What's New in SNAP 2.0

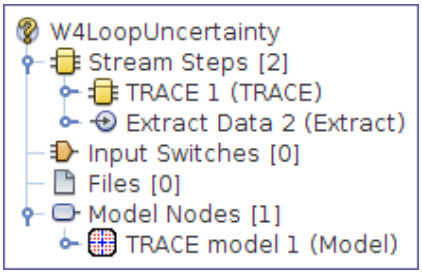
The release of SNAP 2.0 brings a host of new features and improvements to the SNAP application suite.

- SNAP now supports Job Streams, a means of automating complex application interactions.
- Configuration Tool has been overhauled to support defining applications and platforms.
- An update to Job Status allows displaying the run-time progress of streams executing on High Performance Computing (HPC) environments.
- The ModelEditor now supports headless operation for execution on a server machine.
- The ModelEditor has a new Engineering Template plug-in that allows interacting with multiple models at the same time.

Please refer to SNAP's built-in help for more information on these new features.

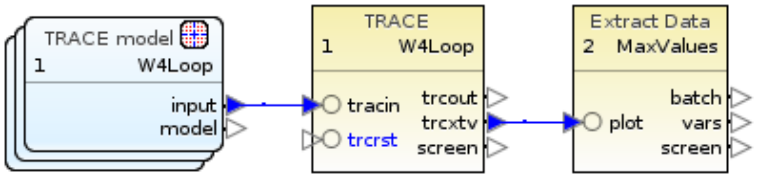
Job Streams

A Job Stream is a component in a pre-processor plug-in that defines the work flow for executing analysis applications based on the current model. A Job Stream is defined by stream steps (single application executions), files, switches (logical operations used to control job stream work flow), and reference models. A model may have more than one job stream, each stream defining a different set of applications to be executed.



Individual streams may contain any number of steps, input switches, files, and models. Stream components are created and edited like any standard SNAP components. Right-clicking the Stream Steps category will allow creating new steps in the stream, while right-clicking a step will allow operations such as cut, copy, remove, etc.. Selecting a step, switch, file, or model will display its properties in the Property View, and all stream components may be added to a 2D View.

Steps, as mentioned above, represent the execution of a target application. In the 2D View, a step is represented as a series of inputs on the left and outputs on the right. Creating complex application flows is as simple as connecting the outputs of one step, file, or model to the inputs of another step. The connections between stream components defines the execution of the stream.



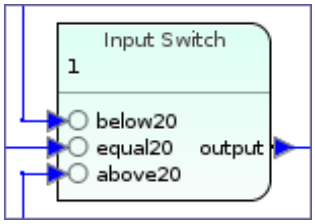
In this example, a TRACE input file exported by the SNAP W4Loop model is used by an invocation of a TRACE executable. Once complete, the resulting XTV file is fed into an Extract Data step, which is a special case of AptPlot used to extract variable data. Any of the resulting outputs could be fed into other steps or placed in remote storage.

Job Streams: Parametric Cases

Job Streams provide numerous options for parametric job execution. Selecting a Numeric Combination or Tabular stream type for the stream will allow executing each step down-stream of a parametric model once for each parametric task. Numeric and Tabular parametric both referenced user-defined numerics as independent values: each parametric task replaces the value of the numeric before exporting the model. With a Numeric Combination, every permutation of independent variable values defines a parametric tasks. Tabular parametrics specify a set of independent variables and an explicit value for each variable in every task.

Additional parametric controls are available in the form of manual parametric combinations, file-set inputs, task filtering, and more.

