Riverware

Technical Documentation Version 7.3

Model Files

Center for Advanced Decision Support for Water and Environmental Systems (CADSWES)
UNIVERSITY OF COLORADO **BOULDER**

Model Files
Table of Contents

Model Files

1. Introduction

RiverWare models are saved as ASCII files in the TCL (Tool Command Language). These files could be viewed and edited with a text editor outside of RiverWare, but their complexity and frequent format changes make direct editing difficult and dangerous. CADSWES neither recommends nor supports direct editing of model files.

2. Loading Models

Loading a new model clears any existing model on the workspace after prompting the user for verification. In the case of a blank workspace, no verification is needed. To load a model:

- Select **File ➤ Open Model...** from the main workspace menu bar or click on the Open Model File toolbar button. A File Chooser dialog appears.

The file chooser filters by the commonly used file extensions for Model Files:

- Model Files: *mdl, *.mdl.gz *.rw and *.rw.gz (default)
- gzip Files (*.gz)
- All *

For RPL sets, the filters include

- Ruleset files (*.rls, *.rls.gz)
- Goal Set files (*.opt, *opt.gz)
- Global function sets (*.gfs, *gfs.gz)
- gzip files (*.gz)
- All (*)

In addition, the user can use asterisks (*) as wildcards in the File name to further filter.Once a file has been selected and the Open button has been clicked, the cursor turns into an hourglass while the model is loading. Once the model is loaded a diagnostic window pops up with information regarding the date and time when the model was last saved, by whom the model was saved, and which version (release) of RiverWare.

If the model is configured to display background images in one or more of the Workspace views, then after the model is loaded, RiverWare reads and displays the background image files. While RiverWare is loading these images, a gray "placeholder" background is displayed in the appropriate views and an

animated "busy" icon is shown at the bottom of the Workspace. All of the usual workspace interaction is still possible while images are being loaded. Note that for many background images, the loading process is so brief that it is not noticeable.

2.1 Reloading Models

On the **Workspace**, the **File ➡ Reopen Model** gives the user previously opened models and directories. This is a quick and easy way to reopen a model. For rulesets there are options to **Reopen** and **Reopen and Load** previously viewed rulesets.

Reopen Model	▶	C:/Temp/periodicTest.mdl.gz
Open Model...	Ctrl+O	R:/prerel/rt/Rules/ruleLanguage.mdl.gz
Save Model	Ctrl+S	R:/prerel/rt/Sim/wqLayeredWithCanalConfl.mdl.gz
Save As...	Ctrl+Shift+S	R:/prerel/rt/Sim/sedMod.mdl.gz
Save As Baseline...		R:/prerel/rt/Rules/iterativeMRM.mdl.gz
Export Objects...		R:/Prerel/rt/Sim/
Import Objects...		C:/Temp/
Export Workspace Image	▶	R:/prerel/rt/Rules/
Export/Import Coordinates...		R:/prerel/rt/Sim/
Print...		G:/RulebasedSimulation/models/
Model Info...		
Summary Info...		
Exit RiverWare	Ctrl+Q	

Listed in this menu are previously opened models and a list of recently accessed directories. Selecting a directory item (shown with a yellow folder icon) opens up a file chooser initialized to the specified directory. Also shown is a single user-configured "default" directory (for the specific type of file), based on an environment variable. This directory is shown as a green folder. If this is defined, the file chooser opened using the **File ➡ Open** menu will open to that default directory the first time within a RiverWare session. Supported environment variables include:

Type of File	Environment Variable
Model Files	RIVERWARE_DFLT_MODEL_DIR
Rulesets (RPL)	RIVERWARE_DFLT_RULESET_DIR
Optimization Rulesets	RIVERWARE_DFLT_OPTSET_DIR
SCT Files	RIVERWARE_DFLT_SCT_DIR

2.2 Save History, Run History and Model File Comments

When a model is loaded, a diagnostic window opens with the **Model Save History**.   The model save history appears as a brown diagnostic message with information regarding the date and time when the model was last saved, the username of person who save the model, and the RiverWare version in which the model was last saved. The last save is part of the Model File Information and is stored with the

model file when a model is saved. The model save history is automatically updated each time a model is saved.

The save history is also accessible within the Model Info dialog accessed by selecting **File ➥ Model Info...** from the main RiverWare workspace. This dialog contains the **Last Save**, **Run History** and **Comment**:

Last Save: As described above, The **Last Save** shows a single line, as information for only the last save operation. This text can be selected and copied to the system clipboard.

Run History: The **Run History** panel lists a user specified number of run history records including

- Version
- User
- Controller used
- Time the run was started
- Run duration
- Run status

You can control how many run history records to save in the **Retain information for N runs** field. This panel has a **Clear Run History** operation to clear all of the records. Because the records are saved in the model file, specifying a large number of runs to save will make the model file larger. Clearing these can recover that space.
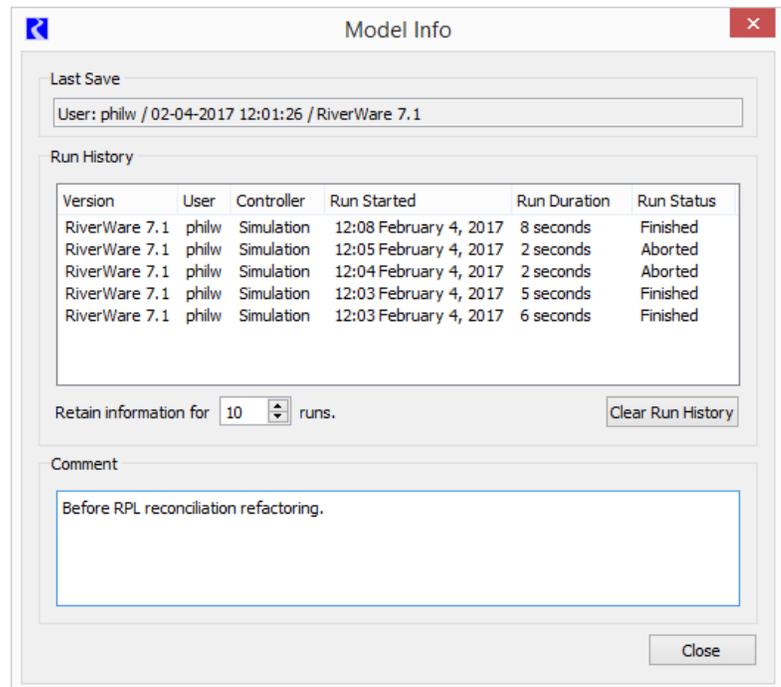
> **Note:** Since run history items are stored in the model file, items for runs performed since the model load will be retained in the model file only if the model is subsequently re-saved.

