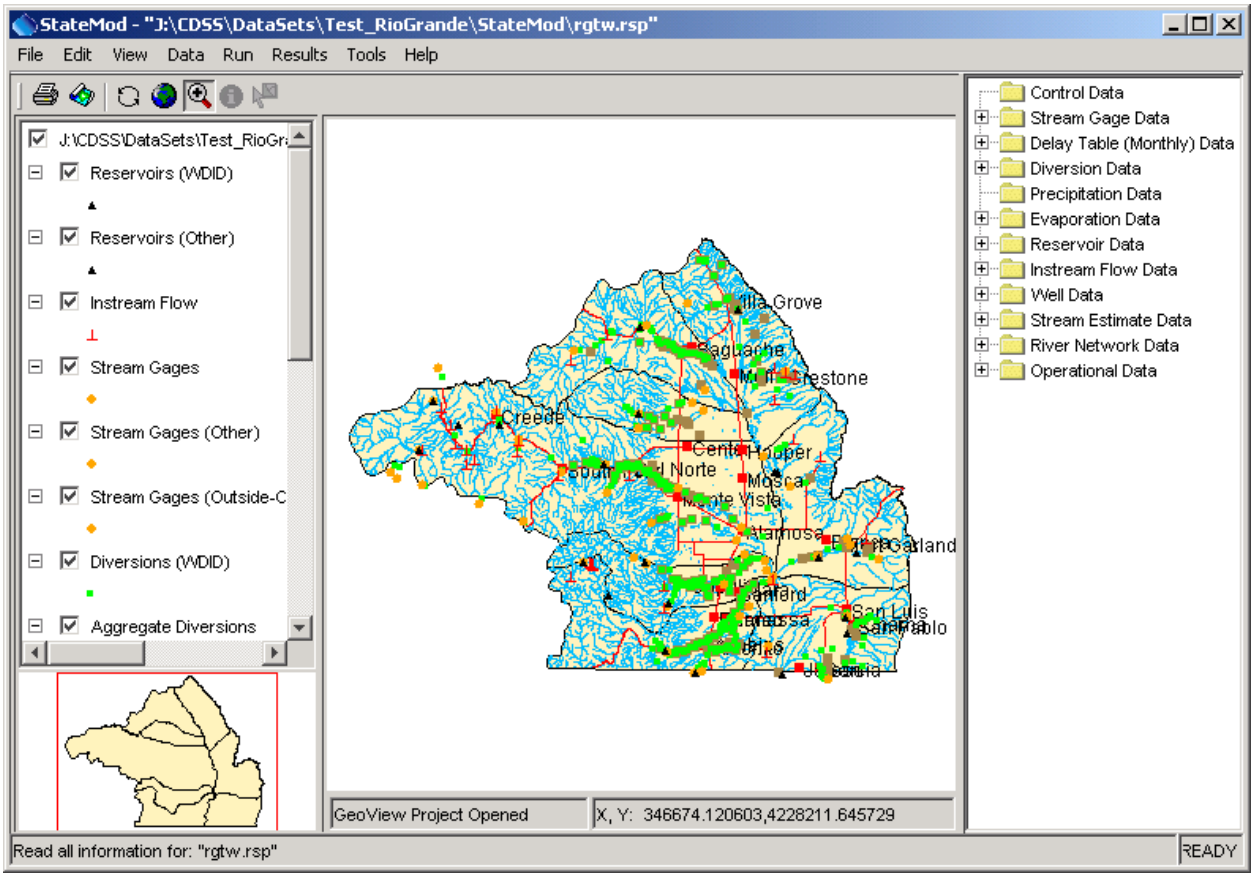
# StateMod Graphical User Interface (StateMod GUI)




**Colorado Department of Natural Resources**
**Colorado Water Conservation Board**
**Division of Water Resources**

Developed by:

***Riverside Technology, inc.***

Version 07.04.00, 2013-04-17

This page is intentionally blank.

This document is formatted for two-sided printing.

# Table of Contents

# Table of Contents

# Table of Contents

# Table of Contents

# Table of Contents

This page is intentionally blank.

# DISCLAIMER for CDSS Products

2002-02-16, Acrobat Distiller

CDSS products include data and software from State of Colorado sources and from external sources like the U. S. Geological Survey (USGS). The following disclaimer applies to CDSS products:

**CDSS products and associated access are under development at this time. Access is provided solely to test and demonstrate CDSS capabilities. In the future, this access may be restricted or offered for a fee. The State assumes no legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed herein. It is the user's responsibility to determine the fitness of the data for a particular purpose.**

This page is intentionally blank.

# Table of Contents

Version 07.00.00, 2007-03-02, Color, Acrobat Distiller

# 1 Acknowledgements

Version 06.03.02, 2006-03-04, Color, Acrobat Distiller

The StateMod GUI has been developed by Riverside Technology, inc. (RTi) with funding from the State of Colorado, Colorado Water Conservation Board as part of Colorado's Decision Support Systems (CDSS). The StateMod GUI continues to be developed as part of the South Platte Decision Support System (SPDSS), Statewide Water Supply Initiative (SWSI), and other projects.

Support for the StateMod GUI can be contacted by emailing cdss@state.co.us.

This page is intentionally blank.

# 2 Introduction

Version 07.04.00, 2013-04-17

The StateMod GUI is a graphical user interface for the StateMod software, which is the water allocation modeling software used within Colorado's Decision Support Systems (CDSS). The StateMod software is a command-line program that does not itself include a graphical user interface. The following resources provide background information about StateMod data files and modeling approach:

- StateMod documentation – provides details about how to use the StateMod software to implement a basin model.
- StateDMI documentation – provides details about how to create StateMod input files using automated procedures.
- TSTool documentation – provides information about general capabilities to create StateMod time series files (the StateDMI documentation focuses on StateMod data sets and related procedures, whereas the TSTool documentation is more general).

Although the StateMod software has been applied mainly within CDSS, the software also is suitable to model basins outside of Colorado and the CDSS effort.

The StateMod GUI simplifies modeling tasks, including:

- Visualizing data
- Modifying data files for "what if" questions (see discussion below related to using StateDMI and TSTool to edit an existing data set)
- Running StateMod
- Visualizing results
- Using advanced tools for data analysis

The StateMod GUI operates on StateMod data sets described by a response file, which lists all the files that are part of a data set (see the StateMod model documentation for more information about response files). It is assumed that StateMod GUI users are also familiar with the StateMod software, including data files and model functionality. If an existing data set is used without modifications (e.g., data and results are viewed but the model is not run), then a relatively limited understanding of StateMod is necessary to use the StateMod GUI.

Because of the large amount of data stored in StateMod files, it generally is not practical to create a large data set from scratch using the GUI. Within CDSS, data management utilities (DMIs) are used to create baseline data sets, which can then be run and modified using the StateMod GUI. For example, the StateDMI software can be used to create the model network, station files, water rights, and other supporting files, using the HydroBase database for data. The TSTool software can be used to create the StateMod time series files. The comments at the top of StateMod input files indicate the program(s) and commands that have been used to create the files. Using DMI programs in CDSS automates processing of large amounts of data, eliminating the need to enter data into a data set using the StateMod GUI or text editors. For systems other than CDSS, a similar approach can be taken, although HydroBase as a data source may need to be replaced with an alternative.

It is possible to use StateDMI and TSTool to automate creation of "what if" StateMod datasets, which results in a clear trail of modifications to data sets and is consistent with the data-centered approach used by the State of Colorado:

1. Start with the master data set, for example from the State of Colorado's website.
2. Copy the data set to a scenario folder, naming the top-level folder as appropriate. Keep all of the individual StateMod names the same so that changes to files and processing can be minimized.
3. Create StateDMI and TSTool command files to automate modification of the dataset. Note that this approach clearly identifies changes to a dataset, whereas recreating the dataset using modified versions of the original command files may hide the changes that are being made. For example, to automate editing a data set to add a new reservoir:
    a. Use a StateDMI command file to read the original reservoir stations file (for example from the data set in the master copy of the data set), and set data for an additional reservoir. StateDMI "Write" commands have a parameter to "Update" or "Overwrite" a file, and the latter adds additional information to the comments at the top of the file to indicate edits.
    b. Similarly, use a StateDMI command file to automate changes to the reservoir rights file.
    c. Use TSTool to read the reservoir target time series, define a new time series, and write the updated file.
    d. The network **.net* file must be updated interactively using the StateDMI network editor. In the future, StateDMI commands may be implemented to automate insertion of a new network node, minimizing or eliminating the need to use the network editor for this task.
4. Open the dataset in StateMod GUI to visualize data and run the model.
5. The model can also be run on the command line and then use tools such as TSTool to view time series.
6. Tools like TSTool can be used to read data and results from multiple datasets for comparison.

The above automated editing approach ensures that a scenario data set is constructed consistent with the CDSS data-centered approach, and it allows the scenario data set to be recreated if the master data set should change (e.g., the State of Colorado extends the data set with more historical data). The above approach also overcomes limitations in the GUI's editing capabilities (see limitations in **Release Notes**).

The StateMod GUI does not check that all user-supplied data are reasonable, although many checks are in place and additional checks are added over time (early StateMod GUI versions relied on the StateMod data-check run mode). After editing data, StateMod should be run in data-check mode to find possible user errors (see **Chapter 6 – Running StateMod**). It is the user's responsibility to read messages generated by StateMod and make appropriate changes to the data set (or acknowledge that the data are acceptable). StateMod creates a log file in the data set directory with a name matching the data set and the extension *.log*. This log file should be reviewed to resolve problems. The StateMod GUI does attempt to minimize data errors by providing choices of appropriate values for data items and for maintaining the relationships between data files.

When a data set is selected, the StateMod GUI reads most of the data for the data set into computer memory. When data are edited, the changes are made to the copy of the data within memory (instead of immediately writing to the data files for each change, which would be slow). The StateMod GUI will automatically detect when changes to data occur and will notify the user before making a model run or closing the StateMod GUI. The user can then choose to save the StateMod input files so that the StateMod software will use the changes. The StateMod GUI will only write the changes to the appropriate files when ***File…Save…*** is selected. If the changes are not saved to files, then the next run of the StateMod model will use the previously saved version of the data files. This type of edit tracking is

necessary because the StateMod GUI and StateMod model are separate programs. **The GUI currently does not flag edited data in the StateMod data files – this is a requirement that is being evaluated**.

StateMod data sets are divided into groups of data set components. Each group includes a primary component and secondary components. For example, for diversion data, the primary component is diversion stations; secondary components include diversion water rights and time series associated with the diversion stations. The diversion station identifier is the primary link between data components. Important data component groups have display windows that list all the data for the group. Where appropriate, additional windows are provided and can be accessed by clicking on the appropriate button within the main data window. For example, a window is shown for diversion data; however, to view water rights for a diversion station, a button is pressed to display a secondary window.

The StateMod data shown in this document are not discussed in detail. For more information about the model, the data necessary to run the model, and modeling guidelines, see the StateMod software documentation. The StateDMI and TSTool documentation includes information about creating data files.

A typical StateMod GUI session consists of the following steps:

1. Run the StateMod GUI. A shortcut may have been configured on the desktop, or for CDSS, the following menu may have been configured: ***Start…All Programs…CDSS…StateModGUI-Version***. Immediately after starting, the StateMod GUI will run `statemod -version` to determine the StateMod model version. This confirms that StateMod can be found and identifies the StateMod version for subsequent actions.
2. Select a data set by selecting a StateMod response file using the ***File…Open*** menu (see **Section 3.4**).
3. View and edit data using the ***Data*** menu (see **Chapter 4 – The Model Network** and **Chapter 5 – Viewing and Editing Data**).
4. Run the model using the ***Run*** menu (see **Chapter 6 – Running StateMod**).
5. View output using the ***Results*** menu (see **Chapter 7 – Viewing Model Results**).
6. Perform additional analysis using the ***Tools*** menu (see **Chapter 9 – Tools**).
7. Exit the StateMod GUI using ***File…Exit***.
8. Optionally use software like TSTool to perform additional data and results viewing and analysis (see the TSTool documentation for more information).

This documentation is organized consistent with the StateMod GUI interface, with chapters for every major menu. There are also appendices for configuration information and documentation that is shared with other software documents. The following chapters are available in this documentation:

**Chapter 1 Acknowledgements** – recognizes contributors to the development and maintenance of the StateMod GUI software.

**Chapter 2 Introduction** (this chapter) – provides background information about the StateMod GUI and its interaction with StateMod.

**Chapter 3 Getting Started** – describes the main interface, including the map features. Refer to this section for questions about how to interact with the map in the StateMod GUI. Refer to the GeoView Mapping Tools appendix for general information about the map interface.

**Chapter 4 The Model Network** – discusses how to create new data sets and view or modify an existing model network.

**Chapter 5 Viewing and Editing Data** – describes the main data windows, which are used to view and edit StateMod data.

**Chapter 6 Running StateMod** – describes how to run StateMod software from the StateMod GUI.

**Chapter 7 Viewing Model Results** – describes how to view StateMod output as graphs and text.

**Chapter 8 Using the Map** – describes how to use the map interface. This chapter is currently a placeholder and material from other chapters will be moved here in the future.

**Chapter 9 Tools** – describes how to use advanced analysis tools, including adding summary information to the map.

**Chapter 10 Troubleshooting** – discusses common problems and how to resolve them.

**Appendix – StateMod GUI Installation and Configuration** – describes how to install and configure the StateMod GUI.

**Appendix – Release Notes** – summarizes software update information.

**Appendix – TSView Time Series Viewing Tools** – discusses the general time series viewing tools used in the StateMod GUI and other CDSS software.

**Appendix – GeoView Mapping Tools** – provides an overview of the mapping interface, which is also used in other software in Colorado's Decision Support Systems.

# 3 Getting Started

Version 07.04.00, 2013-04-18

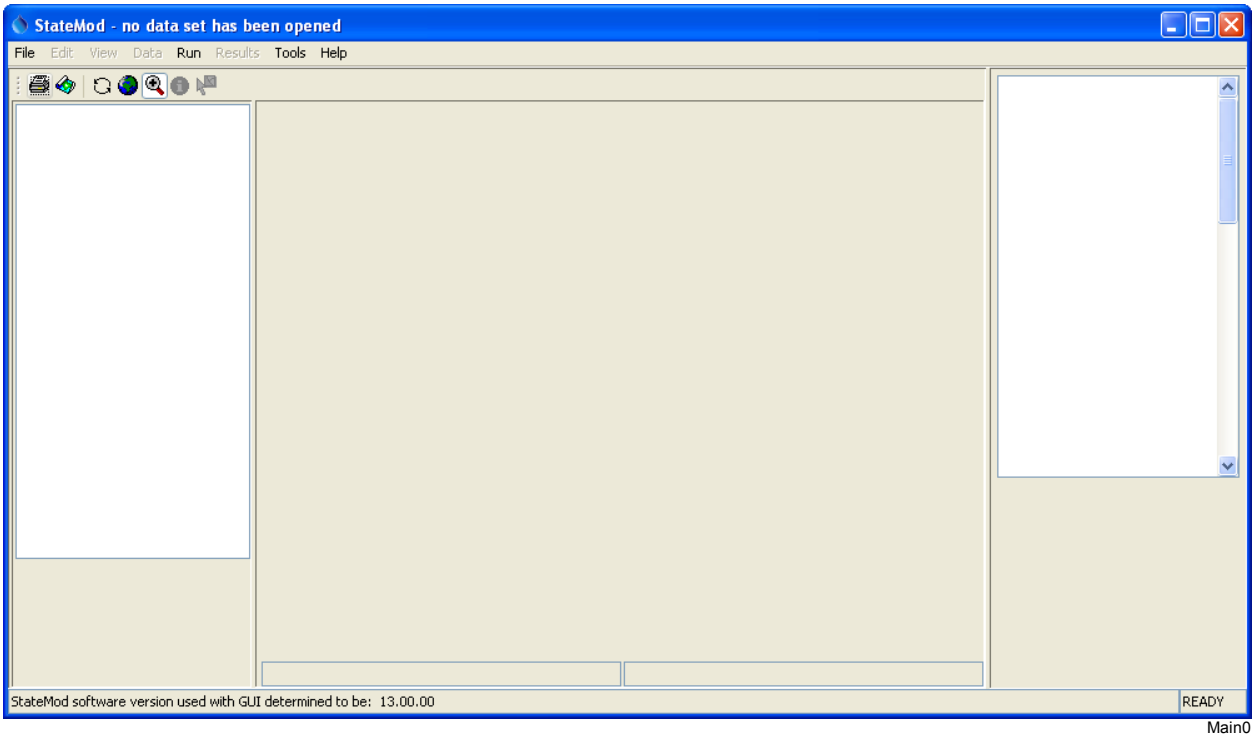
This chapter provides an overview of the StateMod GUI. Additional chapters provide more detail about specific features.

## 3.1 Starting the StateMod GUI

The StateMod GUI is a Java application and therefore is run using the Java Runtime Environment (JRE). The StateMod GUI can be started from the ***CDSS…StateModGUI-Version*** start menu or a shortcut on the desktop (if configured during installation), where ***Version*** is similar to 07.04.00.

