is tied to the dialog and has no affect on the system undo manager.

The Hydraulic mapping allows a core region to be split in either vertical or horizontal locations. Selecting the "Split Locations" editor from within the Hydraulic mapping property view, will enable the Split Location editor. The editor allows any number of split positions to be added. Split rows and columns will have a proper split fraction assigned. Splits can be defined per axial level.

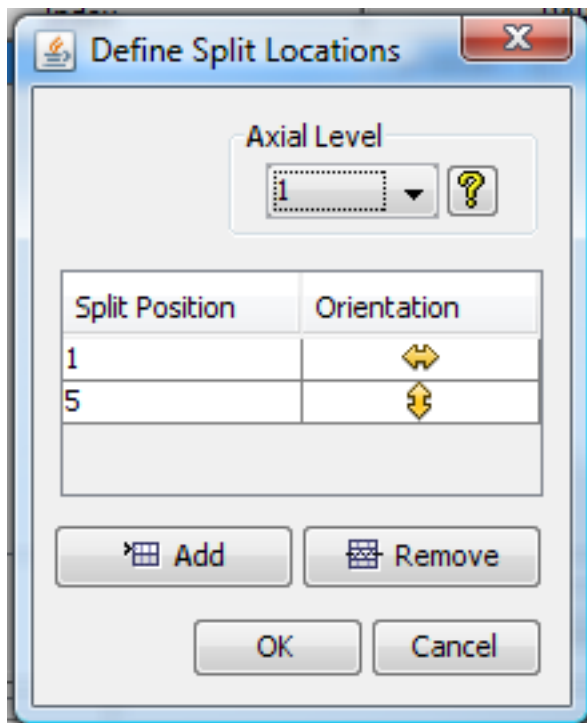


Figure 21.7. Split Location Editor

21.3. TRACE-PARCS Coupled Jobstream

Once the PARCS mapping has been created it can easily be used in a Engineering Template Job Stream. The PARCS Mapping component has an "active" property. Setting this property to true will ensure that the specified mapping will be exported to those Job Steps which require the mapping. Only a single mapping can be enabled in a given TRACE model.

The following set of steps describe how to set up an Engineering Template Job Stream in order to run a coupled TRACE-PARCS problem:

1. Create or import a PARCS model using the PARCS plug-in and save the model (This step should be repeated if more than one coupling case is to be run, as in a steady-state and transient)
2. Create or import a TRACE model. (If creating a new model and the case should be coupled, set namelist itdmr to PARCS)
3. Create a new PARCS Mapping component.
4. Configure or import a coupling map.
5. Set the map active. (This mapping will be provided to each of the coupling runs)

6. If using a TRACE restart model, create a new Case component and import the restart file. (If doing more than one restart, repeat this step)
7. Save the model.
8. Create a new Engineering Template Model.
9. Create a reference model component for each of the TRACE and PARCS models we saved earlier.
10. Create a new Job Stream component.
11. In the Job Stream component, create an additional TraceModel node for each of the restart cases the trace model supplies. Select the restart case property and the case it represents for each model node.
12. Create a new Trace Step for each of the runs which need to be made. Set their names and the names of the TRACE steps to the type of run they represent (standalone/coupled ss/coupled tr/etc)
13. If the first run in the coupling job is a standalone trace job, drag the basic trace model node and the standalone trace step to the view.
14. Connect the standalone Trace Model node to the tracin input of the standalone Trace step.
15. Drop the first trace restart model node onto the view along with the coupled PARCS model.
16. Add the associated trace step to the view
17. Connect the Trace restart model node to the tracin input on the Trace Step (A parcs_inp input appears on the step if the restart model has itdmr=1)
18. Connect the PARCS model node to the parcs_input input on the step.
19. Connect the trctr output node of the previously added step to the trcrst node on the newly added step.
20. Drop the second trace restart model node onto the view along with the coupled PARCS model (These steps assume this is a transient which accepts a TRACE and PARCS restart file).
21. Add the associated trace step to the view.
22. Connect the Trace restart model node to the tracin input on the Trace Step.
23. Connect the PARCS model node to the parcs_input input on the step.
24. Connect the trctr output node of the previously added step to the trcrst node on the newly added step.
25. Connect the parcs_rst output node of the previously added step to the parcs_rsti node on the newly added step.