The **Run History** panel allows you to right click and **Copy selected items** to the system clipboard. When you paste, you can either paste in a standard "formatted" (i.e. HTML table) or "unformatted" tabular form. A sample is shown below of the paste results for an email program:

Comments: Enter comments about a model into the **Comment** section. Like the save history, this information is saved together with the model file. You can write comments or a brief description of a model.

> **Note:** Model Info can be shown in a Model Report using the Model Information (**HERE (Output.pdf, Section 4.2.3.9)**) and Run History (**HERE (Output.pdf, Section 4.2.3.19)**) items.

3.   Locking Models

If more than one user is working on a model, then you may want to lock the model when open to prevent others from simultaneously changing the file. To do this, use the **File ➡ Lock Model when Open** menu item and **then save the model.**   Then a lock file is created in the same directory as the model. This lock file is also created when the model is opened.

The person who first opens the model is notified that they have locked the model file, which prevents write access by other instances of RiverWare. The lock file is in the same directory path as the model file and has the name of the model plus the ".lock" extension. When the model is closed, the lock file is automatically removed.
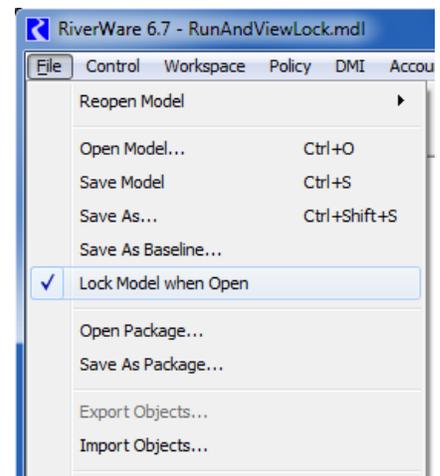
If the model is opened when a lock file already exists, the user is notified that the model file is locked by another instance of RiverWare. The time the lock file was created and the owner of the lock file is shown if that information is available from the system. Then, the model is in a state where the "Save" functionality is disabled. But, you can still make changes within the opened model and "Save As" a different name.

4.   Saving Models

There are three ways to save a model in RiverWare, the default **Save**, **Save As... and Save As Baseline...**

Save: The default **Save** command re-saves the current model with the same name and in the same directory as it was last opened or saved. The save button on the workspace menu bar implements **Save**.

Save As: The **Save As...** command invokes a file chooser to select a new name and/or directory into which to save the current workspace. Once a **Save As...** has been performed, any subsequent **Save** will use the same file name and directory until a new model is loaded, the RiverWare session ends, or a new **Save As...** overrides it.

When using **Save As..,** after you specify the file, a confirmation will allow you to specify whether to **Save Outputs in the Model File** and the **Precision** with which to save outputs. Both are described below.

Save As Baseline: The **Save As Baseline...** is used to create an unalterable baseline model for generating alternate simulation scenarios with the Scenario Manager. Creating alternate scenarios is described in greater detail in the Scenario Manager section.
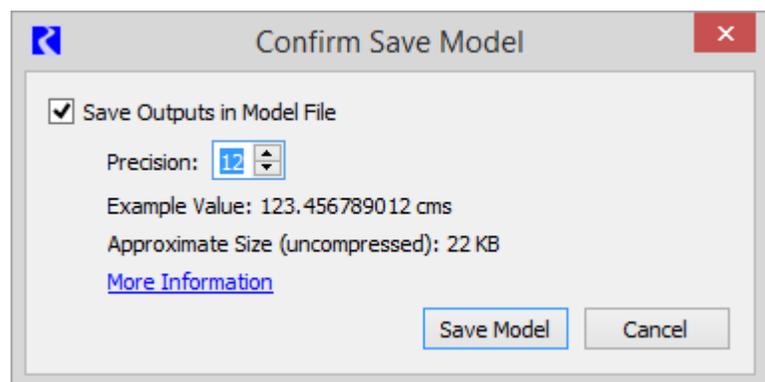
> **Note:** You can save a model for use in the RiverWare Interactive Scenario Explorer (RiverWISE) using the **Export to RiverWISE** option. This is described **HERE (RiverWISE.pdf, Section 2)**.

4.1 Saving Output Values

As models can be large and take up significant disk space, several options are available for minimizing the saved model size. One option is whether or not to save the model with its calculated output values (series values with the **Output** or **Rule** flag). The model file size is greatly reduced if the output values are not saved. This is the recommended option for saving small, frequently run models. Since only the input values are required to reproduce a run, the output may be regenerated by re-running the simulation once the model has been re-loaded. But, if your model is large or takes a long time to run, you will likely want to save results, so that you can look at output values without re-running. Only **Save As...** prompts for whether or not to save output values. The selected choice is then applied for all subsequent **Save**'s until a new model is loaded, the RiverWare session ends, or a new **Save As...** overrides it. The default **Save** behavior for a new session or model is to save output values. All series slot values with the **Input** flag (**I**, **Z** or **i** flag) and all values on non-series slots are always saved in the model file with full precision (17 digits).