## 3.2 Main Interface

The StateMod GUI, when initially opened, appears as shown in the following figure:

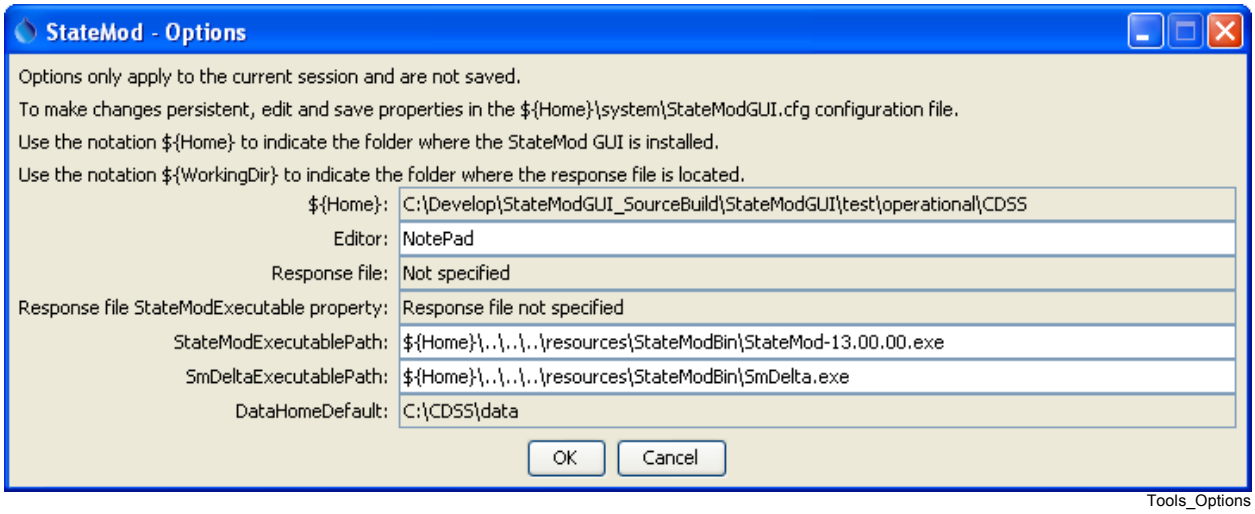



Main0

**StateMod GUI Main Interface after Startup**

The GUI depends on the StateMod software to generate output and it must know the StateMod version to properly interpret some data. Consequently, at startup the GUI runs the StateMod `-version` option. The StateMod version is displayed in the bottom of the main window and also can be displayed using ***Help…About StateMod/StateMod GUI***.

The StateMod executable that is used by the GUI is configured using the ***Tools…Options*** menu as shown in the following figure, and there is flexibility to specify a StateMod executable that is stored with each data set (this is safest to ensure that the data set is published with the StateMod software that was used to run it). The StateMod version is important because some file formats have changed over time and there is not information in the file itself to indicate the version.



Tools_Options

**StateMod GUI Configuration Options**

The first step to using the StateMod GUI is to open a StateMod data set by selecting a response file for the data set. **Section 3.3 – File Menu** provides more information about this step. The above figure illustrates how the `DataHomeDefault` configuration property can be used to control the initial folder shown by the GUI when opening a data set.

Many of the examples in this documentation use a version of the State of Colorado's Colorado River data set, which is saved in the *doc\Training\data_cm2009* folder under the StateMod GUI installation. This folder can be copied to a working folder in order to ensure that the original version distributed with the software remains unchanged (as an archive in case the original needs to be retrieved).

Warning: Not all StateMod data sets have been defined to integrate completely with the StateMod GUI. Common issues are:

- Response files do not indicate locations of GIS files or network. The following response file excerpt illustrates needed information:

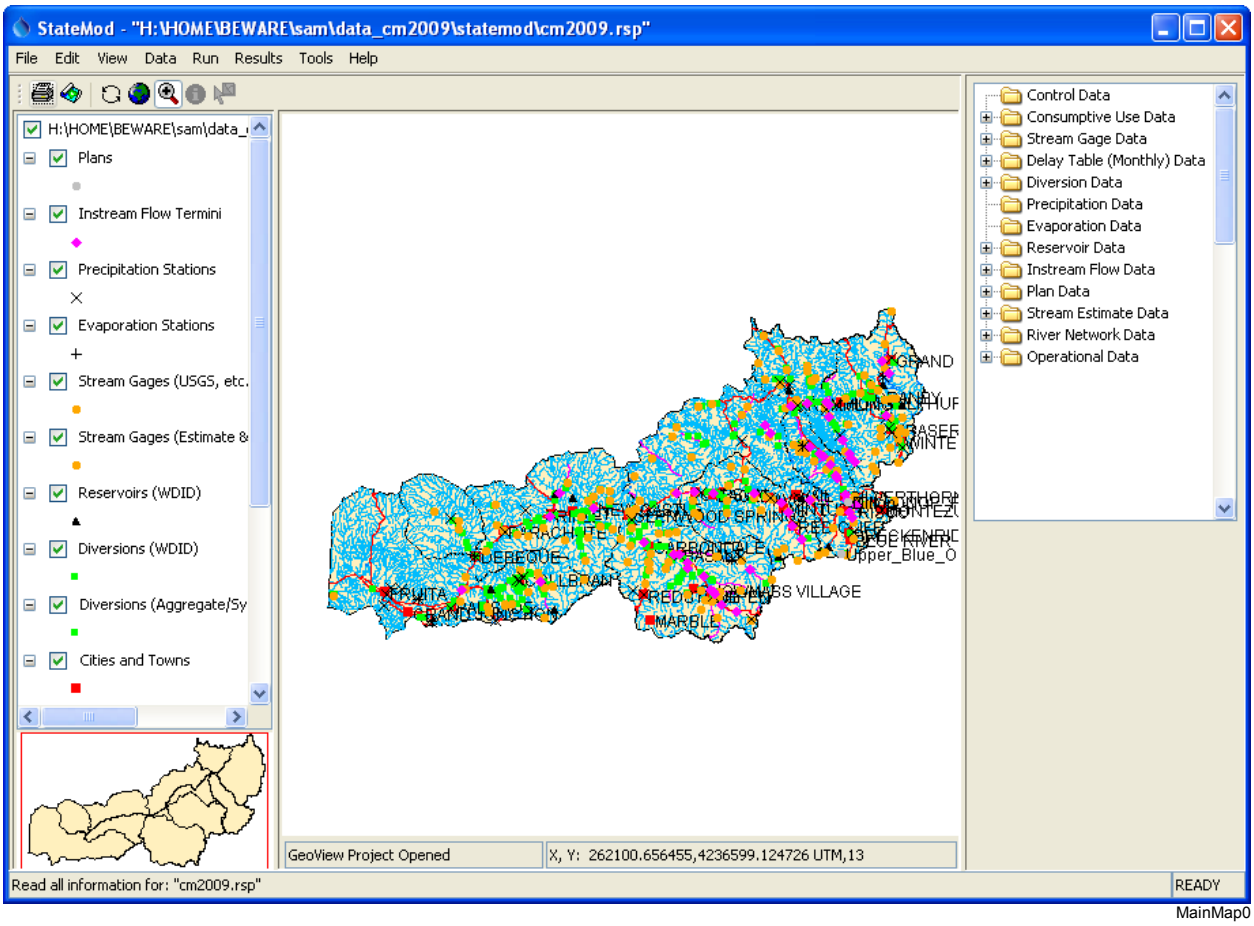
```
#----------------------
# Used by the StateMod GUI but not by StateMod the model...
Network                                 = ..\network\cm2009.net
GeographicInformation                   = cm2009.gvp
# Would like to use the following but StateMod does not support (generates error)
# Instead, use ${WorkingDir} in StateMod GUI configuration file
#StateModExecutable                    = ..\Bin\StateMod-13.00.00.exe
# ... end StateMod GUI items
#----------------------
```

- GIS files are not completely configured:
  - GIS files (shapefiles or CSV files for point data) are not distributed with dataset

  - Geographic information file (also referred to as GeoView Project, **.gvp*) is not provided or is improperly configured
  - Adding stations in the GUI does not result in GIS files being updated (this is a desirable enhancement – see **Chapter 8** for discussion of map features)
- StateMod development has occurred and the GUI has not been updated consistently

The above issues need to be addressed in a more uniform fashion for all StateMod data sets.

After a response file has been selected and a data set has loaded, the main interface will appear similar to the following figure. Note that the location of the response file that was selected is shown in the window title and is set as the `WorkingDir` property mentioned in the StateMod GUI configuration file mentioned above.




MainMap0

**StateMod GUI Main Interface After Opening a Data Set**

A map will be shown if a GeoView Project (**.gvp*) file is referenced in the response file. See **Appendix – Configuring Spatial Data for the StateMod GUI** for more information. The main interface window for the StateMod GUI has several components.

***Title Bar*** (top) — The title bar indicates the response file for the data set and whether any data have been modified.

***Menu Bar*** — The menu bar at the top of the interface contains the menus for each of the major windows

(below title) available within the interface. Each menu is discussed in the following sections.

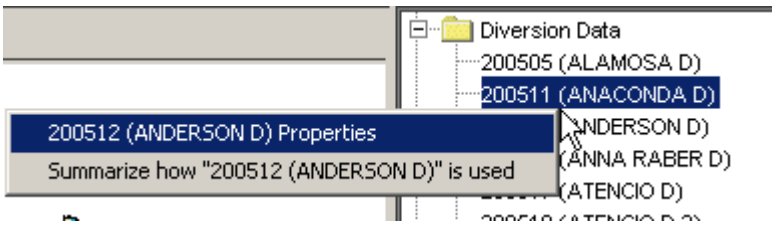
***Map Display*** The map display area occupies most of the main interface and displays spatial data. General features of the mapping tool are described in the **GeoView Mapping Tools Appendix**. See **Chapter 8 – Using the Map** for more information about using the map tools within the StateMod GUI. The left side of the map display lists layers that are shown in the main map. An overview map is shown in the lower-left corner of the interface. Buttons above the layer list and map allow the map to be printed, saved as an image, and provide additional functionality.

If ***Zoom*** mode is selected ( ), a box can be drawn on the main map to zoom into a smaller area. A box can also be drawn on the overview map, which will show the visible extent of the main map.

If ***Info*** mode is selected ( ), drawing a box on the map with the mouse will print basic geographic information about the selected features.

If ***Select*** mode is selected ( ) and a layer is selected in the layer list, clicking on a single feature will display the data window for that feature. Only one feature can be selected (a box cannot be used). For example, use ***Select*** mode to select a diversion from the map.

***Data Set Tree*** (right of map) The data set tree displays the primary component groups for the data set. The groups can be expanded to show the data items for each group.

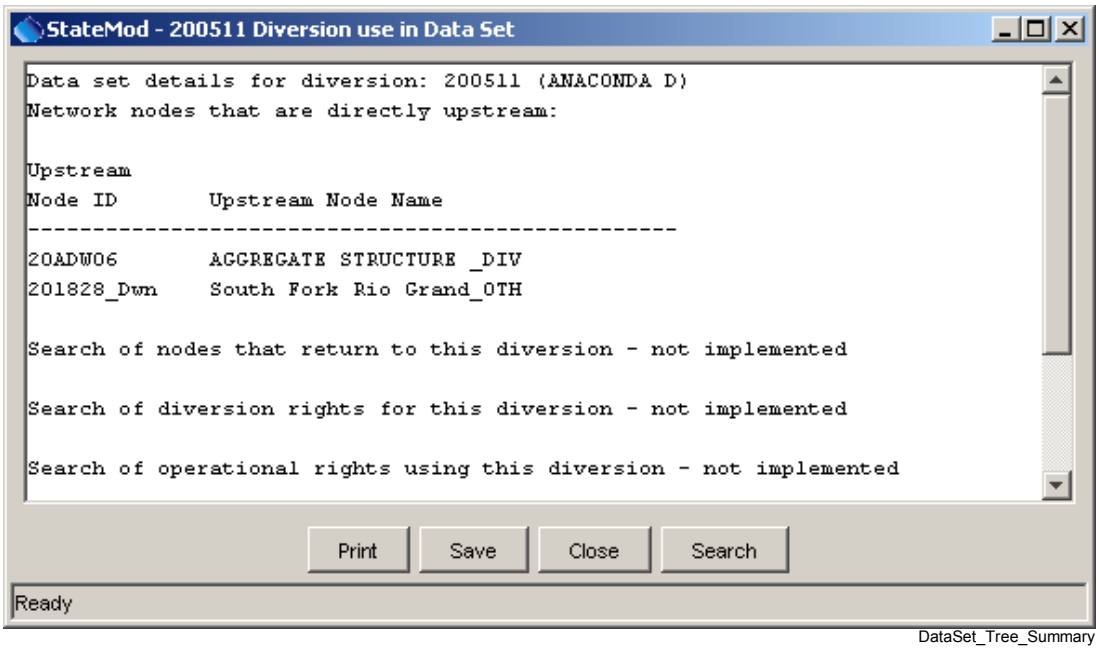



DataSet_Tree

**Data Set Tree Illustrating Display Tools**

Right clicking on an item displays a popup menu. Selecting ***Properties*** results in the main display window for the data component being shown (see **Chapter 5 – Viewing and Editing Data**).

Selecting ***Summarize*** displays a summary of how the component is used (this feature has not been fully implemented):




DataSet_Tree_Summary

**Summary of How a Model Node is Used**

It is envisioned that this feature will be enhanced in the future to facilitate modeling.

***Status Message Area*** (bottom)

Important instructional and feedback messages are displayed in the status message area at the bottom of the main interface. These messages are also printed to the StateMod GUI log file.

The following sections describe the StateMod GUI menus, which are enabled as appropriate.

## 3.3 File Menu – Open and Save Data Sets

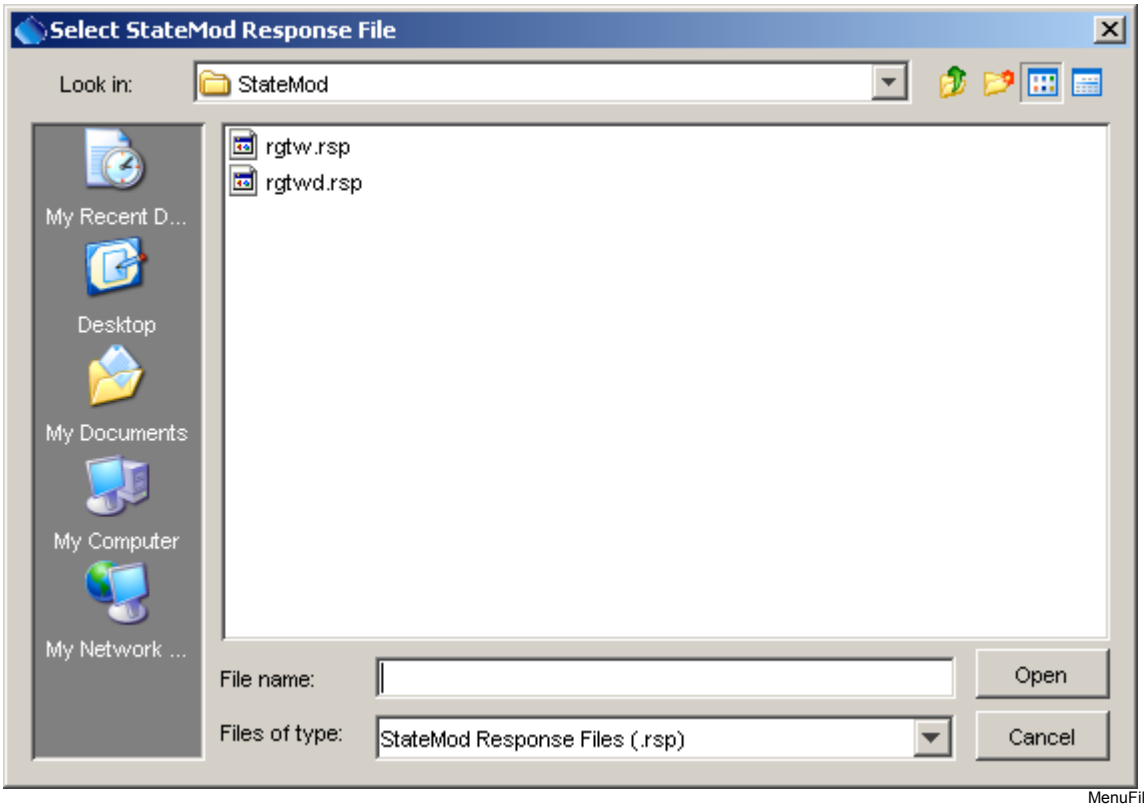
The following choices are available from the ***File*** menu (note that some choices are disabled until a data set has been opened):


Menu_File

The menu choices are discussed in more detail below.

### 3.3.1 File…Open Menu

The ***File…Open*** menu is used to open an existing StateMod data set by selecting a StateMod response (*.rsp*) file, which lists the files in a StateMod data set. The file selector starts from the current working directory, the folder specified by the `DataHomeDefault` configuration property, or the last folder that has been accessed.

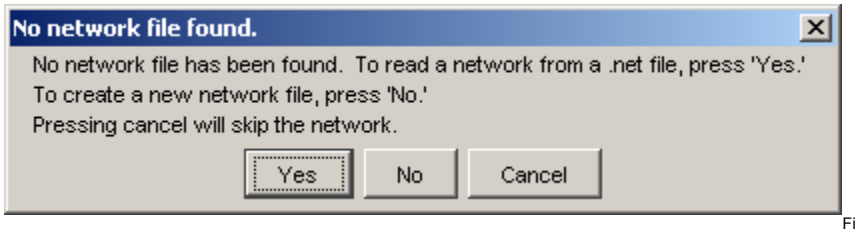



MenuFileOpen

Selecting a response file begins a process to read of all the StateMod data files for the data set. Errors reading any file may result in a warning dialog and that file may not be able to be viewed or edited by the GUI.

Once the StateMod GUI completes processing all input files, the map is displayed in the main interface if a GeoView project has been specified in the response file. The StateMod GUI filters spatial data so that only features that match identifiers in StateMod files are displayed. See the **Configuring Spatial Data for the StateMod GUI Appendix** for more information preparing spatial data for the StateMod GUI.