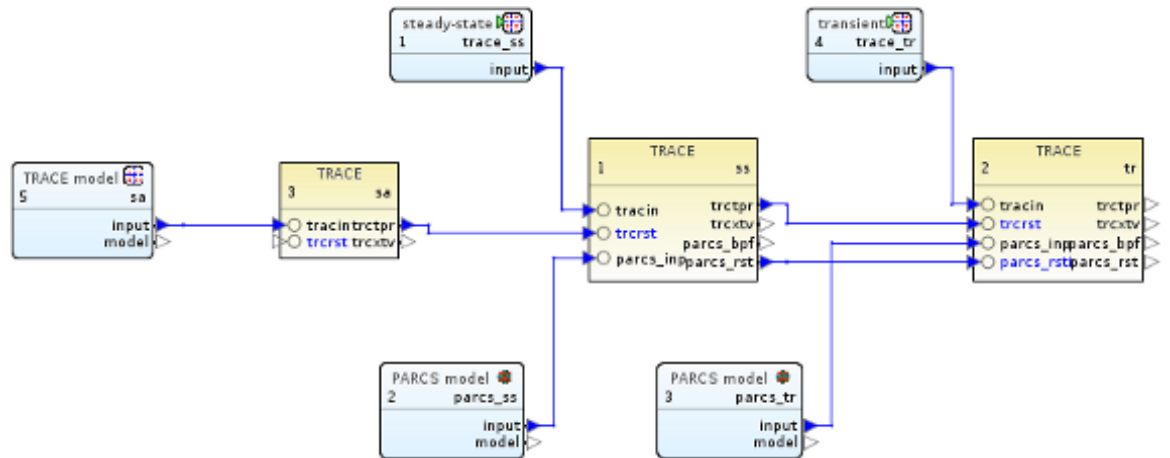


Figure 21.8. TRACE-PARCS Coupled Engineering Template

Chapter 22. FRAPCON Fuel Rod Import

This section describes the FRAPCON fuel rod import capabilities provided by the TRACE plug-in. Currently two modes exist, single fuel rod and multi-rod state condition import. These features will be described in the following sections.

22.1. Single Fuel Rod Import & Creation

The TRACE plug-in provides the ability to create a new TRACE heatstructure from the output of an existing FRAPCON run. A wizard has been provided to assist the user with this operation. To access the feature, right-click on the Thermal node in the Navigator. This will provide a pop-up menu option labeled "Import FRAPCON Fuel Rod". Selecting this option will open the FRAPCON to TRACE Fuel Rod Import Wizard.

The first step in the import process is to identify a source of FRAPCON data to extract information. This information is used to initialize a new TRACE heatstructure. Two modes of source selection are provided: Local File selection and Job Stream selection. The Local File selection mode provides the ability to select files directly off of the local filesystem. This process requires the selection of both a FRAPCON output and plot files.

Note Currently only FRAPCON cases which are run with NPLOT=2 are supported. This is because 2D burnup data is required and is only available when NPLOT = 2.

Generally, the output of a FRAPCON run ends with the .out extension. This file should be specified in the text field provided next to the Output File text label. A local file selection editor is provided when the user selects the folder icon next to the text area. The Plot file should be specified in the Plot File text area. The FRAPCON import wizard supports both the .plot and .pib plot file formats and either can be input into the field. Pressing the Next button at the bottom of the dialog will begin the process of extracting information from the selected files or job. Once complete, a new configuration panel in the wizard will be provided.