4.2 Precision for Outputs

Precision refers to the number of meaningful digits in a numeric value. In the **Save As** confirmation, if you select the option to **Save Outputs in the Model File**, you will also have the option to set the
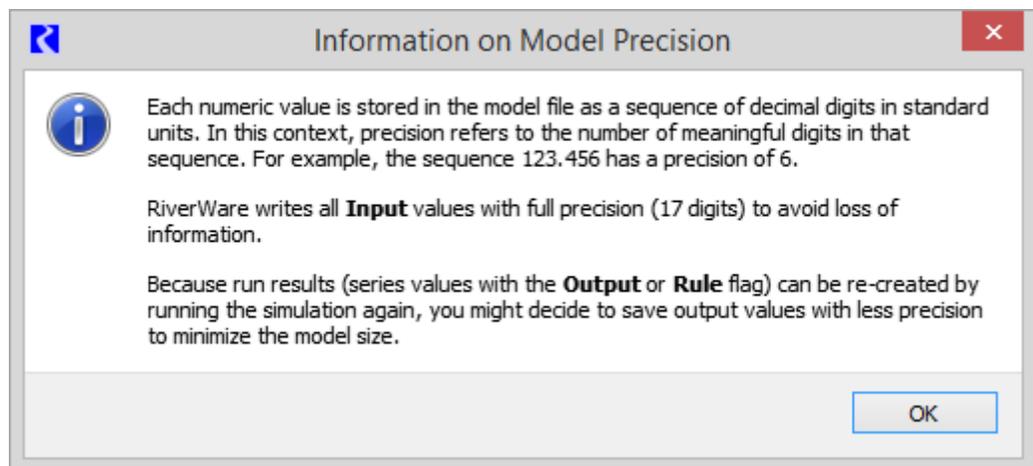
Precision with which to save output values. You can save outputs with 1 to 17 digits of precision. Saving with higher precision will increase model size. In the interface, click the **More Information** link for additional information about precision. All series slot values with the **Input** flag (**I**, **Z** or **i** flag) and all values on non-series slots are always saved in the model file with full precision (17 digits).

4.3 Naming conventions

Although there is no required extension for RiverWare model files, the ".mdl" is often used to identify a file as a model file and a ".rls" is used to identify a ruleset file.

Further, files that end in the ".mdl" suffix will display the RiverWare model file icon in windows explorer. Double clicking the icon will open the latest installed version of RiverWare and load that model.

C:/Temp/periodicTest.mdl

4.4 Compressing Model Files

Model files may be automatically compressed when they are saved by giving their file name a ".gz" extension (the gnu zip compression). When the model is re-loaded, the extension is detected, and the file is uncompressed automatically. Manually compressing and uncompressing files in WINDOWS is accomplished using any zip program such as WinZip. Manually compressed model files which do not have a ".gz" extension will not be recognized by RiverWare, and will produce an error upon loading. Thus, the ".mdl.gz" is often used to identify a file as a model file and a ".rls.gz" is used to identify a ruleset file when compression is desired.

In Windows, the association between RiverWare and the ".gz" file extension has not been made because of the general nature of the ".gz" extension,. You can manually associate ".gz" files with RiverWare so that RiverWare will automatically open when you double click a ".gz" file. To do this (in XP), right click on a .gz file and select **Properties**. Then click the **Change** button and select RiverWare from the list of files.

4.5 Backup files when manually saving

The first time a model is saved to an existing file, a backup of the original file is made. This prevents data loss if an overwrite is unintentional or the save process fails and the model is corrupted. The backup file has the same name as the original with a ".bak" extension. A single backup file is made for each model. When upgrading to a newer version of RiverWare, the backup model file name will contain the previous RiverWare version number to the model file (e.g., SampleModel.6.5.mdl.gz). This ensures that there will be a model file compatible with RiverWare version 6.5 until the user explicitly removes it.

4.6 Automatic Backup Save

The automatic backup save utility saves a backup file in a temporary directory at a defined interval. It is strictly a backup save of the model file, it never overwrite the model that the user opened.

The feature is configured from the workspace **File ➥ Automatic Backup Save Configuration…** menu item, which opens the **Automatic Backup Save Configuration** dialog:

From the dialog, you configure:

- Whether the model file should be Automatic Backup Saved. If unchecked, the remaining fields are irrelevant.
- Whether the Automatic Backup Saved backup file should contain run results (output values).
- The Automatic Backup Save interval. (More on this **HERE (Section 4.6.1).**)
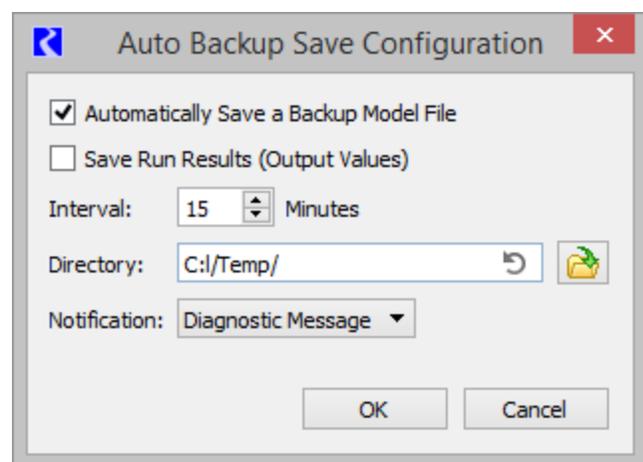
- The Automatic Backup Save directory, by default the user's temporary directory. (More on this **HERE (Section 4.6.2).**)
- The Automatic Backup Save notification:

  1. **None**: RiverWare beeps at the beginning of the Automatic Backup Save and provides no further notification.