The model network file (**.net*) is used by the StateMod GUI to represent the model node network as a diagram. This file is not directly used by the StateMod software but is key to StateDMI data processing and to visualizing the model network. StateMod does use a river network file (**.rin*), which is compatible with the **.net* file. Older StateMod data sets and response files do not include the network file (**.net*). If the network file (**.net*) is not listed in the response file, the following dialog will be shown at startup:




File_Open_NetworkWarning

**Network File Warning**

At this point, the network file (**.net*) can be be added to the data set and will be displayed. Subsequently opening the data set will not display the warning. **Chapter 4 – The Model Network** provides more information about creating and modifying the model network.

### 3.3.2 File…New - Create a New Network or Data Set

The ***File…New*** menu allows the creation of a new data set or network. **It is highly recommended that the StateMod GUI <u>not</u> be used to create new data sets and instead that a data-centered approach is used involving StateDMI, TSTool, or equivalent software**. Although access to the State of Colorado's HydroBase database provides the most functionality with StateDMI, the software also can use comma-separated-value (CSV) files to provide input for processing. TSTool is a general tool that processes data from many sources. Using the StateMod GUI to edit a new data set has the following limitations (also see the **Release Notes**):

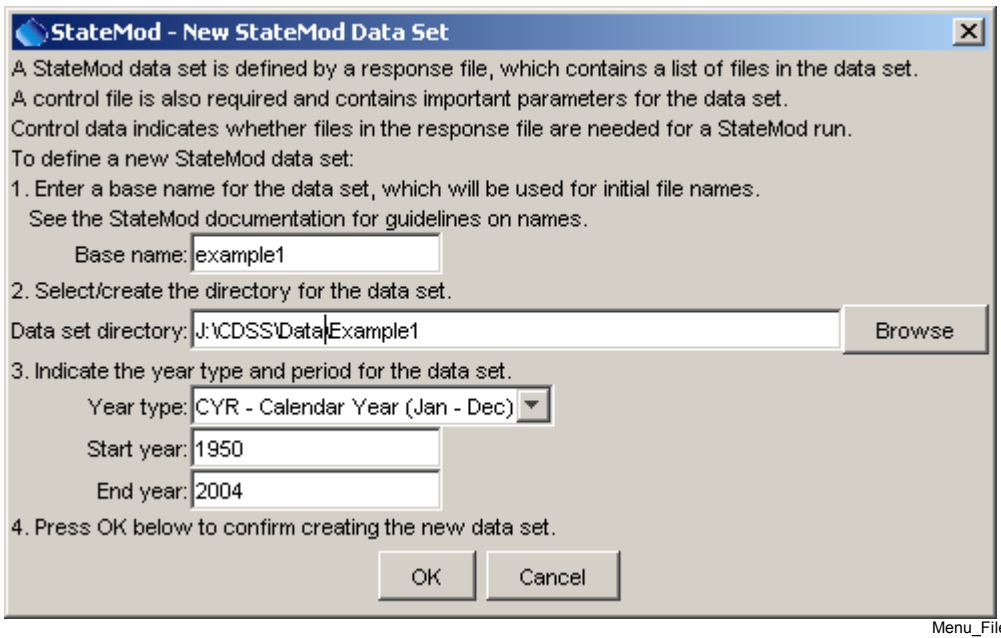
- Not all interactive editing features in the StateMod GUI are completely functional, for example due to StateMod model enhancements that have not been implemented in the GUI.
- The GUI does not include functionality to annotate edited model files. Consequently, tracking of edits does not occur.
- The manual effort necessary to interactively input data will be much more than if using the automated processing provided by StateDMI and TSTool.

Nevertheless, the following sections describe how to create a new data set in the GUI.

#### *Creating a New Data Set*

Use the ***File…New…Data Set*** menu to initialize a new data set, as follows:

1. Start the StateMod GUI without loading a data set.
2. Select ***File…New…Data Set*** to open a new data set and provide initial information:

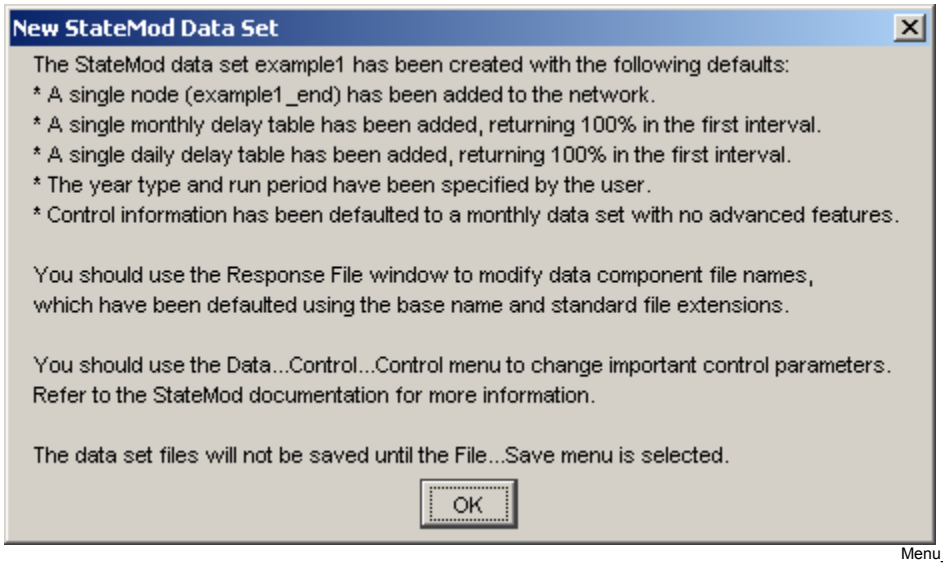



Menu_File_New_DataSet1

**Providing Information to Initialize a New Data Set**

The data set directory must already exist.

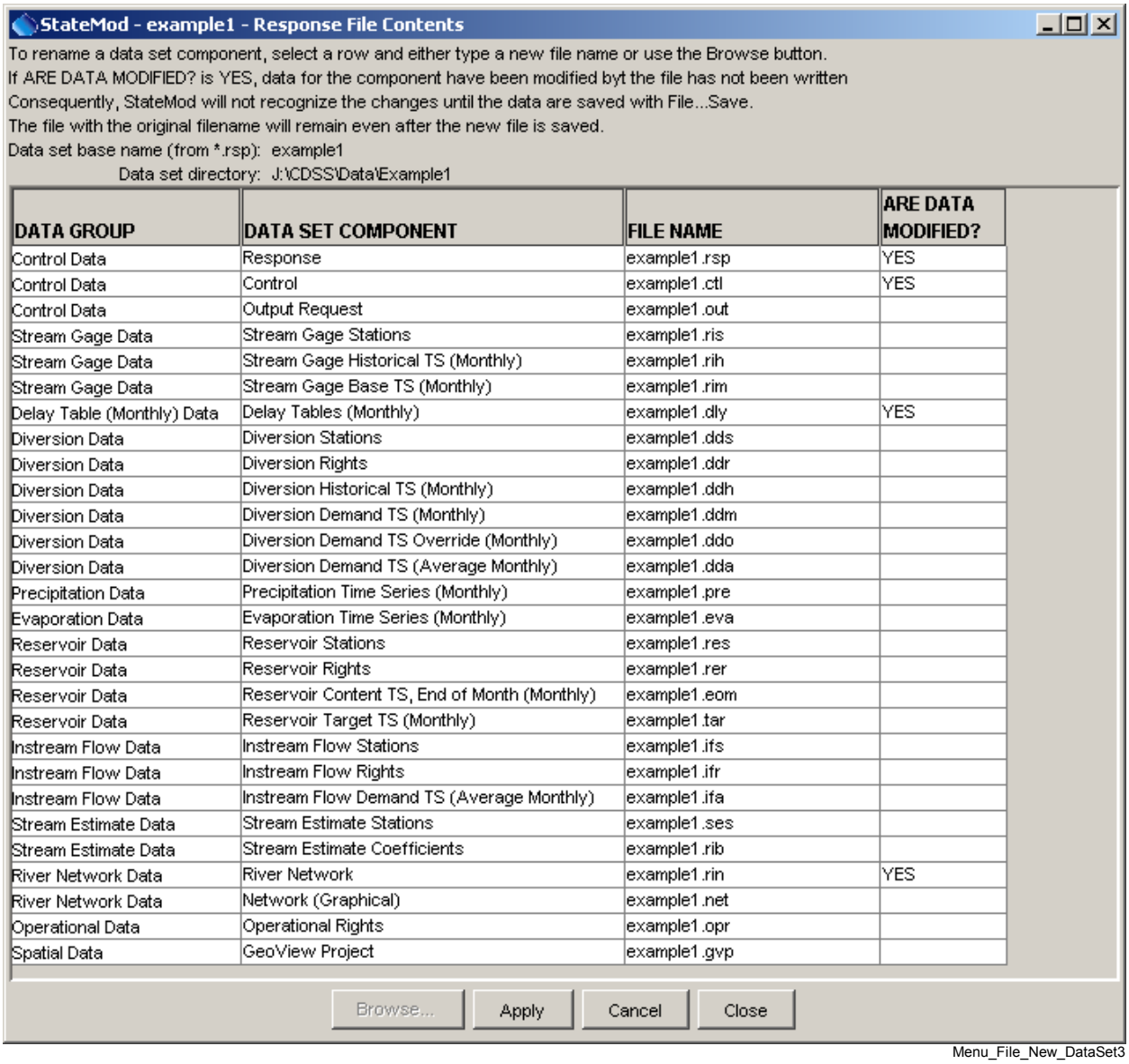
3. Initial data set files will be created as indicated in the following confirmation dialog:




Menu_File_New_DataSet2

**Confirmation of New Data Set**

4. The response file contents are displayed as shown in the following figure:




| DATA GROUP | DATA SET COMPONENT | FILE NAME | ARE DATA MODIFIED? |
|---|---|---|---|
| Control Data | Response | example1.rsp | YES |
| Control Data | Control | example1.ctl | YES |
| Control Data | Output Request | example1.out | |
| Stream Gage Data | Stream Gage Stations | example1.ris | |
| Stream Gage Data | Stream Gage Historical TS (Monthly) | example1.rih | |
| Stream Gage Data | Stream Gage Base TS (Monthly) | example1.rim | |
| Delay Table (Monthly) Data | Delay Tables (Monthly) | example1.dly | YES |
| Diversion Data | Diversion Stations | example1.dds | |
| Diversion Data | Diversion Rights | example1.ddr | |
| Diversion Data | Diversion Historical TS (Monthly) | example1.ddh | |
| Diversion Data | Diversion Demand TS (Monthly) | example1.ddm | |
| Diversion Data | Diversion Demand TS Override (Monthly) | example1.ddo | |
| Diversion Data | Diversion Demand TS (Average Monthly) | example1.dda | |
| Precipitation Data | Precipitation Time Series (Monthly) | example1.pre | |
| Evaporation Data | Evaporation Time Series (Monthly) | example1.eva | |
| Reservoir Data | Reservoir Stations | example1.res | |
| Reservoir Data | Reservoir Rights | example1.rer | |
| Reservoir Data | Reservoir Content TS, End of Month (Monthly) | example1.eom | |
| Reservoir Data | Reservoir Target TS (Monthly) | example1.tar | |
| Instream Flow Data | Instream Flow Stations | example1.ifs | |
| Instream Flow Data | Instream Flow Rights | example1.ifr | |
| Instream Flow Data | Instream Flow Demand TS (Average Monthly) | example1.ifa | |
| Stream Estimate Data | Stream Estimate Stations | example1.ses | |
| Stream Estimate Data | Stream Estimate Coefficients | example1.rib | |
| River Network Data | River Network | example1.rin | YES |
| River Network Data | Network (Graphical) | example1.net | |
| Operational Data | Operational Rights | example1.opr | |
| Spatial Data | GeoView Project | example1.gvp | |

Menu_File_New_DataSet3

**Initial Response File Contents for New Data Set**

The file names should be modified as appropriate and then press ***Apply***. Pressing ***Cancel*** will cancel creation of the new data set. Pressing ***Close*** apply any changes and close the window. The initial files can be saved with ***File…Save***.

5. Additional data can then be added/modified using other display windows and consequently saved with ***File…Save***. If no additional data are added for data components, the files will be written with headers but no data records.
6. Because the model network (**.rin*) requires special setup attention

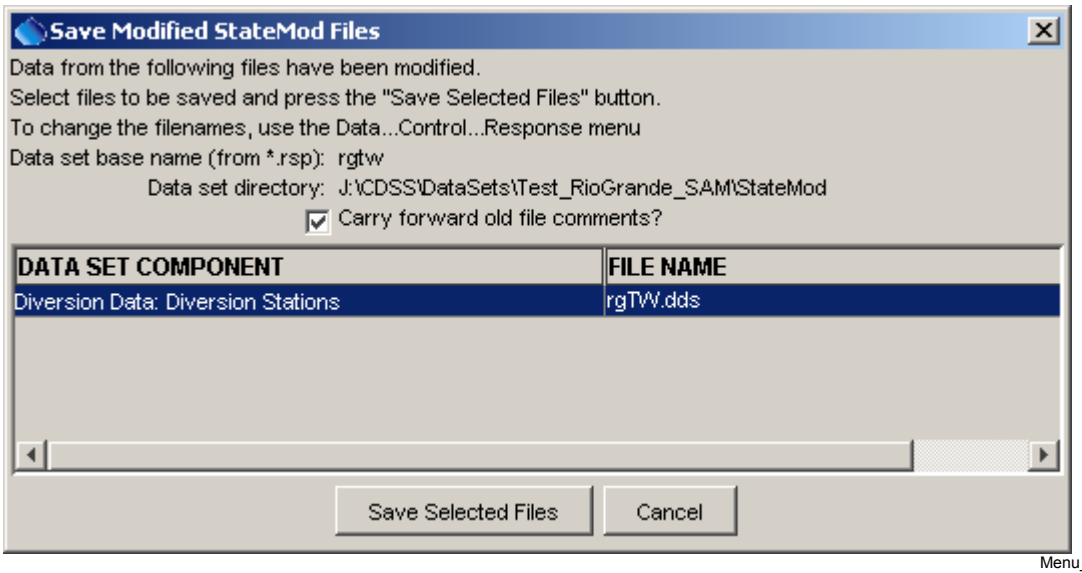
### *Creating a New Network*

Use the ***File…New…Network*** to create a new network (**.net*) file. This is needed when creating a new data set (do after the steps described above) or may be needed because an older StateMod data set does

not include a network file. See **Chapter 4 – The Model Network** for more information. This feature is currently disabled pending software enhancements. Instead, use the network editor in the StateDMI software to create a new network, and use a data-centered approach to create StateMod files based on that network.

### 3.3.3 Save Data Files

The ***File…Save*** menu displays the following dialog, which indicates the StateMod files that need to be written due to user edits:




Menu_File_Save

Only StateMod files that need to be updated are listed and by default all listed files are selected. Only the selected files will be written. Use the ***Shift*** key to select a range of items and the ***Ctrl*** keys to toggle an item on and off. Selecting the ***Cancel*** button will result in no files being written.

The ***Carry forward old file comments*** checkbox, if selected, will carry forward previous file header comments (e.g., from StateDMI, TSTool, or other software), allowing the full modification history of the file to be recorded.

The above dialog is automatically displayed if you try to run StateMod from the StateMod GUI and data have been modified in memory but have not been saved to files.

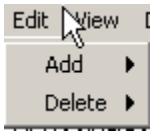
### 3.3.4 File…Exit

The ***File…Exit*** menu exits the StateMod GUI, first asking for confirmation. If data files have been modified, an option to save the files will be provided. The ***X*** in the StateMod GUI menu bar is equivalent to ***File…Exit***.

There is a known issue that sometimes the GUI thinks that files have been modified when the user has not actually made any changes, resulting from the GUI updating file formats or in some cases not understanding newer file formats. This issue needs to be resolved with more development resources.

## 3.4 Edit Menu – Add and Delete Model Data

The following choices are available from the ***Edit*** menu (note that some choices are disabled until a data set has been opened) and provides capabilities to modify a StateMod data set.




Menu_Edit

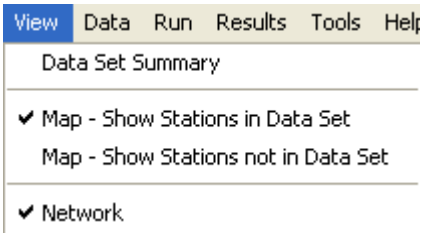
These menus are used for adding new model nodes or related data components and are discussed in **Chapter 4 – The Model Network**.

Once model nodes have been added, the Data menu can be used to view and modify the data components (see **Chapter 5 – Viewing and Editing Data**).

**It is highly recommended that the StateMod GUI not be used to edit data sets and instead that a data-centered approach is used involving StateDMI, TSTool, or equivalent software**. See the discussion above under creating a new data set and the **Release Notes**.