Job Streams: Input Switches

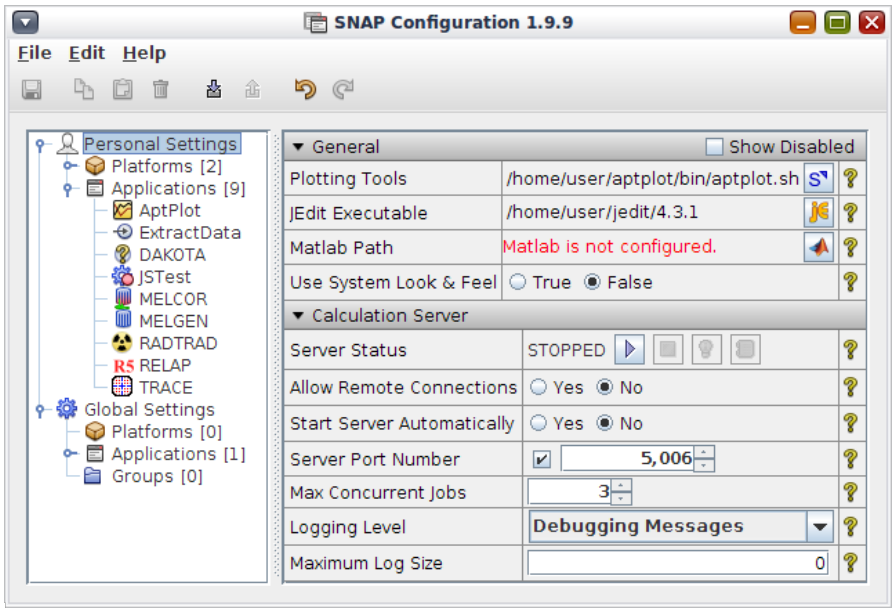


Input Switches allow branching between input sources based on the value of a user-defined numeric. Real and integer numerics can branch based on numeric conditions, including less than, greater than, equal to, inclusive ranges, and exclusive ranges. Each of these ranges also support including values directly at the boundary points. Boolean type numerics can be filtered to specific true/false values.

Coupled with editable numerics placed directly in a 2D View, input switches become a powerful tool for steering the execution of a job stream.

Configuration Tool Changes

The Configuration Tool has been updated to support defining Job Stream applications and the platforms on which streams execute. Opening the Configuration Tool now provides a screen similar to the following:



Configuration Tool usage now follows a pattern similar to the ModelEditor: select a category or definition on the left, and the current configuration is displayed for modification on the right. The standard SNAP configuration options are available from the Personal Settings category. Platforms define the execution environments available to job streams, which can be a standard Calculation Server or a High Performance Computing (HPC) environment, such as a cluster. Applications define executables available on each

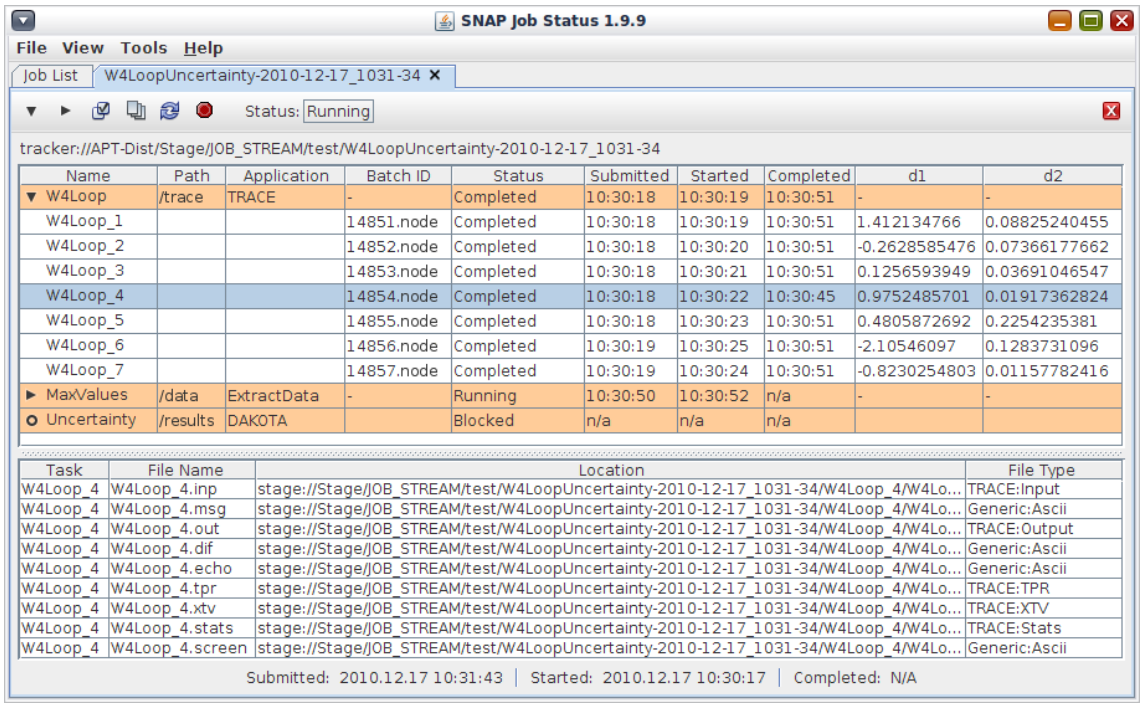
platform, their command line arguments, options, and flavor.

Note: HPC support requires specialized plug-ins tailored to the environment's batch queuing system, and a Tracking Server to monitor stream execution. See the Job Stream documentation for more information.

In addition to the above changes, Configuration Tool now supports copying and pasting definitions. In addition, changes may be reverted and restored with the newly added Undo/Redo functionality.