  2. **Diagnostic Message**: RiverWare beeps at the beginning of the Automatic Backup Save and displays a diagnostic message upon its successful completion. A single message is posted in green if successful.

  3. **Status Dialog**: RiverWare opens a notification window at the beginning of the Automatic Backup Save and closes the window upon its successful completion. If the **Automatic Backup Save** is unsuccessful the notification window remains open and displays the error.

In all cases, if the **Automatic Backup Save** is unsuccessful an error message is displayed and Automatic Backup Save is disabled (with a red highlight message in the diagnostics):

```
Auto-saving the model file failed: Cannot open "C:\Bad\Directory\Model.mdl" for
writing: No such file or directory

Automatic Backup Save has been disabled for model file(s)
```

If you want to see additional messages on the automatic backup save process, turn on **Workspace ➥ Model Load/Save** diagnostic category.

4.6.1 Automatic Backup Save Interval

RiverWare performs activities during which an **Automatic Backup Save** isn't appropriate, for example during a simulation. If an **Automatic Backup Save** comes due during such an activity the **Automatic Backup Save** feature enters "retry mode":

- The **Automatic Backup Save** interval is set to one minute.
- Within a minute of the activity's completion the model file is Automatic Backup Saved.
- The **Automatic Backup Save** interval is restored to its configured value.

4.6.2 Automatic Backup Save Directory

By default the Automatic Backup Save directory is the user's temporary directory. On recent versions of Windows this is:

C:\Users\<user name>\AppData\Local\Temp

It's unlikely a user will have write permission for another user's temporary directory. To facilitate sharing models, if a user loads a model file with Automatic Backup Save enabled and doesn't have write permission for the Automatic Backup Save directory, the Automatic Backup Save directory is changed to the user's temporary directory (with a green highlight diagnostic message):

```
Automatic Backup Save directory has been changed to: C:/Users/Voltaire/AppData/Local/
Temp/
```

Finally, because the **Automatic Backup Save** directory might not be obvious to all users, the configuration dialog has a reset button which resets the Automatic Backup Save directory to the user's temporary directory as shown in the screenshot
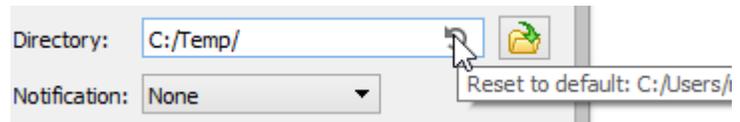
4.6.3 Open Automatic Backup Save Model Menu Item

If the model file has been Automatic Backup Saved the backup file can be opened by selecting the **File ➡ Open Automatic Backup Save Model** menu item. (If the model file hasn't been Automatic Backup Saved, the menu item is hidden.) This is equivalent to selecting the **File ➡ Open Model…** menu item, navigating to the Automatic Backup Save directory and selecting the backup file.

4.6.4 Backup File Name

If the model file name ends with either .mdl or .mdl.gz, the suffix is preserved and the backup file name is <file name>.~1~.mdl or <file name>.~1~.mdl.gz. Otherwise the backup file name is <file name>.~1~. If there isn't a model file loaded from a named file, the backup file name is Workspace.~1~.

5.   Comparing Models - The Model Comparison Tool

When developing a model, it is often desirable to compare a model to some previous version of the model to identify differences. In addition, comparison tools are essential when multiple people are developing a model or there are multiple versions of a model that are to be merged together. The need for a model comparison tool can be illustrated by the following use case:

1.   Bob acquires a copy of the current "official" version of the model file, modifies it, and saves a new model file. The modification process could be lengthy and involve many different sorts of changes, e.g., adding objects and slots, running DMIs, executing runs, interactively changing data, etc.

2.   Bob provides the new model file to Susie, who would like to understand the ways in which the new model differs from the original one. A comparison tool helps Susie locate and understand the differences.

3.   Having reviewed Bob's changes, Susie might then incorporate some or all of these modifications into a new model and then save this new model as the new official version. Note that if only some of Bob's changes are to be incorporated into the official model, or if the official model has changed since Bob originally obtained a copy of it, then Susie cannot simply replace the current official model with the model provided by Bob. Instead Susie should compare Bob's model with the original or updated "official" model to see what has changed and then make those same changes to the original model.

The **Model Comparison Tool** compares the workspace entities (currently objects and slots) of two models. It looks at the properties of the entities and their values. It then shows the differences and allows you to see which items have changed.

Following are the limitations of the **Model Comparison Tool:**

- A relatively small subset of model entities are currently compared by this tool. The compared items are shown in the model selection dialog.
- This tool doesn't support automatic merging. It does support merging by showing differences and by allowing you to copy from the selected property values panel.