## 3.5 View Menu – Enable/Disable Display Features

The following choices are available from the ***View*** menu (note that some choices are disabled until a data set has been opened):

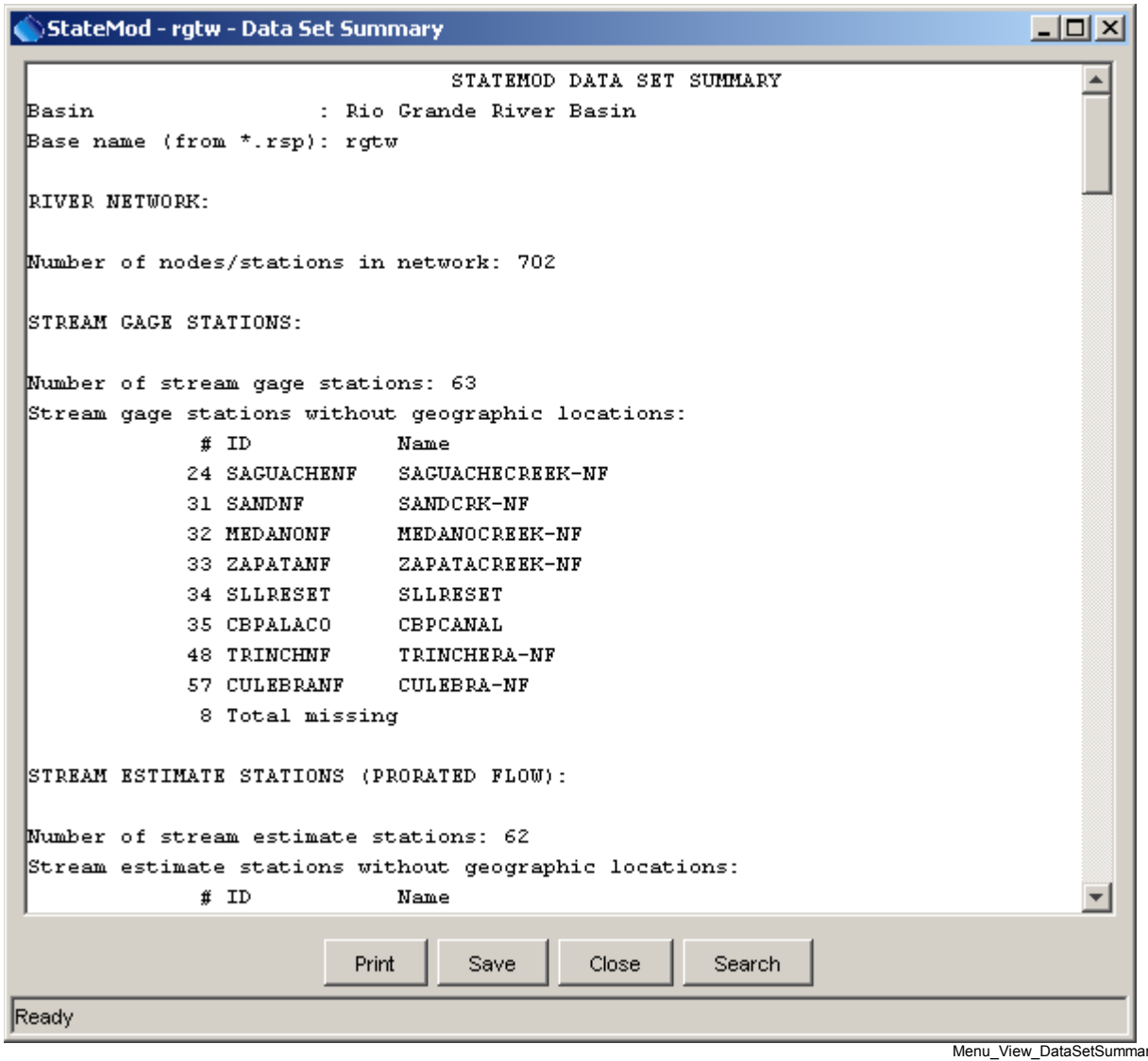



Menu_View

The following sections discuss each menu.

### 3.5.1 Displaying a Data Set Summary

The ***View…Data Set Summary*** menu displays a summary of the basin:




Menu_View_DataSetSummary

**Data Set Summary**

This information is useful for evaluating the complexity of the network. It is envisioned that this summary will be enhanced to include information to help configure and troubleshoot data sets. The above example illustrates how the summary can be used to identify stations that are not included in the spatial data layers.

### 3.5.2 View the Map with Stations in the Data Set

The ***View…Map – Show Stations in Data Set*** menu displays the map (read from information in the **.gvp*), if spatial data have been configured for the data set. See the **Configuring Spatial Data for the StateMod GUI** appendix and **Chapter 8 – Using the Map** for more information about the map interface. This menu option is by default turned on and displays stations that match data set locations.

### 3.5.3 View the Map with Stations Not in the Data Set

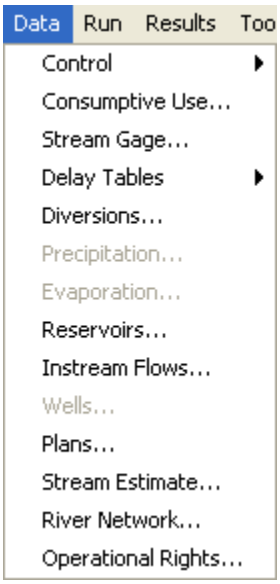
The ***View…Map – Show Stations not in Data Set*** menu causes the map to display stations that are not in the data set. This is useful to show additional spatial data that are available in map layers but which are not currently included in the data set (or at least have identifiers that don't match the data set).

### 3.5.4 View the Model Network

The ***View…Network*** menu displays the model network (read from **.net*), if a network is available for the data set. See **Chapter 4 – The Model Network** for more information about using the network tools.

## 3.6 Data Menu – View and Edit Data

The following choices are available from the ***Data*** menu (note that some choices are disabled until a data set has been opened):

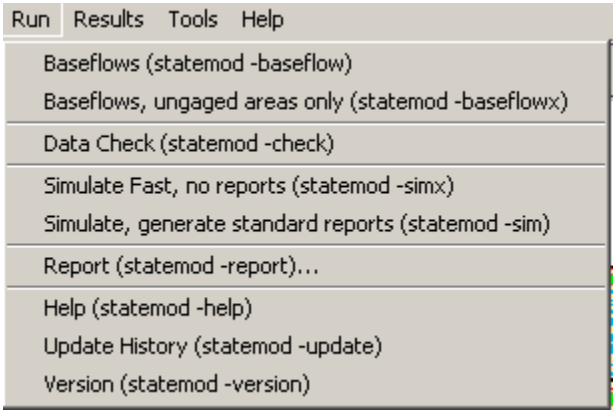



Menu_Data

These menus are used to display the main windows for primary data components. These displays allow viewing or modifying existing stations and other data (see **Chapter 5 – Viewing and Editing Data**). To add or delete data, use the ***Edit*** menu, as discussed in **Chapter 4 – The Model Network**.

## 3.7 Run Menu – Run StateMod

The following choices are available from the ***Run*** menu (note that some choices are disabled until a data set has been opened):

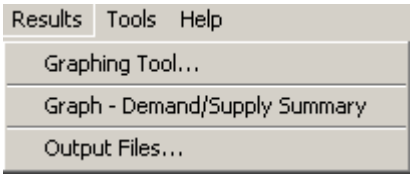



Menu_Run

These menus are used to run StateMod software and related utility software. See **Chapter 6 – Running StateMod** for more information.

## 3.8 Results Menu – View StateMod Results

The following choices are available from the ***Results*** menu (note that some choices are disabled until a data set has been opened):

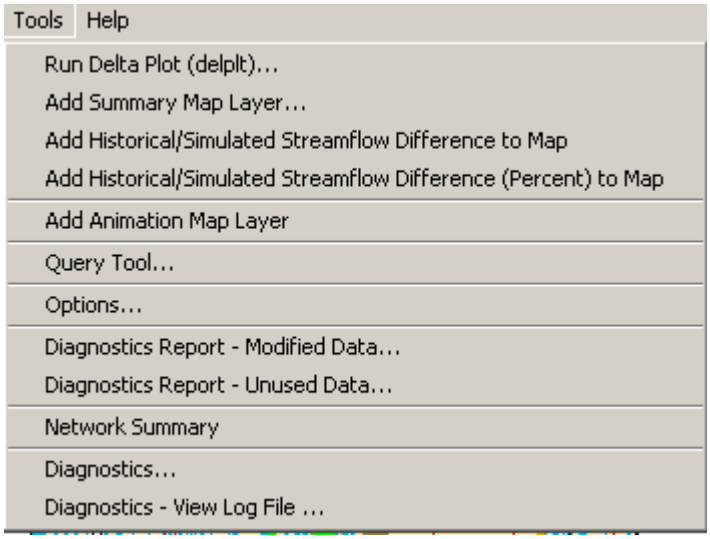



Menu_Results

These menus are used to view StateMod results. See ***Chapter 7 – Viewing StateMod Results*** for more information.

## 3.9 Tools Menu

The following choices are available from the ***Tools*** menu (note that some choices are disabled until a data set has been opened):

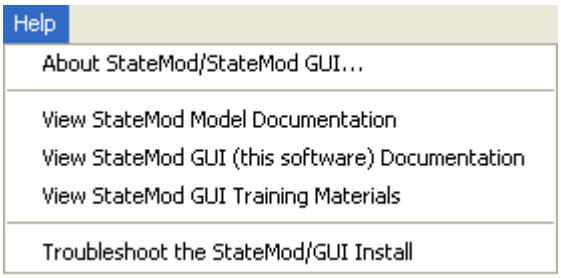



Menu_Tools

Tools provide various useful features and are described in **Chapter 9 – Tools**.

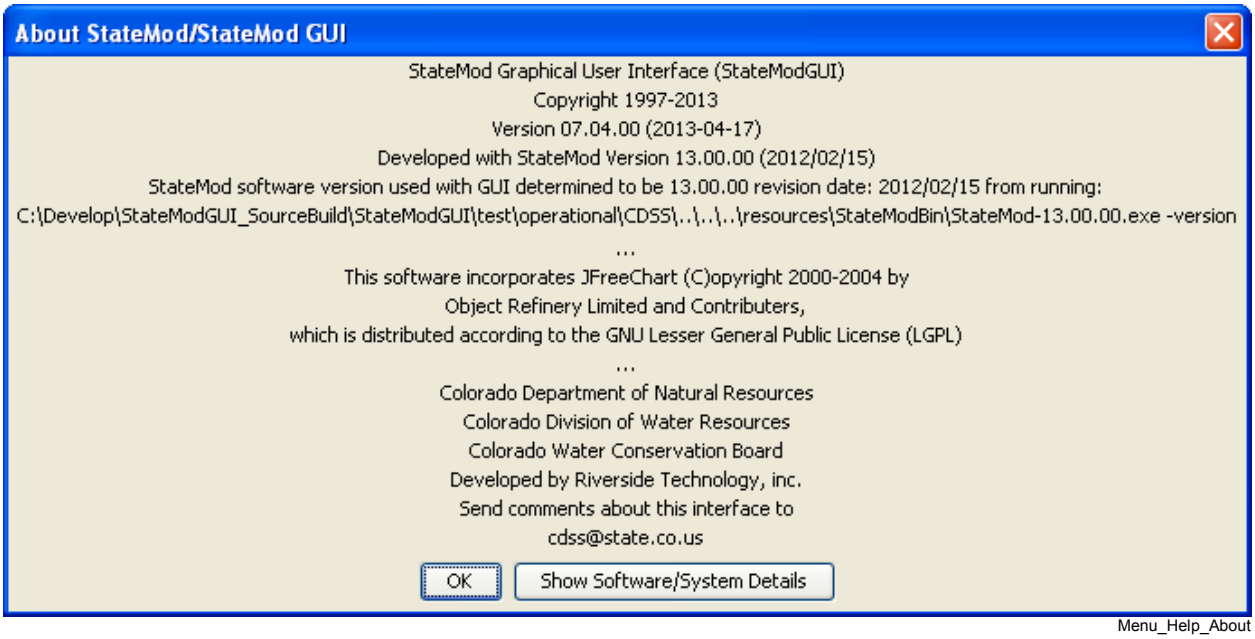
## 3.10 Help Menu

The following choices are available from the ***Help*** menu:




Menu_Help

The ***Help…About StateMod/StateMod GUI*** menu displays the software version for the StateMod GUI, the StateMod model version that was used for development, and the StateMod model version that has been detected by the StateMod GUI at run time.




Menu_Help_About

Use the software version information when reporting problems or suggesting enhancements.

# 4 The Model Network – Adding and Deleting Data

Version 06.03.02, 2006-03-04

**It is recommended that modelers use a data-centered approach to creating data sets, for example using the StateDMI and TSTool software within the State of Colorado's Decision Support Systems. This will ensure that creation of data sets is automated, repeatable, and transparent.** The following are known limitations when editing data using the StateMod GUI (see also the **Release Notes**):

1. The GUI is not current with the StateMod model features.
2. Modeling conventions have not been defined to ensure that spatial data locations are defined for all model data.
3. The GUI does not track edits. Consequently regenerating the files using an automated approach will lose manual edits.

The model network describes the connectivity between stations. Because other data are associated with stations, the network is a key component in organizing and navigating a data set. This chapter presents an overview of the network and describes how data can be added to or deleted from the network. Once data have been added, the data viewing and editing windows described in **Chapter 5 – Data Viewing and Editing** can be used. The following information is most useful to someone who is going to make changes to the model network.

## 4.1 Model Network Overview

The model network consists of the following information:

1. A list of model nodes of various data types (e.g., diversions, reservoirs, stream gages).
2. Connectivity information that indicates the downstream nodes for each node (it is implied that river networks collect but do not diverge).
3. Information about area and precipitation above flow locations, for use in prorating known flows to estimated streamflow locations.
4. Node coordinates, annotations, and symbol properties, to allow the network to be drawn in schematic fashion.

Although the StateMod river network file contains river node identifiers and downstream information, it does not contain some of the other information listed above. Consequently, the StateMod GUI uses an XML network file (**.net*) that is compatible with, and in addition to the StateMod river network file (**.rin*).

The coordinates in the model network data allow a schematic representation of the network to be created. These coordinates normally do not correspond to geographical coordinates.

The StateDMI documentation provides a full description of the network file and is not duplicated here. The following sections focus on creating and modifying the model network within the StateMod GUI. StateDMI also allows creation and editing of the network and should be used to edit the network when an automated process is being used to create other data files from the network file. The StateMod GUI

should be used to edit the network if an automated process is NOT being used (e.g., to make minor changes to create a scenario or when tools like StateDMI are not available).

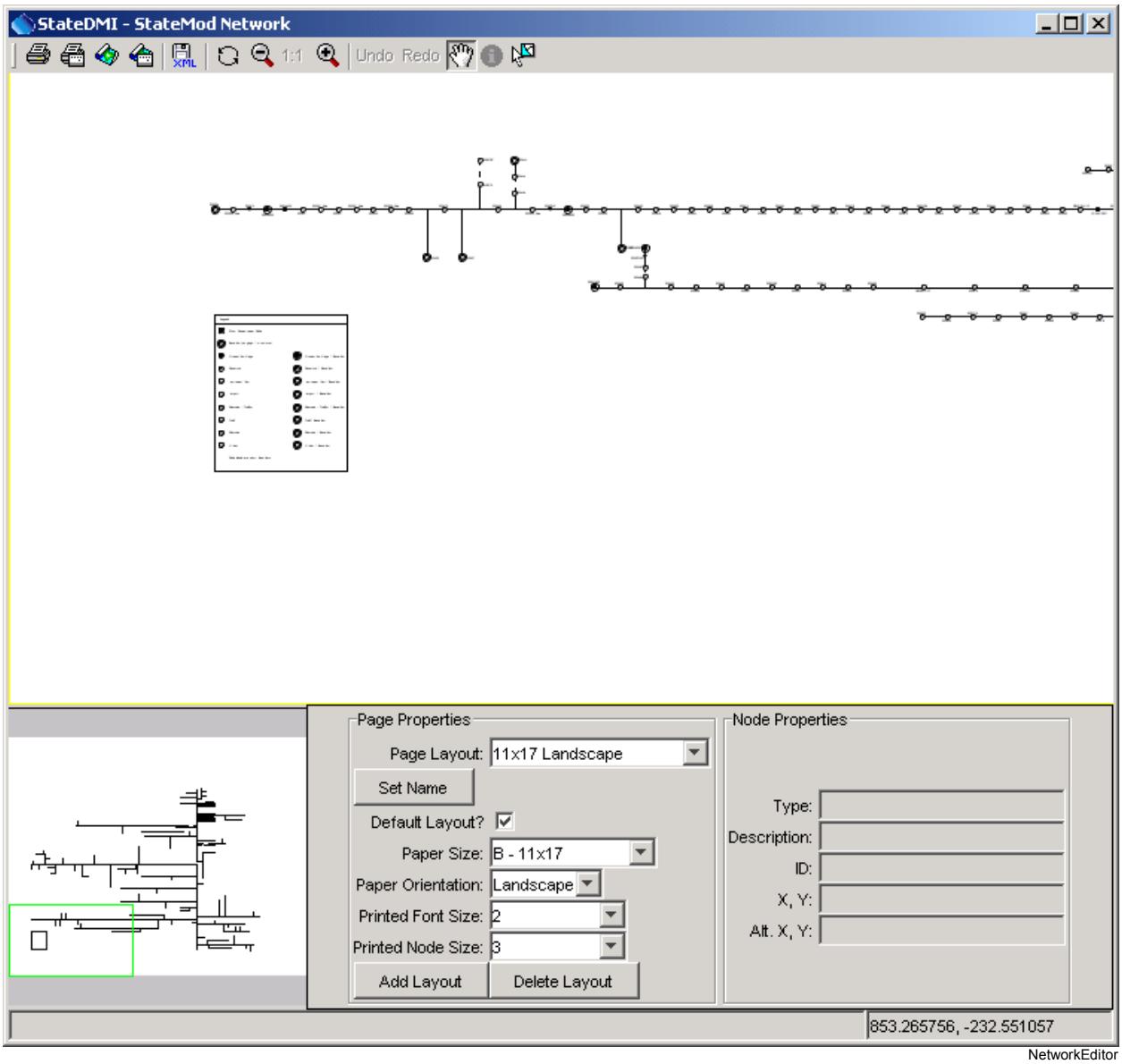
The model network for a StateMod basin model typically consists of a single end node to which all streams in the basin drain. Upstream of the end node is a branching stream network, which may also include human-constructed canals, ditches, and reservoirs. Key locations on the stream network, corresponding to StateMod station types, are locations where model calculations need to occur. A StateMod data set DOES NOT contain river (line) data. Instead, all model locations are stations and the station relationships are defined in the StateMod river network file.