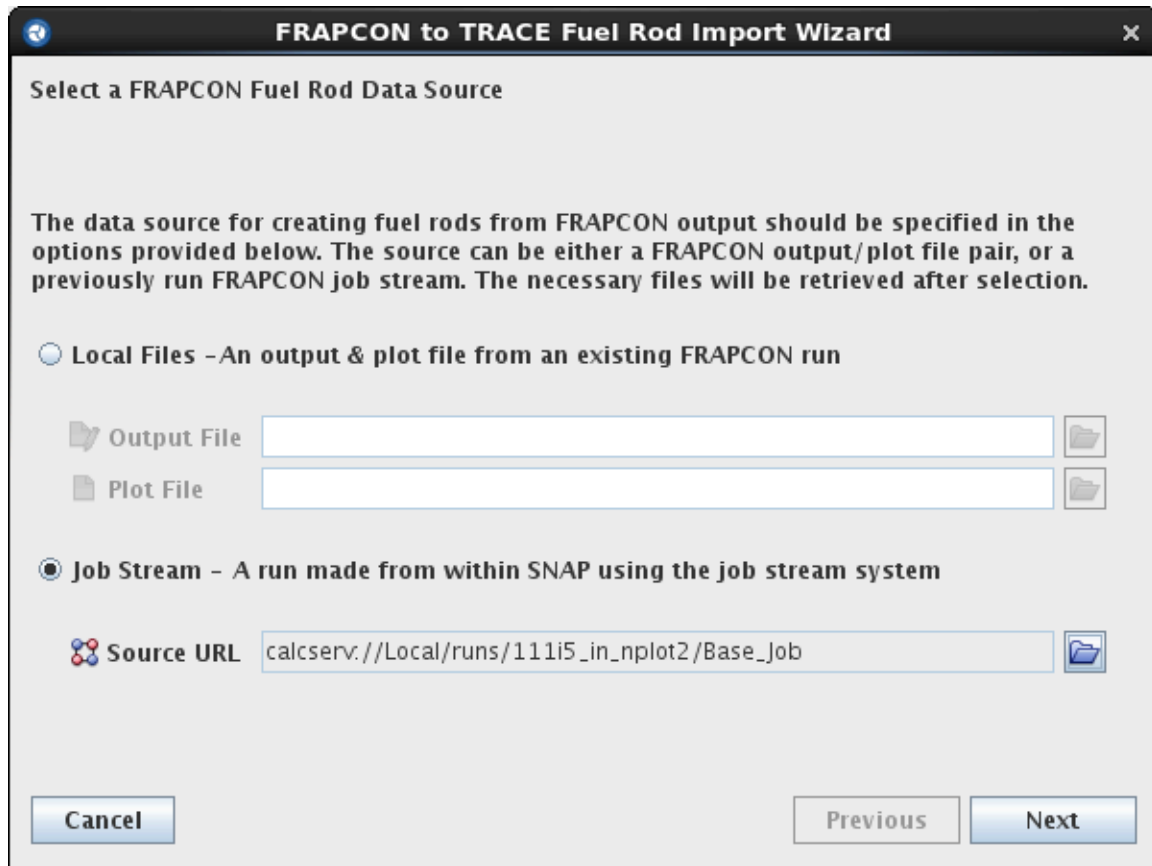


Figure 22.1. FRAPCON-TRACE Fuel Rod Import Source Selection UI

The configuration panel in the wizard is required for specifying the timestep in which data will be extracted. A drop-down combo box is provided for selecting the desired timestep. Selecting a timestep in the will will provided radial power and intial radial temperature plots in the lower portion of the panel. These are provided as visual aids for determing some basic information about the selected timsteps. Once the desired timestep is specified, selecting the Finish button will be construction of the new heatstructureT