5.1 Accessing the Model Comparison Tool

From the RiverWare workspace, use the menu: **Utilities ➡ Model Comparison Tool...**

5.2 Selecting Models to Compare

The tool opens and shows a dialog where you specify the two models to compare.
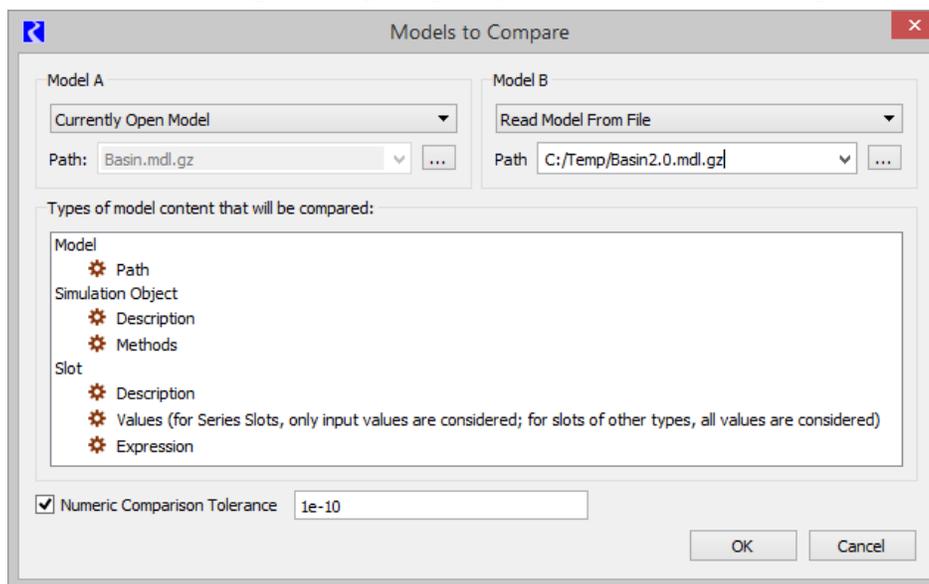
The two models are referred to as **Model A** and **Model B**. For both models, choose from either:

- **Currently Open Model**: Choose the currently opened model, including any changes made since last saving the model.
- **Read Model from File**: Choose a model on the file system. Use the ellipsis button to open a file chooser to select the model.

The lower portion of the dialog shows which entities and properties in the model file will be compared. Note, this is a small subset of entities/properties in the model file. For example the path of the two models is compared, but the run timestep is currently not compared. The list of content will be expanded in future releases. The following is a description of each entity compared, and for each of the

entities, the properties that are considered. Note that not all entities of a given type will have all of the properties, e.g., not all slots have expressions.

- Model
  - Path - The location of the model on the file system are compared as text.
- Simulation Object
  - Description - The text description as shown on the Open Object dialog's Description tab are compared as text.
  - Methods - The user-selectable methods, shown on the object's **Methods** tab are compared as text. Note, the tool compares all categories and methods, whether they are visible or not (perhaps because they are specific to a certain controller or dependent on other methods/categories). They are listed alphabetically by category.
- Slot
  - Description - For each slot, the description is compared as text.
  - Values - The values of slots are compared using the tolerance described below. The slot values are shown in the appropriate format based on the type of slot. For tables, periodic and scalar slots, the values are compared using the full precision in standard values on the slots. For series slots, only input values (I flagged) are compared. Note, certain slot types or values may not display exactly as shown in RiverWare dialogs. For example, values using the DateTime unit type are compared but the values shown are the numeric encoding. This will show you a difference and you can look at the slot for the actual datetime value.
  - Expressions - For expression slots, the RPL expression is compared.

If desired, specify an absolute tolerance for numeric comparisons in the **Numeric Comparison Tolerance** field. The value 1e-10 is used by default to avoid flagging as different those values with very small differences that occur due to machine precision.

When finished, click **OK**. The tool then performs the comparison and the main **Model Comparison Tool** is then shown.

> **Note:** The comparison happens when the selection dialog is confirmed or the refresh button is clicked. Afterwards, the tool displays the results of that comparison, which might no longer apply if you have since changed the workspace or deleted the file.

5.3 Tour of the Model Comparison Tool

The **Model Comparison Tool** contains four main areas as shown in the screenshot:

- The **Models** panel shows the names, sources and paths of the selected models.
- The **Comparison Results** area shows a hierarchical view of the comparison results. This shows where there are differences between the two models.
- The **Selected Property** panel shows the property (if any) for the currently selected item in the comparison results view. This shows the differences are highlighted in orange.
- When a series slot's values are selected, a plot is also shown. Data from Model A is shown with a blue line, while model B data is shown in red. A legend above the data reinforces the color usage.

Sets: Names and locations of the two models

Number of differences

Comparison Results:

Hierarchical view of Results

Selected Property of Set A and Set B

Highlighted Differences

Plot of Series Slot Value Differences
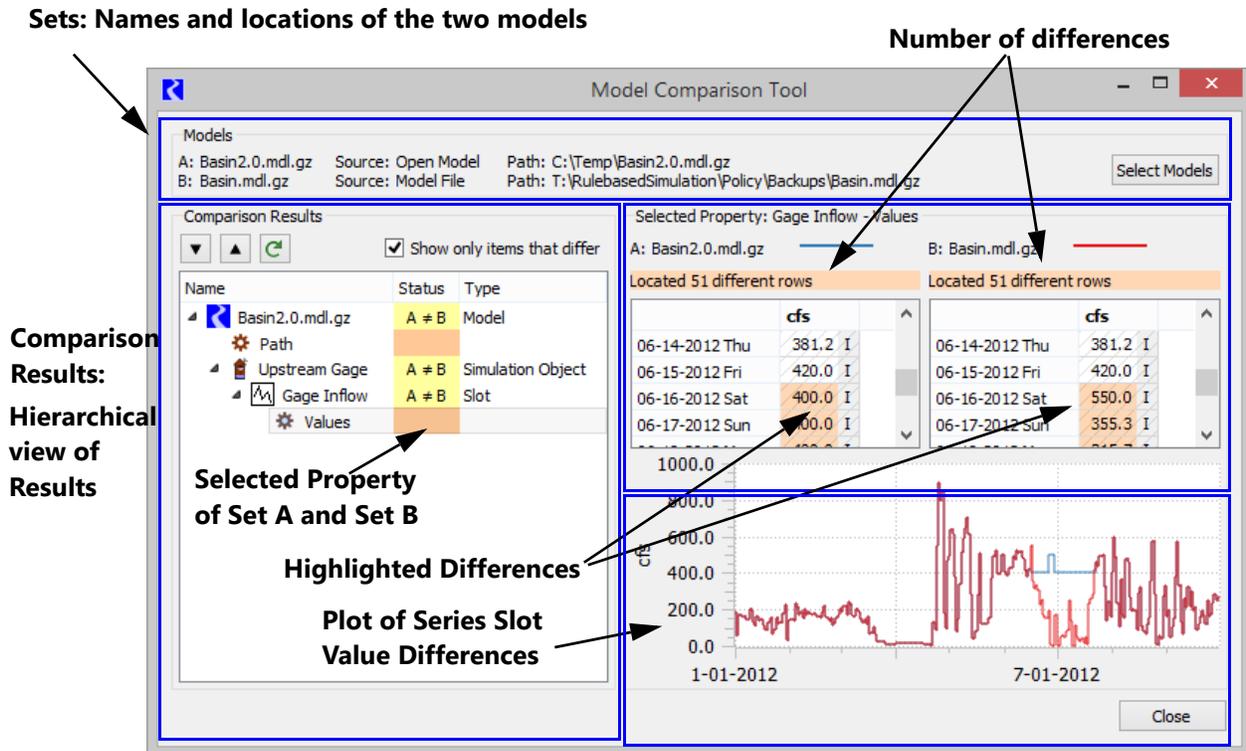
The following sections describe these areas and how to use the tool.