Water is introduced into points in the network using base flow locations, which are stream gages or stream estimate stations, the latter being locations where streamflow is estimated by prorating flows from gaged locations. The resulting water is then allocated through the system, as the simulation occurs. To properly define the system, station types in the network must accurately match physical system features. The network diagram facilitates visual checking of the network and allows printing of the network, for more comprehensive review.

## 4.2 Model Network Interface

The ***View…Network*** menu displays the editor window for the StateMod generalized model network (**.net*). In addition to the basic display of the network diagram, it is envisioned that additional tools will be added to the network editor to allow for more targeted use in the StateMod GUI (and StateDMI), for example to display the return flow locations, and to display the stations that are referenced in an operating rule.

The following figure shows the network editor after a network file has been read and displayed:




NetworkEditor

**Network Editor**

The network editor consists of the following areas:

- Tools (top) – initiate actions (e.g., printing), switch mode, edit tools
- Main canvas (middle) – area where editing occurs
- Overview window (lower left) – indicates the current view as a subset of the total network
- Page properties (lower middle) – the settings used for the network display, if printed
- Node properties (lower right) – the properties of the node that was last selected.

### 4.2.1 Network Tools

The tools that are available include the following:

Print the entire network using the selected layout (page size, orientation, etc.) This is useful for generation of final products.

Print the visible network using letter-sized paper. This is useful for troubleshooting or reviewing specific parts of the network.

Save the entire network to an image file.

Save the visible network extent to an image file. This is useful for creating inserts for documents.


Save the network to the XML file.

Refresh the network (redraw).

Zoom out by 50%, based on the current extent.


Reset the scale to match the layout.

Zoom in by 50%, based on the current extent.


If a node position has changed, allow it to be undone (or redone).

Pan the visible extent of the network – currently this is the default when clicking on other than a node.

Information tool – currently unused. It is envisioned that this tool could be enabled to show model-related data from a data set.

Select a feature – currently this is the default when clicking on a node.

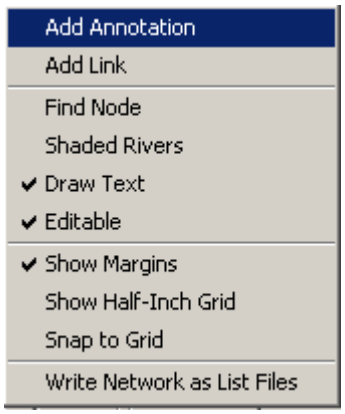
### 4.2.2 Main Canvas

The main canvas displays the network for the current scale and location. Use the tools to scroll, pan, or zoom to a specific region.

To move an existing node, select it with the mouse and drag to the new location. Use the ***Undo/Redo*** tool if necessary to discard a change.

See sections below for information about adding/moving/deleting nodes and other actions.

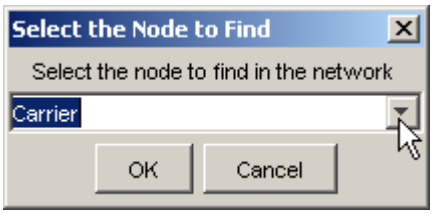
Right-clicking on the canvas (not near a node), displays the following menu:




NetworkEditor_Popup

The actions for the menu items are described in the following table.

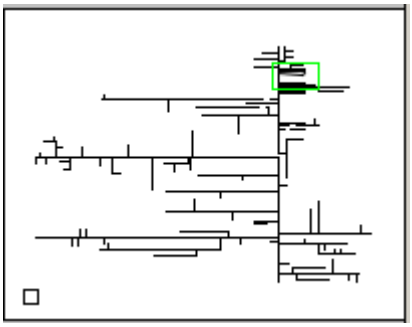
**Network Editor Popup Menu Items**

| Menu Item | Action |
|---|---|
| ***Add Annotation*** | Add an annotation at the point where the mouse was clicked. See **Section 4.2.7** below. |
| ***Add Link*** | Add a link between nodes. See **Section 4.2.8** below. |
| ***Find Node*** | Display the following dialog, listing all nodes in the network.<br>  <br><br>NetworkEditor_Popup_FindNode<br>After selecting a node and pressing ***OK***, the network will scroll so that the selected node is in the center of the network window. |
| ***Shaded Rivers*** | If selected, shade the rivers based on stream order. This is useful to emphasize upstream to downstream progression. |
| ***Draw Text*** | If selected, draw text labels on the network. Text can be turned off if only the lines need to be printed. |

| Menu Item | Action |
|---|---|
| ***Editable*** | If selected, the network is editable. If it is important to protect a network from editing, the network can be made non-editable. Editing actions will then be prohibited in the session. |
| ***Show Margins*** | If selected, the page margins are shown, representing an approximate boundary within which drawing should be limited. It is recommended that network features not extend into the margins. |
| ***Show Half-Inch Grid*** | If selected, a grid of lines will be drawn at half-inch intervals. This is useful for layout purposes. |
| ***Snap to Grid*** | If selected, nodes will be restricted to being positioned on grid lines. |
| ***Write Network as List Files*** | Prompt for a base file name and then write delimited list files for each station type, to be used as lists of stations with commands files. Each file is listed in order of upstream to downstream. This recognizes that it can be more generic to use list files with StateDMI processing, rather than reading from the network itself. This approach is being evaluated as list files are used. Issues to be resolved include:<br>1. DIV and D&W nodes both exist in the network and are written as separate lists. Therefore two commands may be needed when processing the lists.<br>2. Stream gages (FLO nodes) are written as one list and baseflow stations (FLO and other stations where baseflow is True) are written as separate lists. Users must decide which list to use. |

#### 4.2.3 Overview Window

The overview window indicates the current extent of the network in the main canvas.

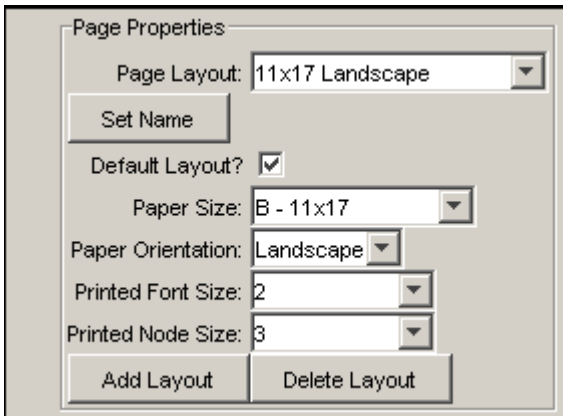


NetworkEditor_Overview

Click anywhere in the overview window to center the main canvas view on that point. Or, drag the overview window extent box to a new location to reposition the network in the main canvas.

### 4.2.4 Page Properties

The page properties can be set for multiple layouts using the ***Page Properties*** settings.




NetworkEditor_PageProperties

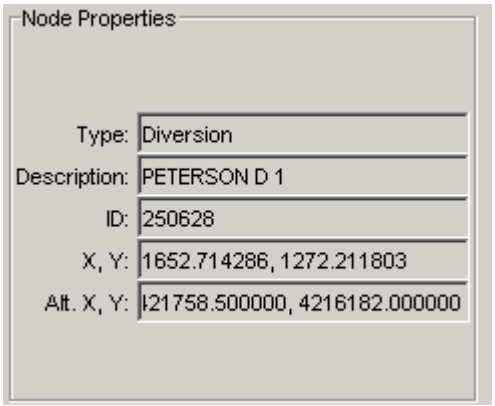
Because one of the primary products related to the network is a printed network diagram, the network is essentially configured as a document. Therefore, the graphics and text on the diagram are scaled (unlike some map and graph displays where the text point size is constant even when the data scale changes).

Modelers responsible for data sets should define one or more layouts for the network to allow printing on common page sizes. Often, there is so much detail on the network that a hard copy can only be printed on large paper sizes. However, more unreadable versions may be appropriate for review. Once layouts are defined, only minor changes should be required. It is recommended that the ***Page Layout*** name include the page size and orientation.

Network editing should typically occur using the page layout that will be used in production printouts. Differences in the relative dimensions of page sizes can cause some scaling in output when switching between layouts.

### 4.2.5 Node Properties

The node properties area in the network editor shows the properties for the most recently selected node.

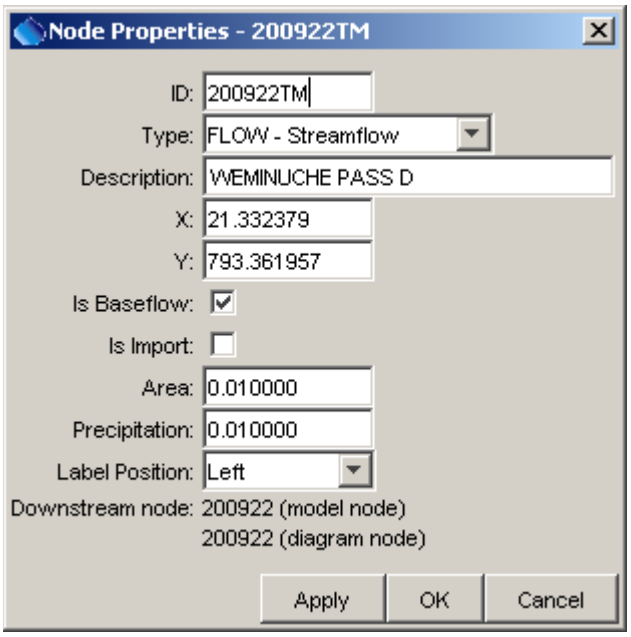



NetworkEditor_NodeProperties

This is useful when scanning network node information. See the next section for information about changing node properties.

### 4.2.6 Adding/Deleting/Changing a Node

To add a node and insure that data in station files is kept consistent with the network, use the ***Edit…Add*** menu items, as described in **Section 4.9** below. To delete a node, use the ***Edit…Delete*** menu items, as discussed in **Section 4.10** below. A node is moved by selecting the node on the network and dragging to a new location. To move multiple nodes draw a box around nodes and then move the group. Node properties for an existing node are edited by selecting a node in the network, right clicking, and pressing the ***Properties*** menu item, which will display a dialog similar to the following:




NetworkEditor_Popup_NodeProperties

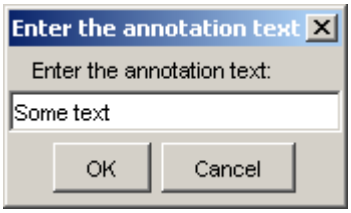
**Node Properties Dialog**

The node types correspond either to StateMod station types or to node types needed for visualization (e.g., confluences), which are not transferred to StateMod files. Although the legacy CDSS Makenet software allowed `Import` and `Baseflow` node types, these types are no longer supported. Instead, node types correspond to StateMod station types, with the `Other` node type used where needed. The ***Is Baseflow*** check indicates that ***Area*** and ***Precipitation*** information are available for the node – these data are used when processing stream estimate stations.

### 4.2.7 Adding/Deleting/Changing Annotations

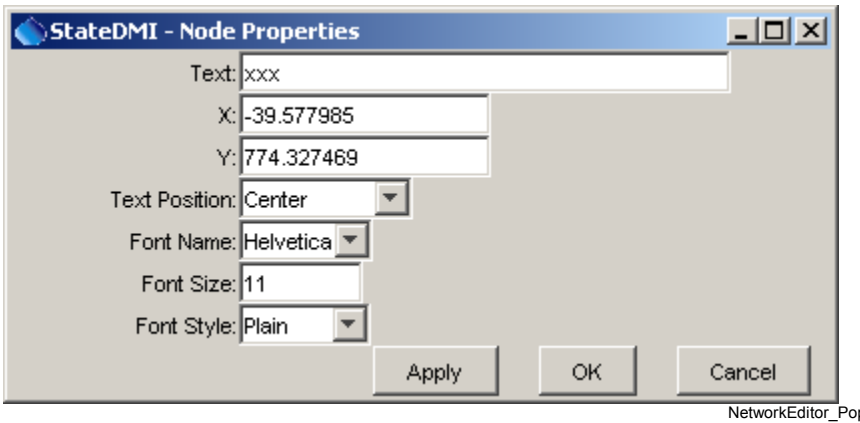
Annotations are text labels that can be drawn on the network. They are typically used for title, author, revision date, stream names, etc., using font sizes appropriate for the information.

To add an annotation, right-click at a point of interest (not near a node) and select the ***Add Annotation*** menu item, which will display the following dialog:




NetworkEditor_Popup_AddAnnotation

Pressing ***OK*** displays the annotation text centered at the point where the mouse was clicked. Once an annotation is added, it can be moved and its properties can be set by right clicking on the annotation anchor point and pressing ***Properties***:




NetworkEditor_Popup_AnnotationProperties

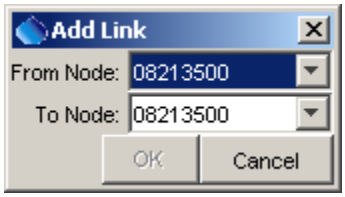
An annotation can be moved by selecting the annotation and dragging it to the new location.

An annotation can be deleted by right clicking on the annotation and pressing the ***Delete Annotation*** menu item.

### 4.2.8 Adding/Deleting Links

Links are dashed lines between nodes, typically used to represent an operational relationship between nodes (e.g., to represent carrier ditches). Annotations can be placed next to links to describe the link.

To add a link, right-click on the network (not near a node) and use the ***Add Link*** menu item. The following dialog will be shown:




NetworkEditor_Popup_AddLink

After selecting nodes and pressing ***OK***, the link will be drawn between the nodes as a straight dashed line.

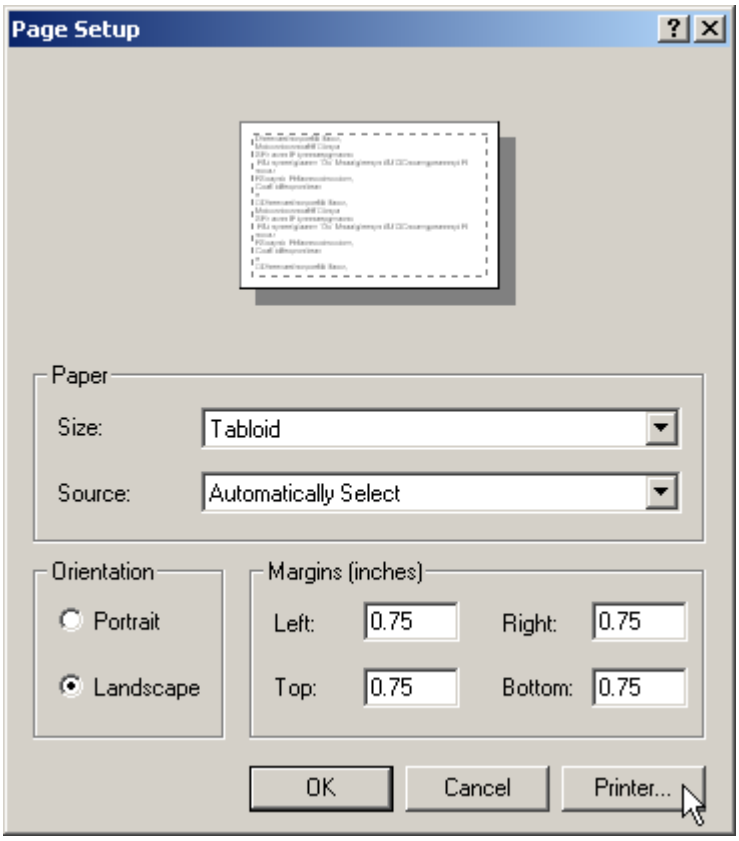
To delete the link, select one of the nodes involved in the link, right-click and select ***Delete Link***. If the node is involved in more than one link, a list of links will be shown.

## 4.3 Printing a Model Network

To print the entire network, use the  tool and follow the procedure described below. To save the visible network as an image, use the  tool and follow the procedure described below. Note that when printing, curved graphics are drawn using a technique called "anti-aliasing," where curves are created by using shades of gray. This may result in graphics that are difficult to read for some page sizes.

When the print tools are used, several dialogs are shown, as required by the Java and Microsoft environments. Although options are available in various dialogs, the following approach is recommended (improvements are being evaluated):

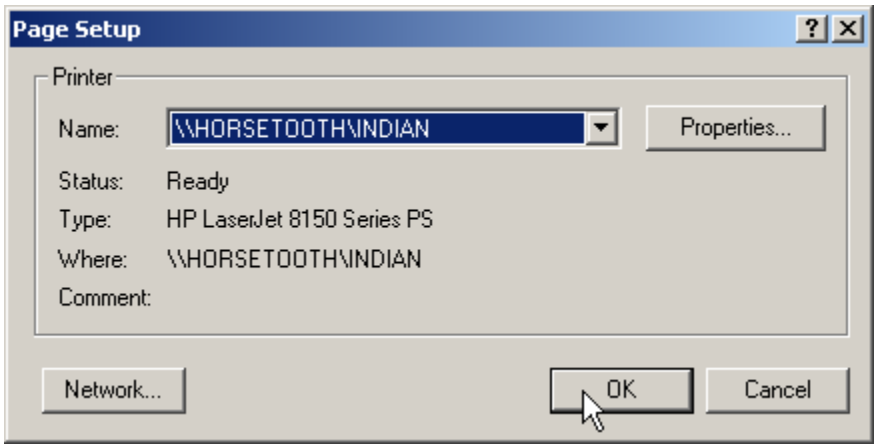
1. After selecting one of the tools mentioned above, a Java ***Page Setup*** dialog will be shown (this should be the same regardless of Windows version):




NetworkEditor_Print1

Select the printer of interest by using the ***Printer…*** button, as discussed in the next item.