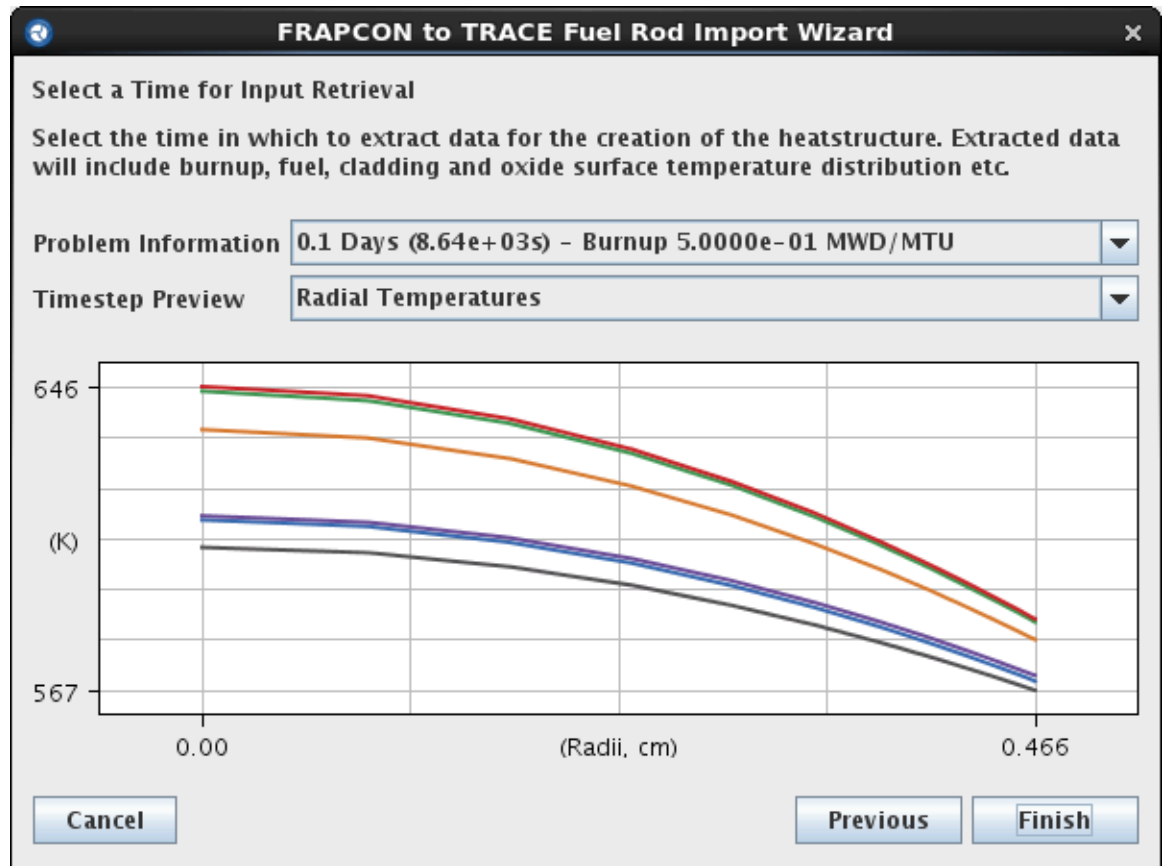


Figure 22.2. FRAPCON-TRACE Fuel Rod Import Configuration UI

New heatstructures will be automatically added and selected in the Heatstructure category of the Navigator Tree. These new heatstructures are given a unique component number and have a component description of "Imported FRAPCON Heatstructure".

22.2. Multiple Fuel Rod Importing

The TRACE Plug-in supports mapping multiple FRAPCON fuel rod output files to heat structures, and importing the output of those FRAPCON runs before executing TRACE as part of a Job Stream. The process begins by defining the future FRAPCON run output sets that will be generated. This is accessed through the Thermal node in the Navigator, by opening the FRAPCON Runs property editing dialog, shown in [Figure 22.3, “FRAPCON Fuel Rod Import Rod Dialog”](#). Existing output files can be imported in this dialog by pressing the Import button. This will allow selecting the FRAPCON output and plot files, and select an output time.