Models panel: The models panel shows information about the models that were compared, including, for each model, the name (usually the file name), the source (i.e., whether it was the currently opened model or read from a file), and the full file system path. Use the **Select Models** button to choose different models and initiate another comparison. This opens the dialog shown **HERE (Section 5.2)**.

Comparison Results: The **Comparison Results** panel shows a hierarchical view of the results of the comparison. The hierarchy is based on the structures of the two model. Objects contain slots as children entities. Aggregate objects have object elements as their children. Objects and slots are listed in alphabetical order within the hierarchy level. Currently, there is no account information.

Double-clicking on an object or slot item will open the appropriate dialog to view the version of that entity, if one of the models compared was the open model.

Columns include the **Name**, **Status** and **Type**. The Model, Objects, and Slots show their particular icon. The **Status** column shows the results of the comparison. The following table gives the possible status values and their meanings:

Status	Meaning
A = B	The item is identical in Models A and B.
A ≠ B	The item is different in Models A and B.

| Status	Meaning
A	The item exists in model A but not in Model B.
B	The item exists in Model B but not in Model A.

The **Type** column lists the type of that row: either a Model, Simulation Object, Slot, or Property. The comparison tool ultimately compares **Properties** of the items, not the items themselves. In the **Status** column, a lighter yellow shading indicates that one or more of the item's properties differs between the models. The darker orange color indicates that the value of the property is different.

The Comparison Results panel has buttons on the top for navigation and control of what is shown:

Button	Text	Meaning
▼	Next Difference	Select the next property that has a difference
▲	Previous Difference	Select the previous property that has a difference
⟳	Recompute Differences	Recompute the differences between the two models. This is useful if you make changes to Model A or B
☑ Show only items that differ	Show only items that differ	Show only items that are different in the results panel. This is the default behavior.

> **Note:** When the models are selected, the **Model Comparison Tool** compares the specified models as described **HERE (Section 5.4)**. If you make changes to one of the chosen models, it is not immediately reflected in the comparison tool. Use the **Recompute Differences** button ⟳ to update the status in the tool.

Selected Property: The **Selected Property** panel shows the values for the selected property where applicable. If a yellow row (Model, Object, Slot, etc...) is selected, a note is shown that "No Property Selected". Only property values are shown in these views. When a **Property** row is selected, the panel shows the values for that property. Depending on which property is selected, its value will be displayed as text, a table of slot value, possibly including a plot, or as a RPL expression. For example, in the screenshot below, there are differences in the Methods on Muddy reservoir. Clicking on the Methods row shows the differences in the Selected Properties area. (This is a character by character comparison of the text, hence there are two differences identified.)
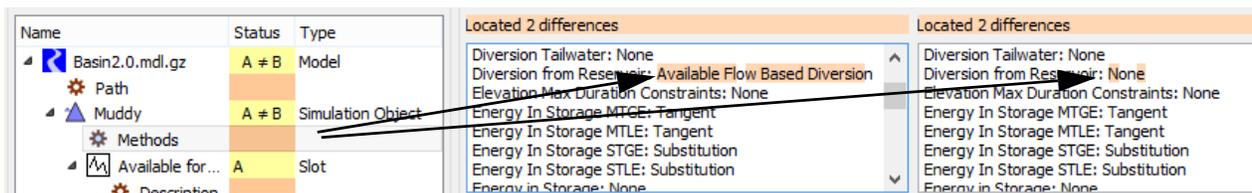
Right click on the property values and choose **Copy** to copy the data to the system buffer. This works for both text differences and slot values. You can paste these to an email, document, or spreadsheet.