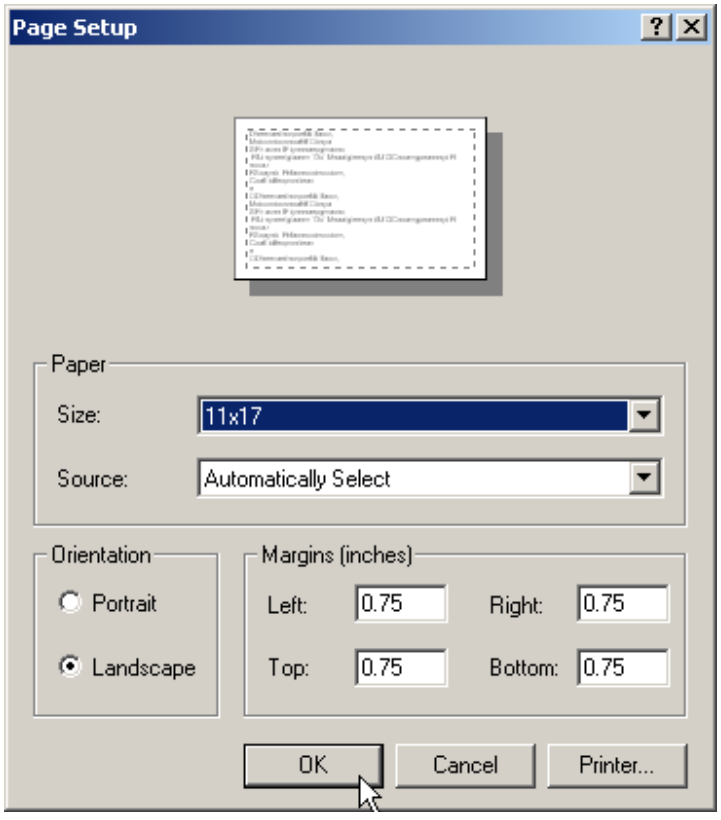
2. A Windows ***Page Setup*** dialog will be shown:




NetworkEditor_Print2

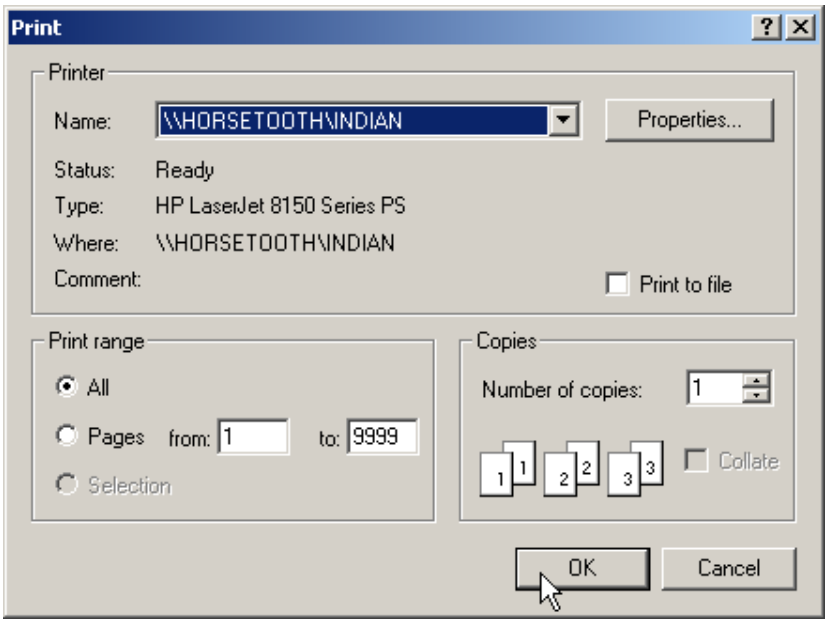
Pick a printer that can handle the page size specified in the current network editor page layout and press ***OK***.

3. In the original dialog, select the paper size to match the current network layout and press ***OK***:




NetworkEditor_Print3

4. A Windows ***Print*** dialog will be shown:




NetworkEditor_Print4

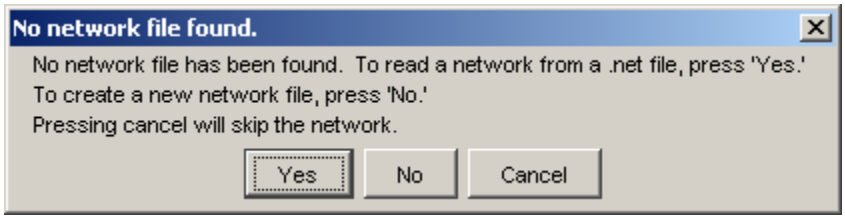
DO NOT change the printer settings. Simply press ***OK*** to finish printing.

## 4.4 Saving a Model Network as an Image

To save the entire network as an image, use the  tool and select an image file. To save the visible network as an image, use the  tool and select an image file.

## 4.5 Adding an Existing Model Network to an Existing Data Set

Older StateMod data sets did not directly reference the general network file (**.net*). When opening older data sets, the StateMod GUI will display the following warning:




Menu_File_Open_Warning

**Warning Message When the Data Set Has No Network File**

If a model network is available, either from the older Makenet software or from StateDMI, press ***Yes*** and select the **.net* file. Saving the response file from the StateMod GUI (see the ***File…Save*** menu) will associate the network file with the data set and the warning will not be displayed the next time that the data set is opened.

## 4.6 Creating a New Model Network for an Existing Data Set

The StateMod GUI does not allow a network to be created for an existing data set because the ***Edit…Add*** and ***Edit…Delete*** menus attempt to keep the various data files synchronized with the network. To create a new network to use with an existing data set, use the StateDMI software. Once the network is created, use the procedure described in **Section 4.4** to attach the network to the data set.

## 4.7 Creating a New Model Network With a New Data Set

If ***File…New*** is selected, a completely new data set will be initialized. At this time, a new network can be started. Using the ***Edit…Add*** menu items will create new model data (see **Section 4.9** below). The positions of the model nodes on the network can be adjusted during this process.

The StateDMI software can also be used to create a network and automate data set creation.

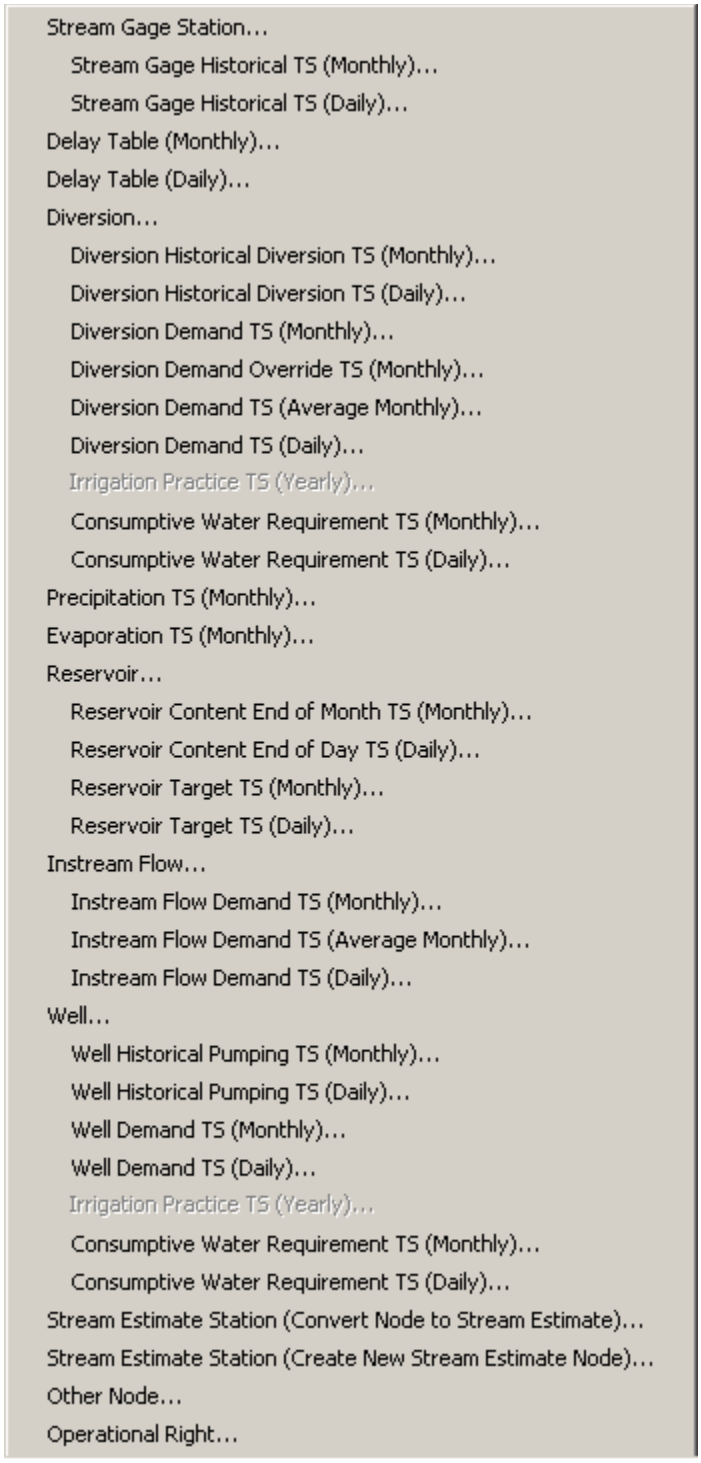
## 4.8 Synchronizing a Model Network with a Data Set

The model network, when used with the StateMod GUI, is primarily a visualization tool and therefore is secondary to the core data files. The features that have been implemented help enforce consistency of the network with other data. If the network becomes inconsistent with other data files, then it is recommended that the StateDMI software be used to edit the network. The network file can then be copied over the previous version in the data set and the Edit menu can once again be used, as described below.

## 4.9 Editing a Model Network – Adding Data

The lists of nodes in the network diagram must remain consistent with the list of stations in the various StateMod files. Therefore, when adding data, the ***Data…Add*** menu should be used. **Important: If the data set is created and maintained using an automated procedure (e.g., StateDMI and TSTool software within CDSS), it is recommended that the command files used to create the set be edited and DMIs be rerun, rather than editing the data set directly.**

The following figure illustrates the data that can be added:




Menu_Data_Add

**Edit…Add Menu**

The above figure illustrates that not all data that can be added are represented in the network and in some cases secondary data components can be added. Secondary components may need to be added because an initial modeling effort used defaults for data but later work requires the addition of more specific data. Therefore, constraining data edits through the menu allows a single entry point for users. When adding data that are represented in the network, a node will automatically be added to the network by

interpolating or extrapolating coordinates. The network interface can then be used to position the node in the diagram, if appropriate. In the future, data may be added from the network interface, but in this case, the intervening dialogs shown below will still be needed to provide important required and initial values for the data.

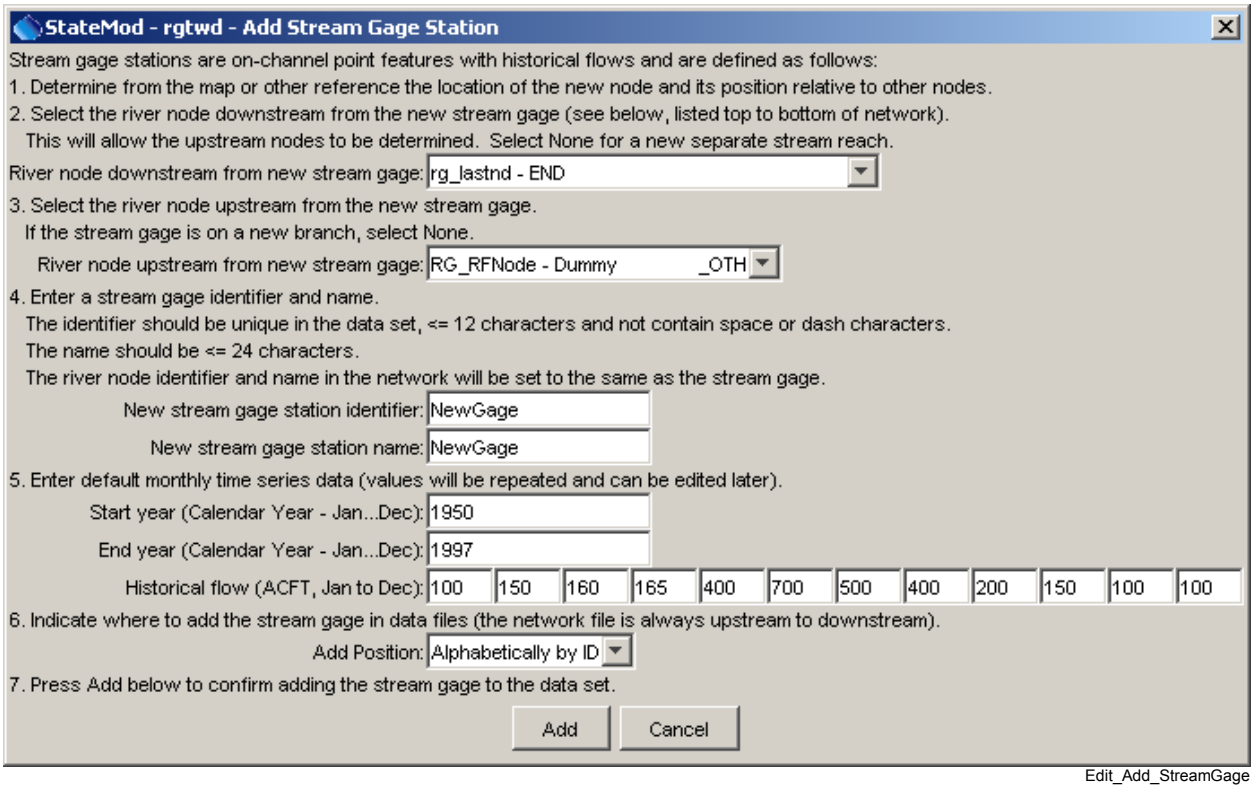
The basic procedure to add data is as follows:

1. Select the appropriate ***Data…Add*** menu item.
2. Fill out the information that is requested (see examples below). Default information can be specified and can be edited in more detail in step 4.
3. Acknowledge the information by pressing the ***Add*** button, resulting in new data being added to one or more data lists in memory.
4. The appropriate data window for the data will be shown and can be used to further edit the information.
5. If the item that was added corresponds to a node in the network, automatically add the node to the network.
6. If appropriate, interactively use the network editor to position the node.

The following sections illustrate features to add data.