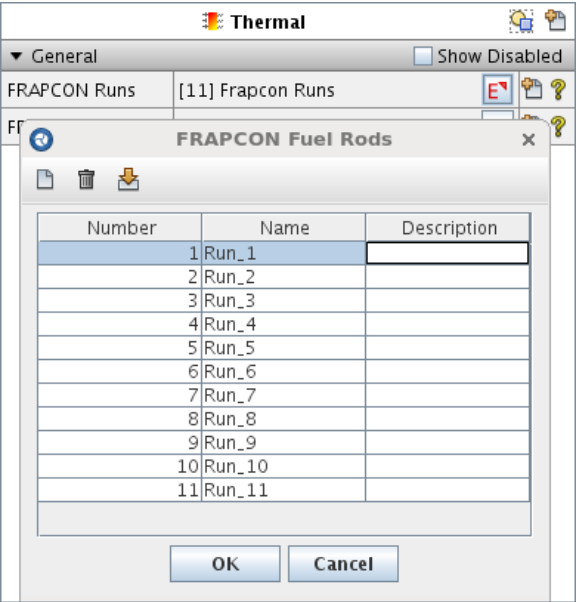


Figure 22.3. FRAPCON Fuel Rod Import Rod Dialog

Once the FRAPCON runs are defined, they must be mapped to the fuel rod heat structures in the model. This mapping is defined through the FRAPCON Mapping property editing dialog accessed through the Thermal node in the Navigator. This dialog, shown in Figure 22.4, “FRAPCON Fuel Rod Mapping Dialog”, allows mapping one or more heat structures to FRAPCON rods. When multiple FRAPCON runs are mapped to a TRACE heat structure, a multiplier is used to average the FRAPCON values before applying the values to the heat structure. An error will be reported if the multipliers for a heat structure do not sum to 1.0.

The fuel rod mapping may be imported from a CSV file by using the import button. This mapping will create any missing FRAPCON runs encountered during the import.

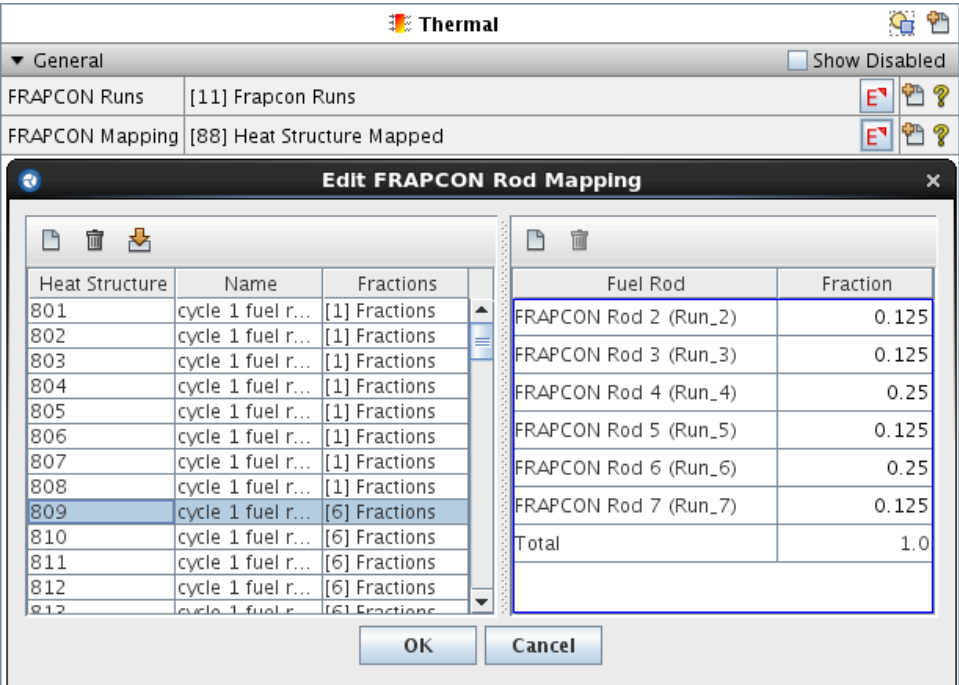


Figure 22.4. FRAPCON Fuel Rod Mapping Dialog

22.2.1. FRAPCON Fuel Rod Job Stream

If the FRAPCON fuel rod mapping data has been defined, TRACE job steps will allow activating the FRAPCON Output Retrieval properties. These properties, shown in [Figure 22.5, “FRAPCON Fuel Rod Job Stream Properties”](#), are only available if the Rod Mapping has been created, and the FRAPCON plug-in is installed. They allow activating the FRAPCON input nodes on the TRACE job step, and entering the output time desired. The output time can be defined in Days or allow selecting the last timestep.