Plot: When **Values** for a series slot are selected, a plot is shown within the **Selected Properties** panel. The plot shows the entire set of data with two lines, one for each model. Model A data is shown in blue, while Model B data is shown in red. A legend is shown above the data.
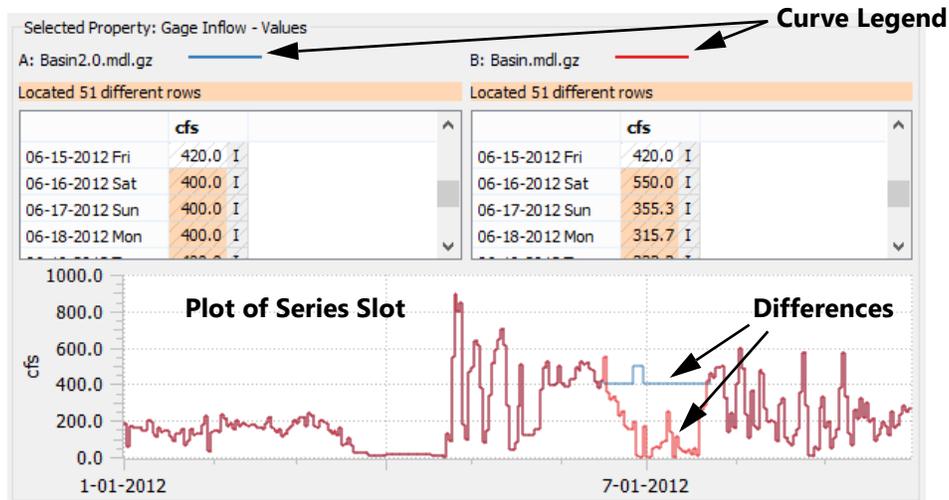
Right click on the plot area for options to Auto-Scale, Zoom, Move, Copy Plot Image, Export Image, or Print. Draw a rectangle on the plot to zoom in.

5.4 How the Model Comparison Tool Works

In RiverWare, only one model can be loaded at a time. To be able to compare two models, RiverWare must create in-memory representations of the two sets of data and values to be compared. If Model A is the opened model, then RiverWare needs to read Model B and then create representations of values that can be compared. To do so, RiverWare synchronously executes a batch RiverWare process which:

- Loads the model file
- Creates a representation of the model
- Writes that representation to a file
- The original RiverWare instance then reads the file into memory
- It repeats the same steps for the other model, if necessary
- With both sets of information in memory, RiverWare can then compare the properties and values and present the results in the tool.

Note: Because of this process, comparing large models can take a long time to load the model, write a representation of the data, and then read that data back into RiverWare.

6.   RiverWare Packages

A model and associated files can be grouped together and saved as a RiverWare "package". These always contain a model file and can also contain ancillary files including:

- Rulesets
- Global Functions Sets
- Optimization Goal Sets
- System Control Tables (SCTs)
- Workspace Images,

A package is used to group the files together into one file and compress them. This functionality is often used to share a set of files with other users. For example, you can package all the files used with a model and send them to a co-worker. The co-worker opens the package to create all the files in the destination directory and then RiverWare automatically opens them.

Packages are accessed through the workspace's File menu. The **Save As Package...** and **Open Package...** menus on the workspace File menu allow you to work with packages:

6.1 Saving a Package

Use the **Save As Package...** menu to save a set of files together as a package. This opens a dialog where you specify the package file name. Use the [icon] button to use a file chooser or type in a
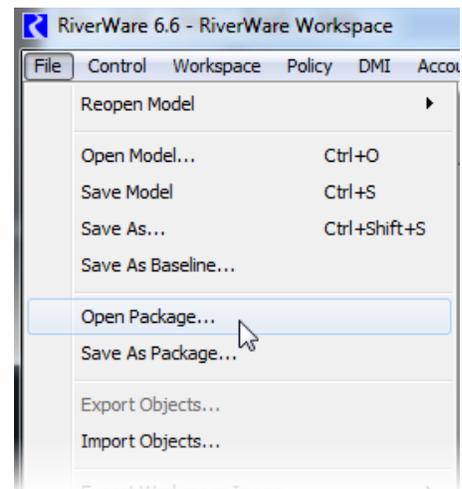
file path and name. Also, choose the components to include. By default, all available files will be checked.
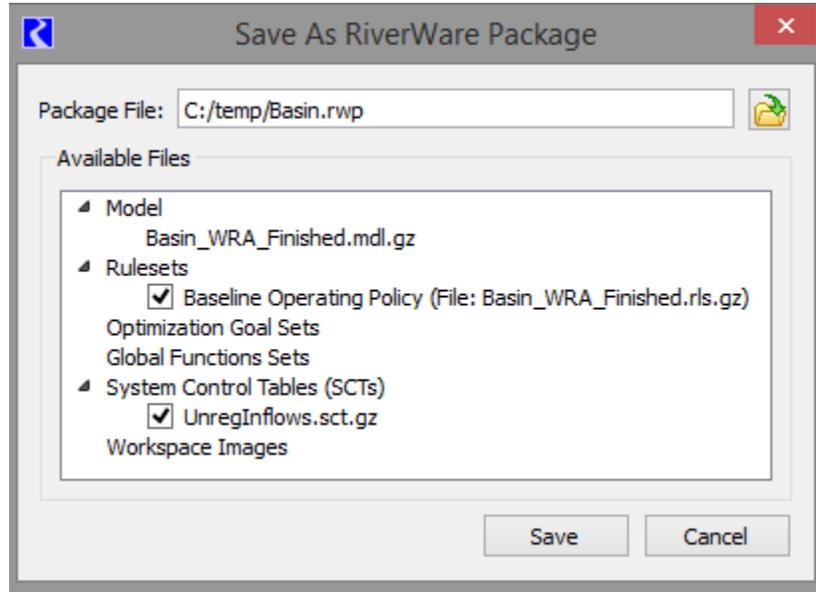
The model is not checkable and is always included in the package. This includes the model file and any RPL sets that have been configured to be saved into the model file itself. Click **HERE (RPLUserInterface.pdf, Section 1.3)** for more information on embedding sets in a model.

RPL sets (Rulesets, Goalsets, and Global Functions Sets) that are open and that have been saved as files will be listed as individual options for inclusion in the package. If a set has been created, but not saved to a file, you will get a warning message that the set must be saved to be available in the list.