Job Status Changes

Job Status has been updated to monitor job streams running on an HPC platform that supports Tracking Server communications. Viewing a Tracking Server stream opens a specialized tracking tab, seen below. Within the table, streams are broken down into highlighted steps. Parametric steps can be expanded to monitor individual tasks, or collapsed to obtain a higher-level view of stream execution. Additionally, the stored files of a single-task step or parametric task can be displayed in the Files Table by selecting that step or task.



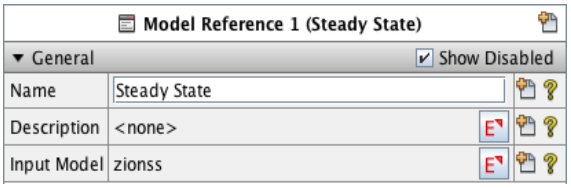
Also new in Job Status is enhanced file display for streams run on the user's machine. Images, Documents, PDF files, and Plot Files associated with a job can now be opened in the standard application registered with their corresponding file type. View File actions are available from the right-click pop-up menus of steps for Calculation Servers streams and File Table rows in Tracking Server streams.

ModelEditor Headless Mode

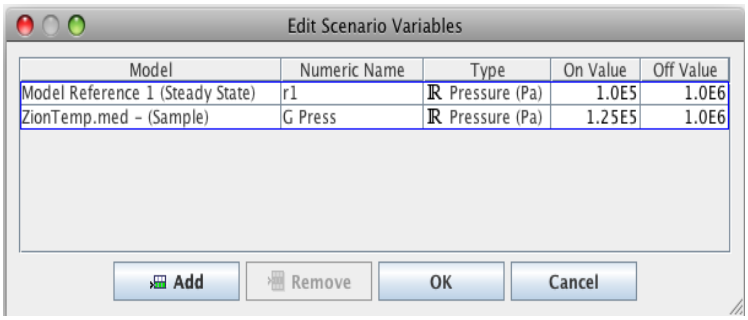
The ModelEditor now supports headless execution, allowing the execution of ModelEditor batch scripts on server machines. An MEBatch executable is included as part of the installation process: mebatch.exe on Windows and mebatch.sh on Mac OS and others. Invoking this executable with a batch script as an argument will run the ModelEditor in headless mode, executing the batch commands in the file. While running, all console output normally directed to me.screen will be written directly to the console. Once the commands in the batch file have all been executed, the MEBatch executable exits.

Engineering Templates

The ModelEditor now includes the Engineering Template plug-in. Engineering template models contain no model data themselves, but allow high-level access other models. These models can then be submitted to the calculation server using job streams. This allows for parallel model execution of coupled analysis codes (TRACE/PARCS or RELAP/PARCS). Additionally the shared numeric values of the underlying models may be modified through the template model. The numerics can be used to provide restricted access to properties that can control how the analysis will proceed.



An engineering template model contains one or more model references to underlying models. The models are selected by choosing the locally accessible ModelEditor document through the **Input Model** property on a Model Reference component. The referenced models are opened in the background of the Model Editor. This allows the engineering template to access the shared numerics of the referenced models. If a reference model is opened explicitly by the user, the background references are closed. Each engineering template model will contain its own copy of the model loaded in the background. However a template model may only contain one reference to a given model.



Shared variable values for the underlying model can be grouped together using Scenarios. A scenario contains a list of shared values for the underlying models that have an *On* value and an *Off* value defined. When the scenario is selected, the pre-defined *On* value is set for each of the variables. Mutually exclusive scenarios can be grouped together in a Scenario Group. The scenario groups ensure that only one of their included scenarios is active at a time. The grouped scenarios do not have to reference the same variables. This allows for a template model to contain a number of different scenario groups that when combined together fully define a transient case. For example, one scenario group might allow the user to select between three different transient types, while a second scenario group controls an initial valve position, and a third group controls an initial pump state.

Engineering templates also contain Global Variables. Global variables allow the template user to directly set the value of one or more underlying model variables from the engineering template. Global variables may be integers, booleans or reals, and can be inserted directly to the view just like shared numerics. The global reals have a single set of pre-defined engineering units, which are converted into the correct plug-in units automatically. Global variables can be selected for scenarios, parametric job streams, and job stream component logic such as input switches, and job step filters. It should be noted that an error message will be displayed if both a global variable, and one if its associated model variables are defined in the same scenario or parametric stream type.

Engineering template models may be displayed in either SI or British units, without changing the values inside the underlying model. Editing a Global pressure in Pascals which sets the initial pressure in a model using PSI will result in the entered value being converted to PSI in the resulting model. All displays of engineering units inside the template model will be in the templates units.