Retrieve FRAPCON Data	<input checked="" type="checkbox"/>	<input checked="" type="radio"/> Time	<input type="text" value="550.0"/> (days)	 
	<input type="checkbox"/>	<input type="radio"/> Use Last Timestep		

Figure 22.5. FRAPCON Fuel Rod Job Stream Properties

If the Retrieve FRAPCON Data option is activated, the TRACE Job Step will enable two additional input nodes, one for FRAPCON Plot files, and one for FRAPCON output files. These are set inputs that will require the a file entry for each of the FRAPCON fuel rods defined in the TRACE model. These files must be defined using external file set objects inside a TRACE model, but can be defined using parallel FRAPCON steps in an Engineering Template. [Figure 22.6, “FRAPCON Fuel Rod Job Stream”](#) below displays a job stream that includes a parametric FRAPCON run providing the files for a TRACE job.

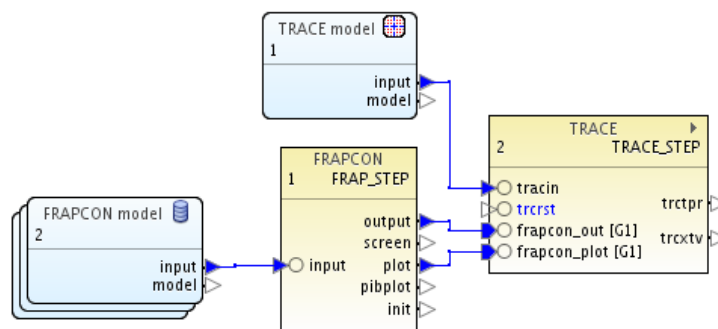


Figure 22.6. FRAPCON Fuel Rod Job Stream

22.2.2. FRAPCON Fuel Rod Import Data

The FRAPCON Fuel Rod importer converts FRAPCON output values into TRACE heat structure input. The following properties are retrieved from the input echo portion of the FRAPCON output file:

TRACE Property	FRAPCON Values
Pitch to Diameter Ratio [PDRAT]	Reactor Conditions : Rod Pitch [PITCH] / Rod Design : Cladding Outer Diameter [DCO]
Fuel Pin Rougness [RFCLAD]	Fuel Pellet : Pellet Surface Roughness [ROUGHF]
Cladding Rougness [ROUGHCLAD]	Rod Design : Surface Roughness [ROUGHCLAD]
Fraction of Theoretical Fuel Density [FTD]	Fuel Pellet : Fuel % Theoretical Density [DEN] / 100.0
Rod Plenum Height [HFRPLEN]	Rod Design : Plenum Length [CPL]
Pellet Shoulder Width [DISHSD]	Fuel Pellet : Dish Shoulder Width [DISHSD]
Pellet Dish Depth [HDISH]	Fuel Pellet : Dish Height [HDISH]
Pellet Height [ZPELL]	Fuel Pellet : Pellet Height [HPLT]
Sintering Temperature [TFSINT]	Fuel Pellet : Sintering Temperature [TSINT]
Maximum Density Change [RSNTR]	Fuel Pellet : Density Increase [RSNTR]
Gadolinia Concentration [GADC]	Fuel Pellet : Gadolinia Weight Fraction [GADOLN]

Table 22.1. FRAPCON Input Echo Values

Note The Gadolinia Weight Fraction is a boundary condition value in FRAPCON, and will be applied across all of the TRACE heat structure nodes.

The following properties are retrieved from the FRAPCON output file major edit data for the selected time step:

- Spring Volume Fraction [VFSPRING]
- Crud Thickness [CRUDTO]
- Gap Gas Mass Fractions [GMIX]

The following properties are retrieved from the PARCS Plot File for the selected time step:

TRACE Property	FRAPCON Plot Values
2D Burnup [BURN2D]	fburnup
Axially Averaged Burnup [BURNUP]	burnup
Cladding Oxide Thickness [OXLAYER]	oxide
Cladding Creepdown [UCRPDOWN/ CREEPDOWN]	cladidef
Fuel Pin Strain [UFSWELL/FUELSWELL]	fswell
Fuel Densification [FUELDENS]	fdensif
Hydrogen Concentration [CEXH2]	h2

Table 22.2. FRAPCON Plot File Values.

Note Axially averaged properties (e.g. UCRPDOWN) will be initialized as well as the associated axial array values (e.g. CREEPDOWN). This provides support regardless of the models namelist variable values.