SCTs that are open and that have been saved as files will be listed for inclusion in the package under the SCT category. You can individually select whether or not to include each SCT from the list. If an SCT has been created, but not saved to a file, you will get a warning message that the SCT must be saved to be available in the list.

If simulation, accounting, or geospatial images are used in the model, they are available as options for inclusion in the model package.

In creating the package, all selected files will be saved to a temporary directory for packaging as shown in the diagram to the right. When packaging is completed, these files and the temporary directory will be deleted, leaving just the specified package file.

Although not mandatory, we recommend using the suffix ".rwp" to represent RiverWare packages.

If you have made changes to any of these items since they were last saved to files, the copies made for the package will include these changes, but the original files will not. You must save them in normal RiverWare operation to save the changes to the individual files. But, creating a package saves everything as it currently appears in the session without affecting any of the individual files on disk.
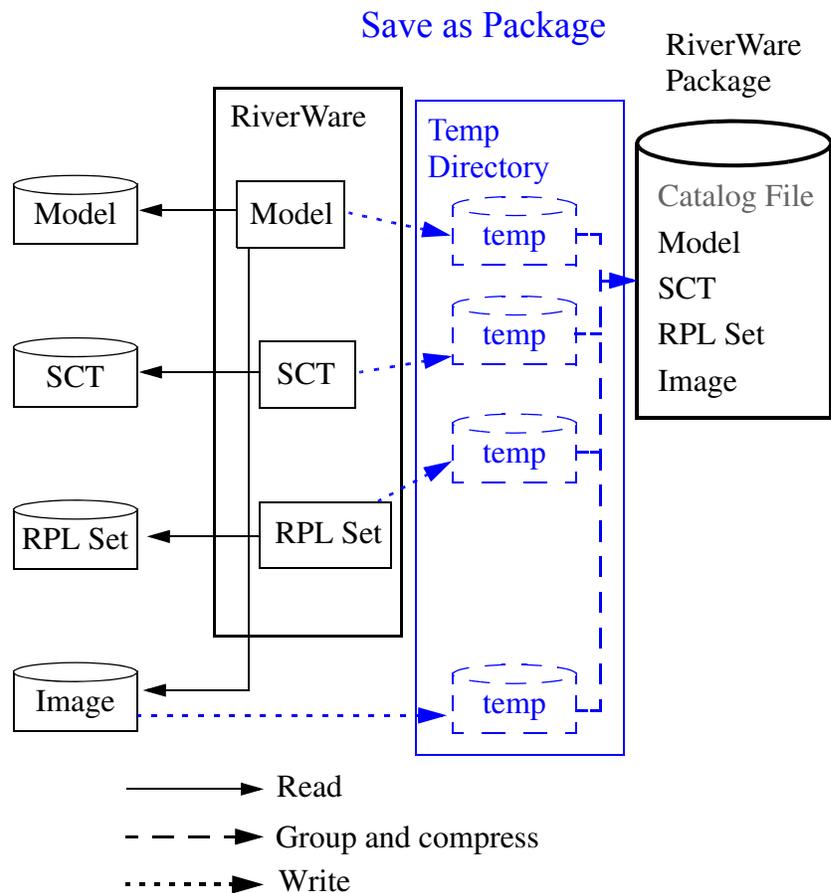

6.2 Opening a Package

Use the **Open Package** option in the workspace **File** menu to unpacks all the files and then open those files into the RiverWare session. This menu opens a dialog in which you specify the package file to open and a destination directory to which the files will be unpacked.

The package file can be chosen via a file chooser [icon] or the name can be typed into the **Package File** line (environment variables are supported). Similarly, the directory where the files will be opened to can also be specified by selection or by typing.

Once the package file is chosen, a default "**Destination Directory**" is shown, which consists of a subdirectory using the name of the package. Change the directory as desired.
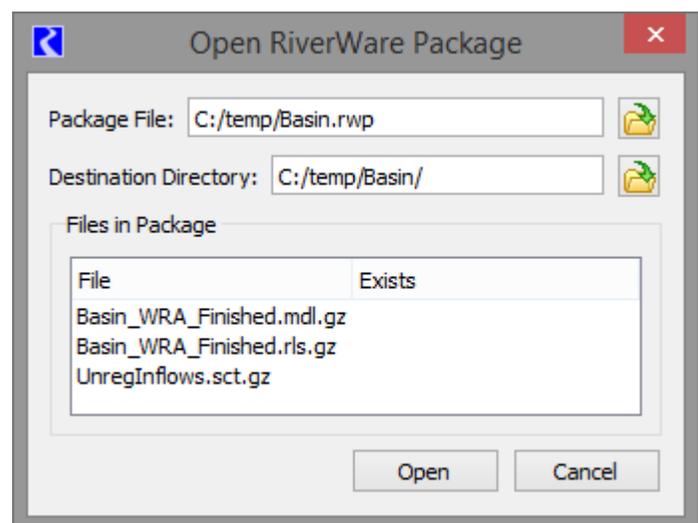
The **Files in Package** list displays the names of the files that are in the package. If files corresponding to these names already exist in the specified **Destination Directory**, this will be indicated with a "yes" in the **Exists** column of the list. If any of the files currently exist, the message at the bottom appears warning that the existing files will be overwritten when opening the package.

When the package is opened, files in the package will be extracted to the specified directory. The model file will be loaded, and RPL sets and SCT files will be opened. If a RPL set was loaded when the package was saved, it will be reloaded when the package is opened. The workspace images loaded with the model will now have the extracted image files as their source. Extracted files are persistent in their directory even after the RiverWare session is closed, so are just like any other set of RiverWare files. The files can be repackaged at any time into the same, or a different package file using the "Save as Package" functionality described above.