Chapter 23. TRACE Parallel Task Sets

The TRACE plug-in uses Parallel Task Sets to support TRACE's course-grained parallel (multi-task) mode of operation provided through the Exterior Communications Interface (ECI). Each Parallel Task Set represents a set of TRACE input that will be run by multiple TRACE processes in parallel. Users of this capability are strongly encouraged to study the "Multi-Task Mode of Operation" section of the TRACE User's Manual as well as the ECI documentation and examples included in the TRACE distribution.

Note This feature is currently only available for the developmental version of patch 5 ("V 5.0 P5 Dev").

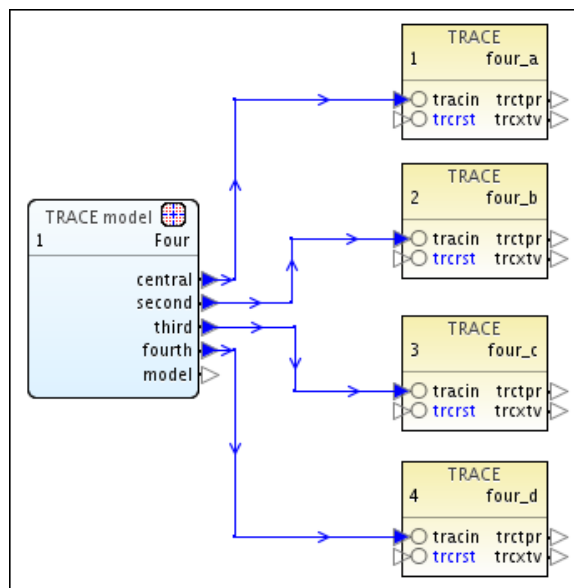


Figure 23.1. Example Multi-Task Job Stream

Parallel Task Set components contain a set of Parallel Task definitions that determine how the model will be split up. Any number of Parallel Task Sets can be created and each represent a different way to split the same model. These task definitions are used to generate parallel input files as part of a Job Stream submission. During Stream submission, the TRACE model will be automatically split into multiple input files based on the component selected for each Parallel Task. The splitting process will also create any EXTERIOR components required in each of the generated parallel input files.

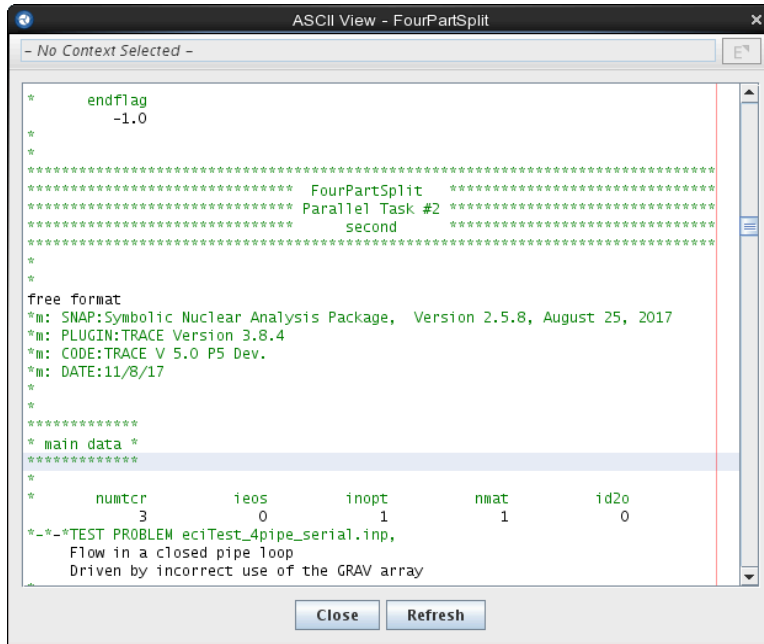


Figure 23.2. Example ASCII Viewer

The contents of a Parallel Task Set can be viewed by using the "Show ASCII" right-click pop-up menu item on that task set in the Navigator. This will show the ASCII input files that will be for each of the Parallel Tasks defined for that set. A comment header will be included at the top of each section of the file indicating which task is being written.

23.1. Getting Started

The following is a brief description of how to create and use a parallel task set for an existing model. Refer to [Section 23.2, "Parallel Task Dialog"](#) to for more information about the Parallel Task dialog.

1. Create a new "Parallel Task Set" component in the Navigator.

Task sets represent a set of parallel input files that will be generated by splitting the model. Any number of task sets can be created to represent a different way to split the same model. One task set could be used in a Job Stream to speed the execution of a large model. Many task sets could be used in a Job Stream to compare the performance difference of splitting the model different ways.

2. Edit the Parallel Tasks property.

This will open the Parallel Tasks editing dialog.

3. Create one or more "Parallel Tasks".

Each parallel task represents a separate input file and thus a separate TRACE process. For example: Splitting the stream generator into a separate model.

4. Add the desired components to each task.

For example: The stream generator's pipes, tees, and heat structures. All unselected components (i.e. everything else in the model) will be placed in the "Central Process".

Note Components can also be added via the right-click pop-up menu much like adding to a Sub-System.

5. Create a new Model Node.
6. Set the Model Node's "Parallel Task Set" property to the newly created task set.

This will replace the normal "input" connection on the Model Node with the Parallel Task inputs. The "Central Process" input will always be listed as "central". The remaining inputs will be named to match each of the Parallel Tasks defined in the set.

7. Create a new TRACE Job Step for each parallel task in the set.

One for "central" and one for each of the satellite processes.

8. Connect the Model Node to each TRACE Job Step.

9. Submit the Job Stream.

Note The "Parallel Driver" program used to coordinate analysis codes using the Exterior Communications Interface is built into the Job Stream system. It is not necessary to run a separate driver instance.

23.2. Parallel Task Dialog

This dialog is used to manage how the model will be split into parallel tasks. The list of *Parallel Tasks* is on the left with the Central Process always at the top. Tasks can be added or removed by pressing the corresponding buttons just above the list.

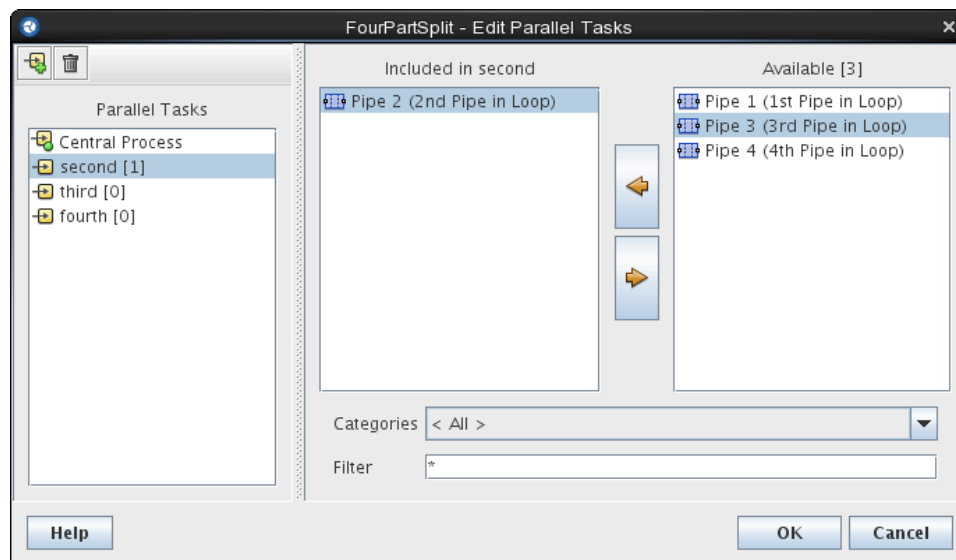




Figure 23.3. Parallel Task Set Dialog

On the right side of the dialog is the list of components that are included in the currently selected task and the list of components that can be added. Selecting a task in the parallel tasks list on the left will update the included and available lists on the right. Components can be added to and removed from the currently selected task with the Add () and Remove () buttons. This will move them from the available list to the included list, and vice versa. The filtering options below the included and available list can be used speed the selection of particular components.