### 4.9.1 Adding Stream Gage Data

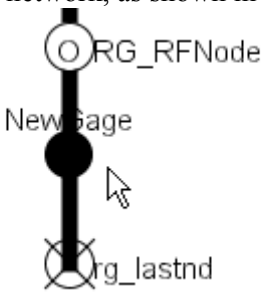
The ***Edit…Add…Stream Gage Station…*** menu adds a new stream gage station:




Edit_Add_StreamGage

**Data for New Stream Gage Station**

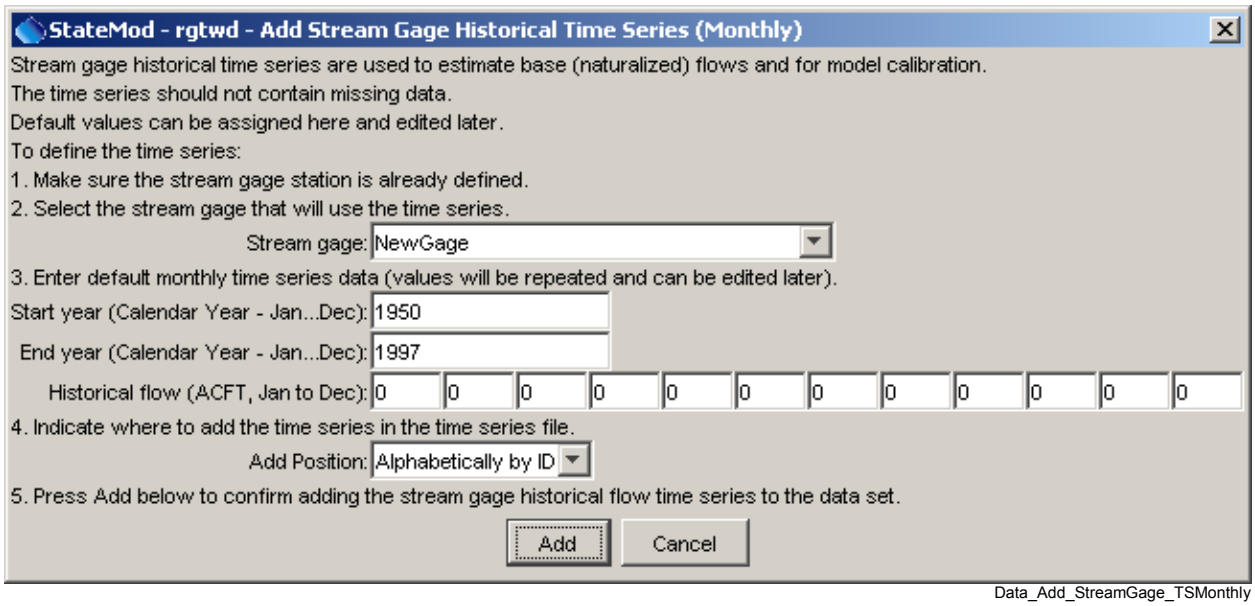
A node will automatically be added to the network, as shown in the following figure:




Edit_Add_StreamGage2

**Stream Gage Added to Network**

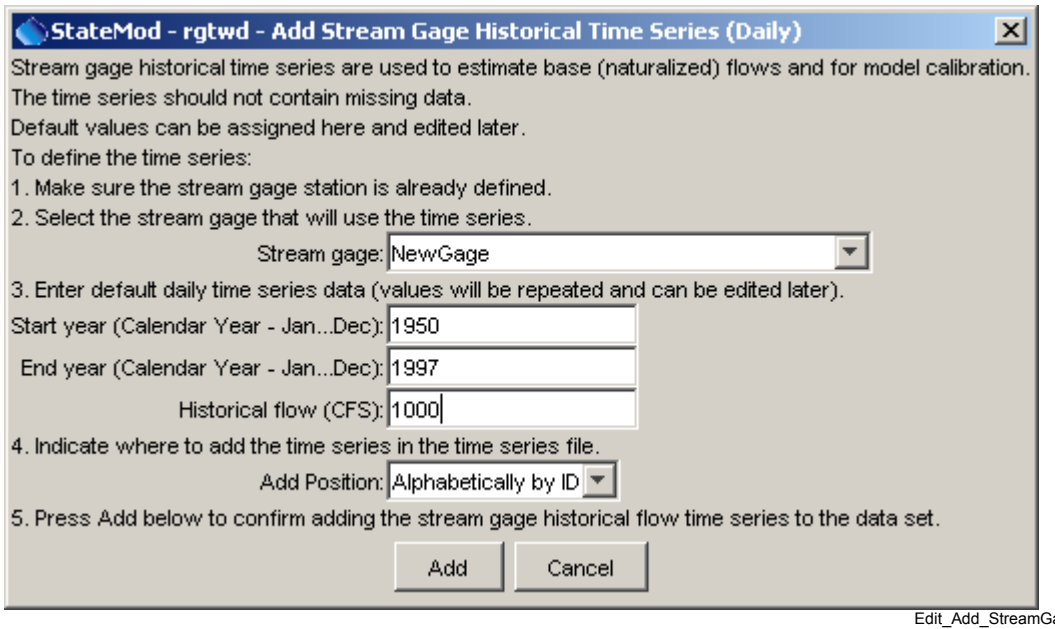
The ***Edit…Add…Stream Gage Historical TS (Monthly)…*** menu adds a new or redefines a monthly historical time series for an existing stream gage station:




Data_Add_StreamGage_TSMonthly

**New Stream Gage Historical Time Series (Monthly)**

The ***Edit…Add…Stream Gage Historical TS (Daily)…*** menu adds a new or redefines a daily historical time series for an existing stream gage station:




Edit_Add_StreamGage_TSDaily

**New Stream Gage Historical Time Series (Daily)**

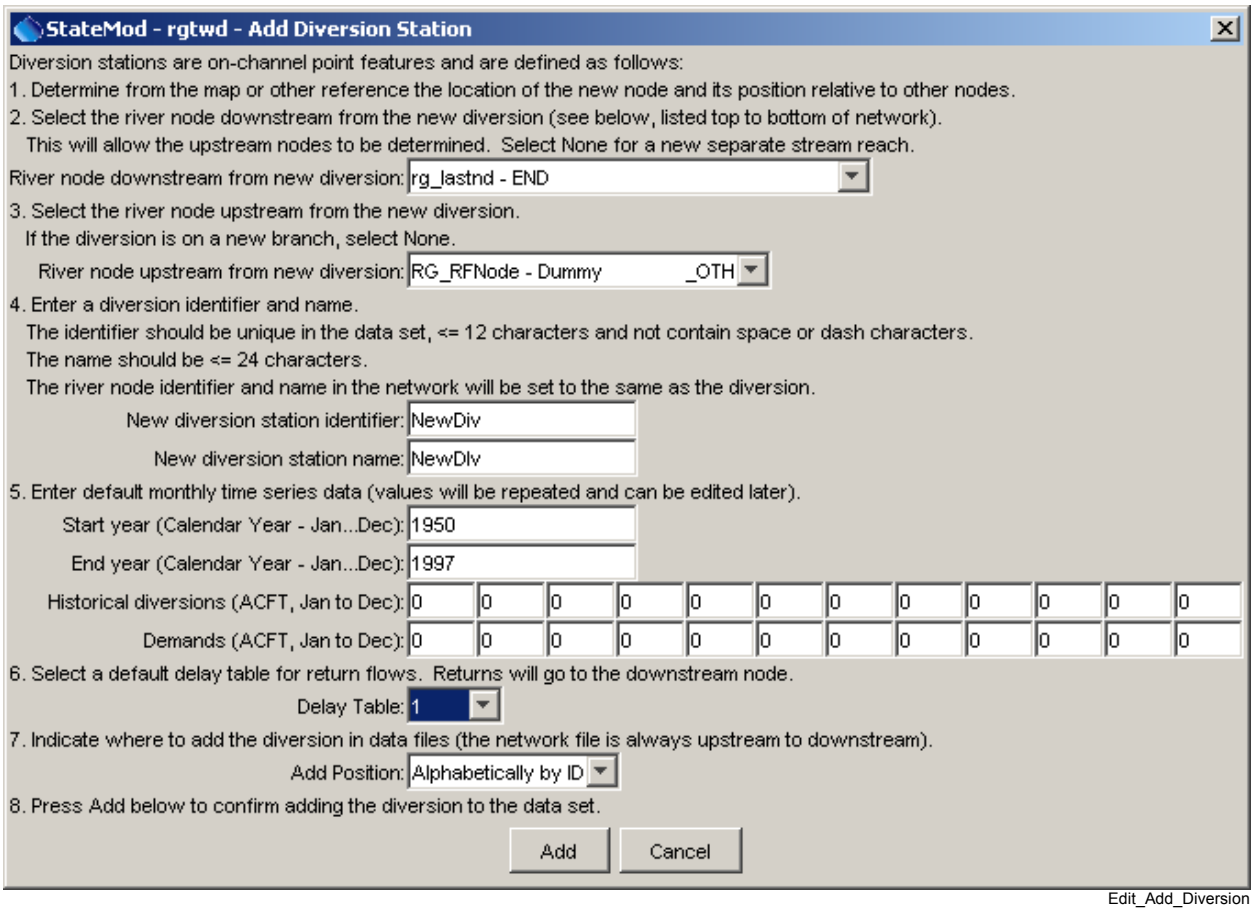
### 4.9.2 Adding Delay Table Data

The ***Edit…Add…Delay Table (Monthly)…*** menu adds a new monthly delay table:

The ***Edit…Add…Delay Table (Daily)…*** menu adds a new daily delay table:

### 4.9.3 Adding Diversion Data

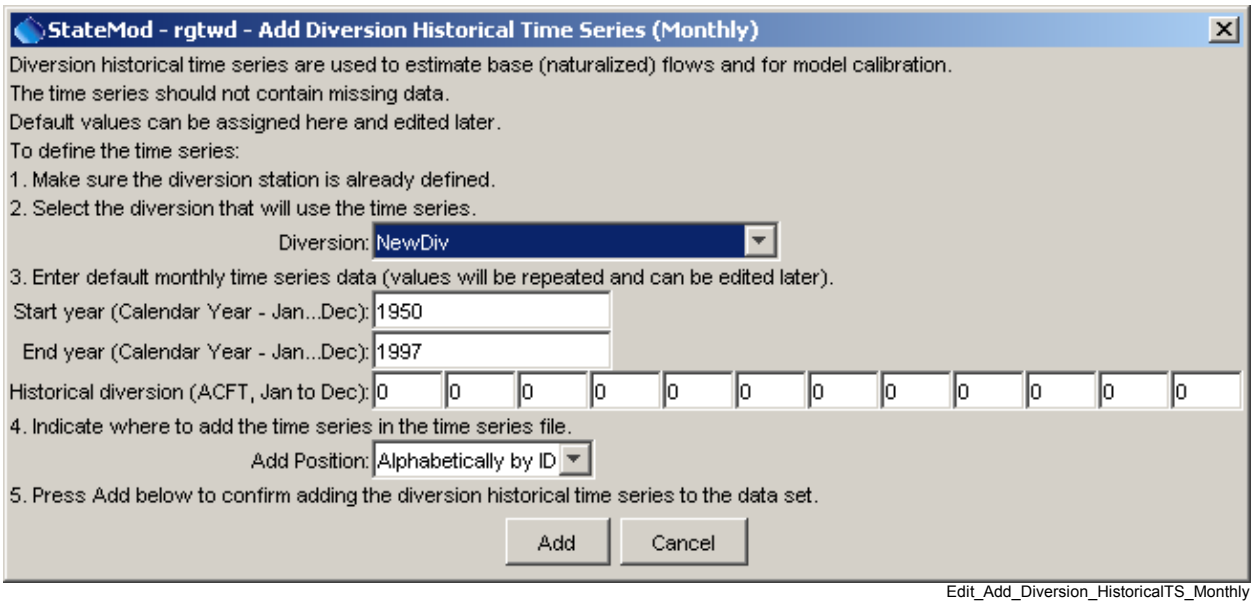
The ***Edit…Add…Diversion…*** menu adds a new diversion station:




Edit_Add_Diversion

**New Diversion Station**

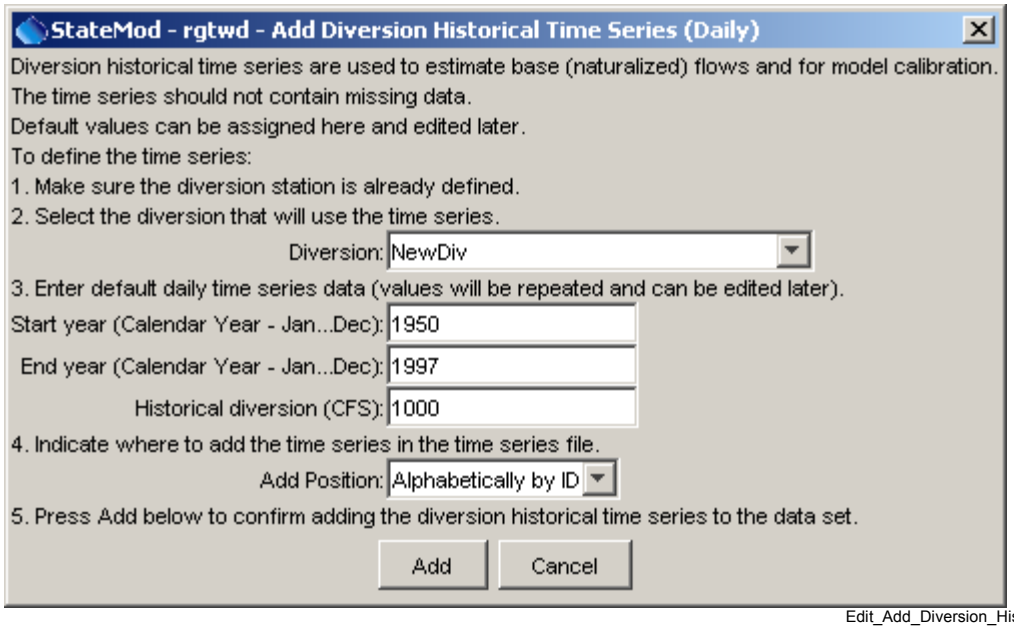
The ***Edit…Add…Diversion Historical Diversion TS (Monthly)…*** menu adds a new or redefines a diversion historical time series (monthly) for an existing diversion station:




Edit_Add_Diversion_HistoricalTS_Monthly

**New Diversion Historical Time Series (Monthly)**

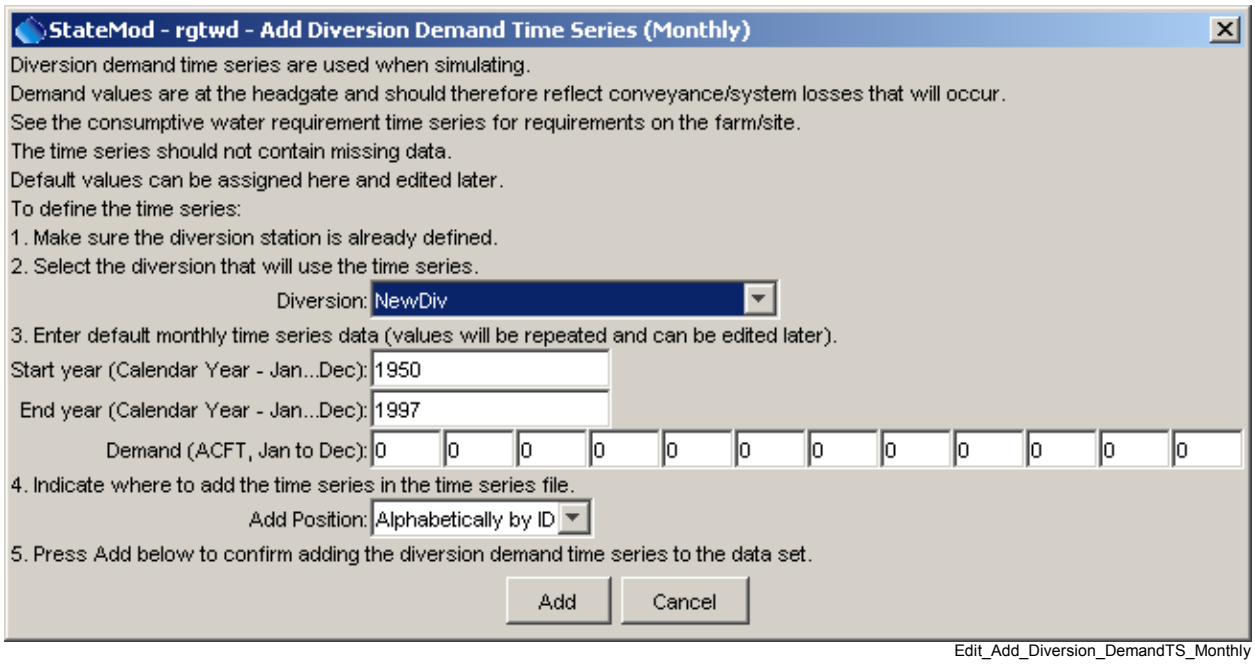
The ***Edit…Add…Diversion Historical Diversion TS (Daily)…*** menu adds a new or redefines a diversion historical time series (daily) for an existing diversion station:




Edit_Add_Diversion_HistoricalTS_Daily

**New Diversion Historical Time Series (Daily)**

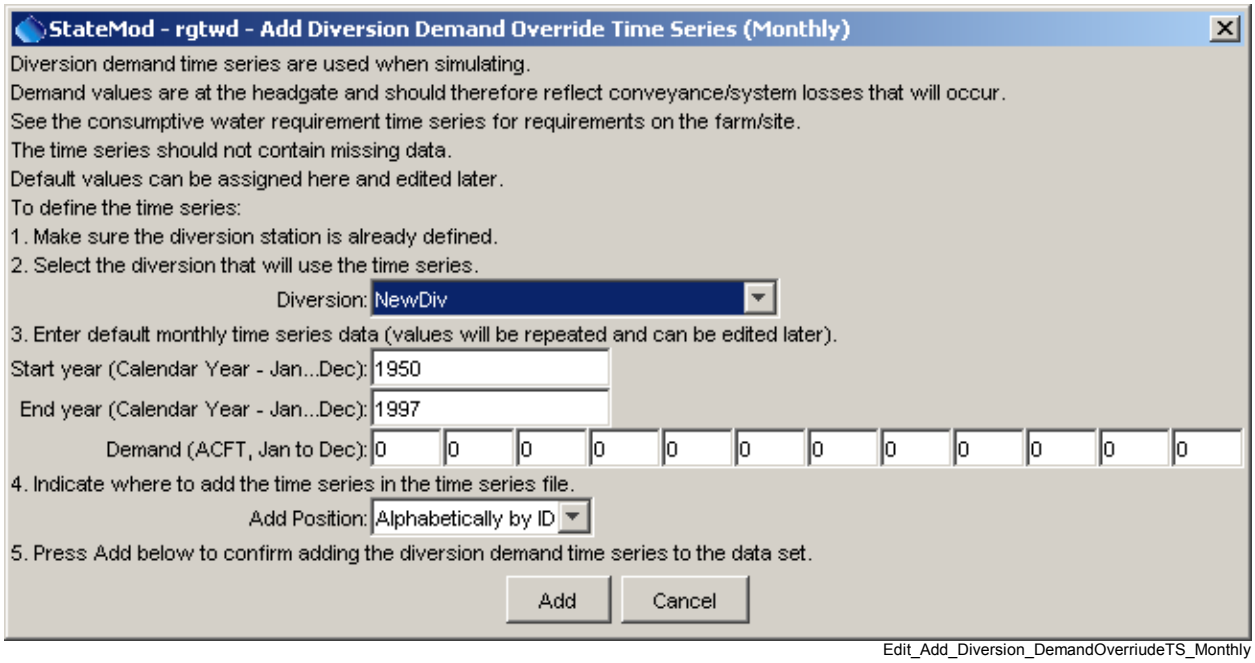
The ***Edit…Add…Diversion Demand TS (Monthly)…*** menu adds a new or redefines a diversion demand time series (monthly) for an existing diversion station:




Edit_Add_Diversion_DemandTS_Monthly

**New Diversion Demand Time Series (Monthly)**

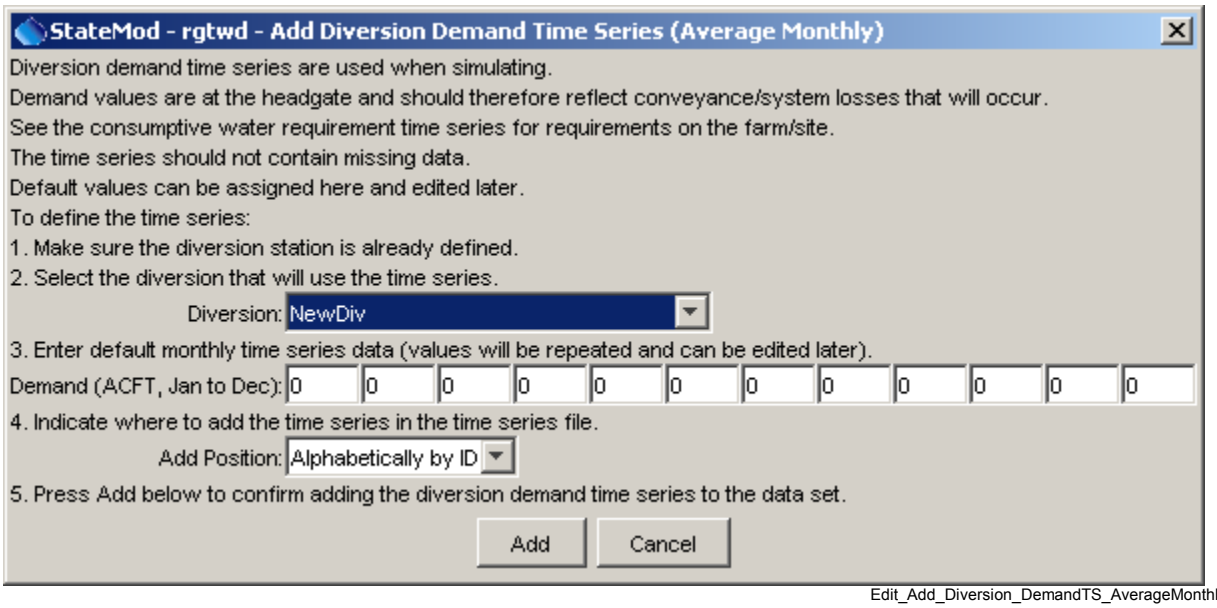
The ***Edit…Add…Diversion Demand Override TS (Monthly)…*** menu adds a new or redefines a diversion demand override time series (monthly) for an existing diversion station:




Edit_Add_Diversion_DemandOverriudeTS_Monthly

**New Diversion Demand Override Time Series (Monthly)**

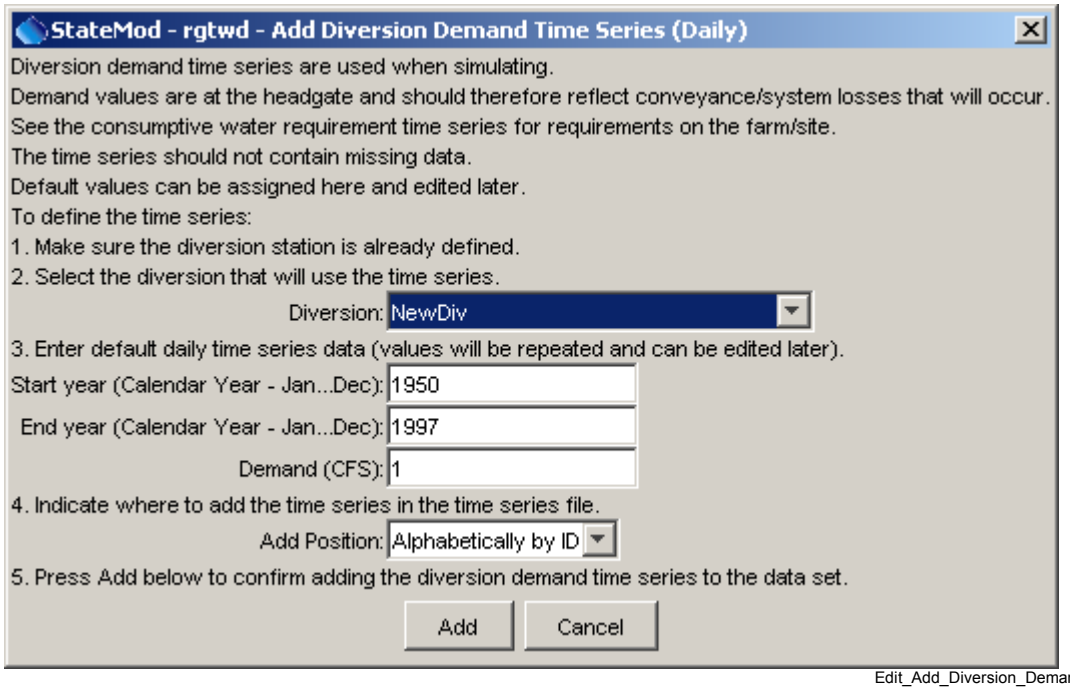
The ***Edit…Add…Diversion Demand TS (Average Monthly)…*** menu adds a new or redefines a diversion demand time series (average monthly) for an existing diversion station:




Edit_Add_Diversion_DemandTS_AverageMonthly

**New Diversion Demand Override Time Series (Average Monthly)**

The ***Edit…Add…Diversion Demand TS (Daily)…*** menu adds a new or redefines a diversion demand time series (daily) for an existing diversion station:

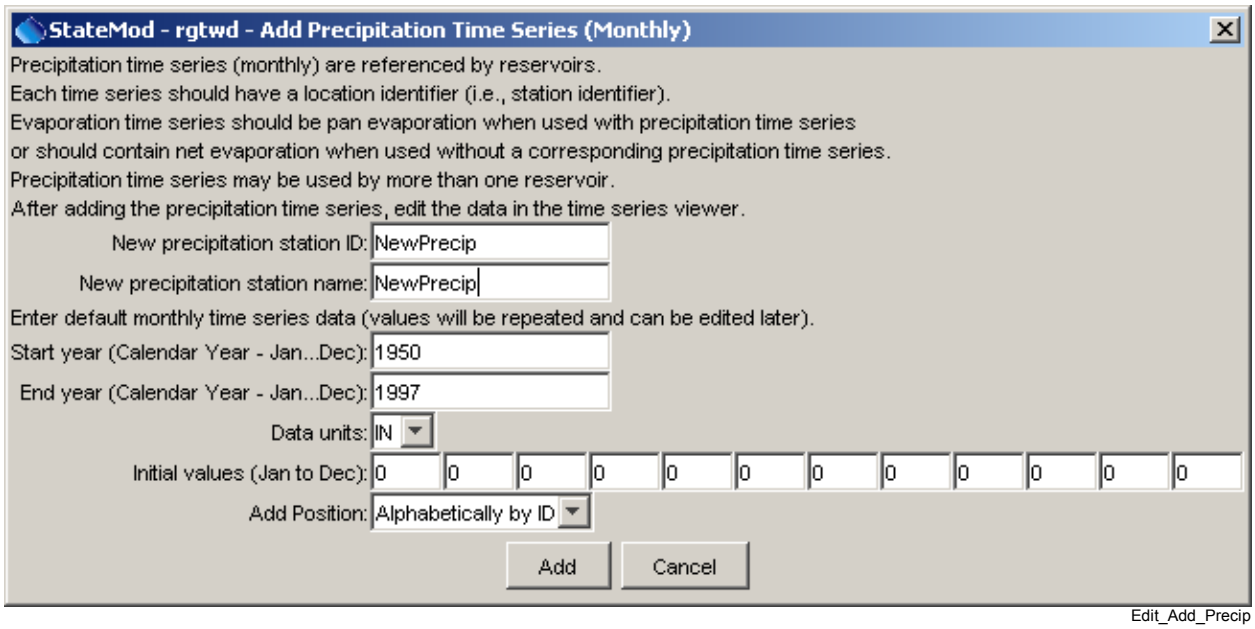



Edit_Add_Diversion_DemandTS_Dailly

**New Diversion Demand Time Series (Daily)**

### 4.9.4 Adding Precipitation Data

The ***Edit…Add…Precipitation TS (Monthly)…*** menu adds a new monthly precipitation time series:

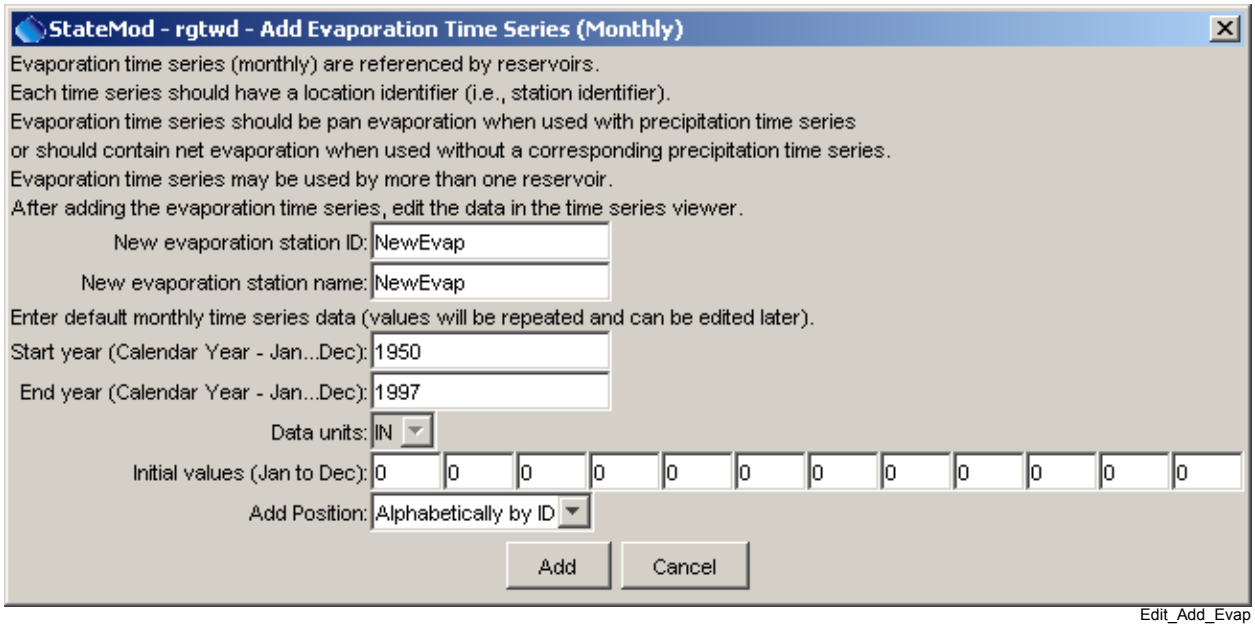



Edit_Add_Precip

**New Precipitation Time Series**

### 4.9.5 Adding Evaporation Data

The ***Edit…Add…Evaporation TS (Monthly)…*** menu adds a new monthly evaporation time series:

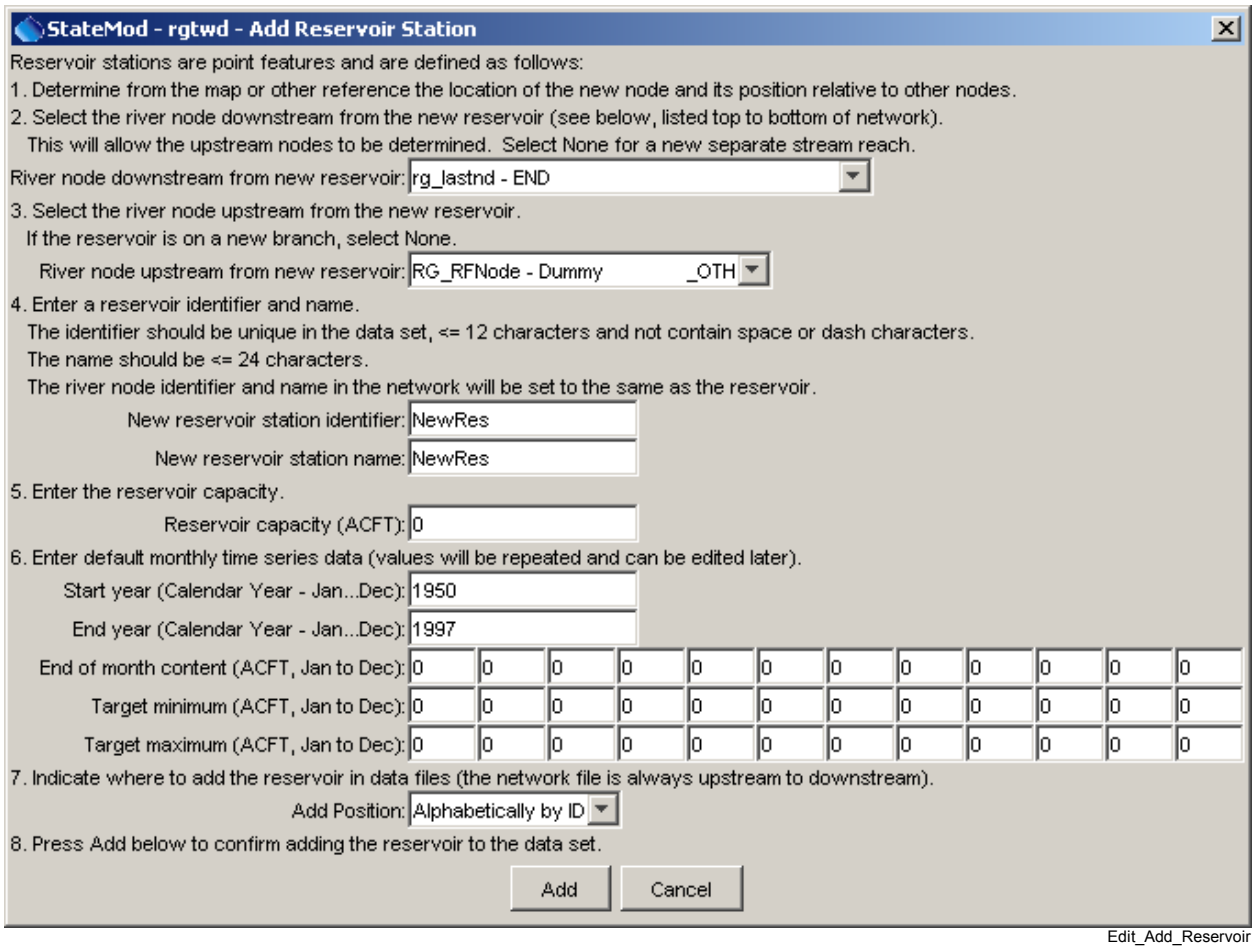



Edit_Add_Evap

**New Evaporation Time Series**

### 4.9.6 Adding Reservoir Data

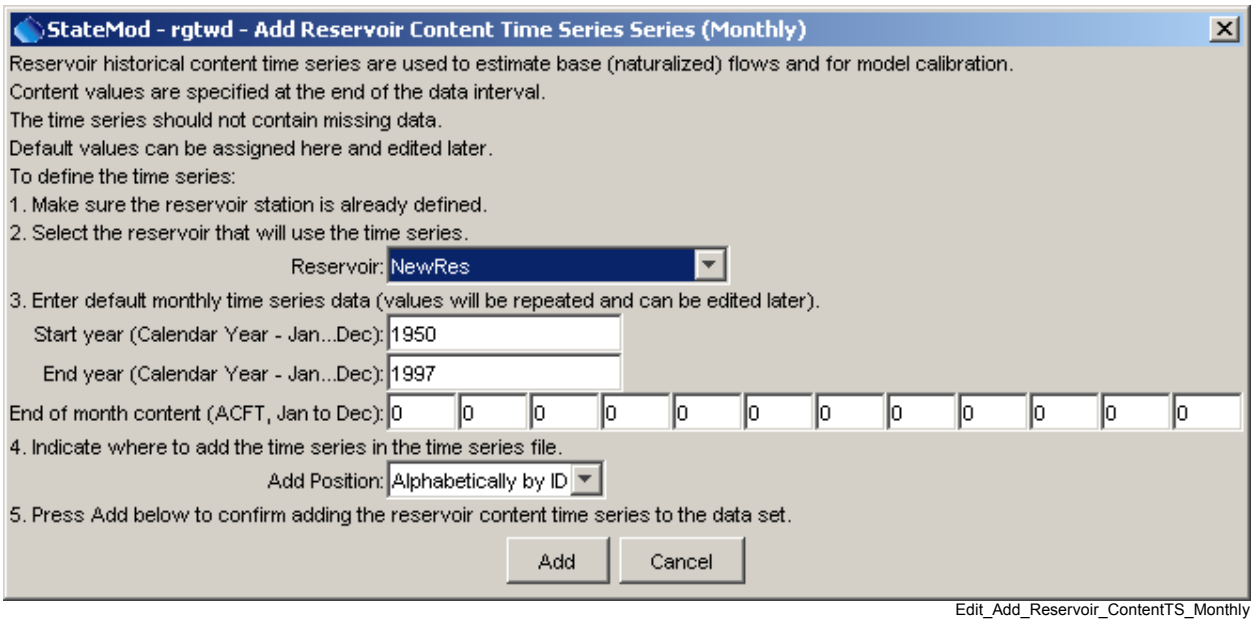
The ***Edit…Add…Reservoir…*** menu adds a new reservoir station:




Edit_Add_Reservoir

**New Reservoir Station**

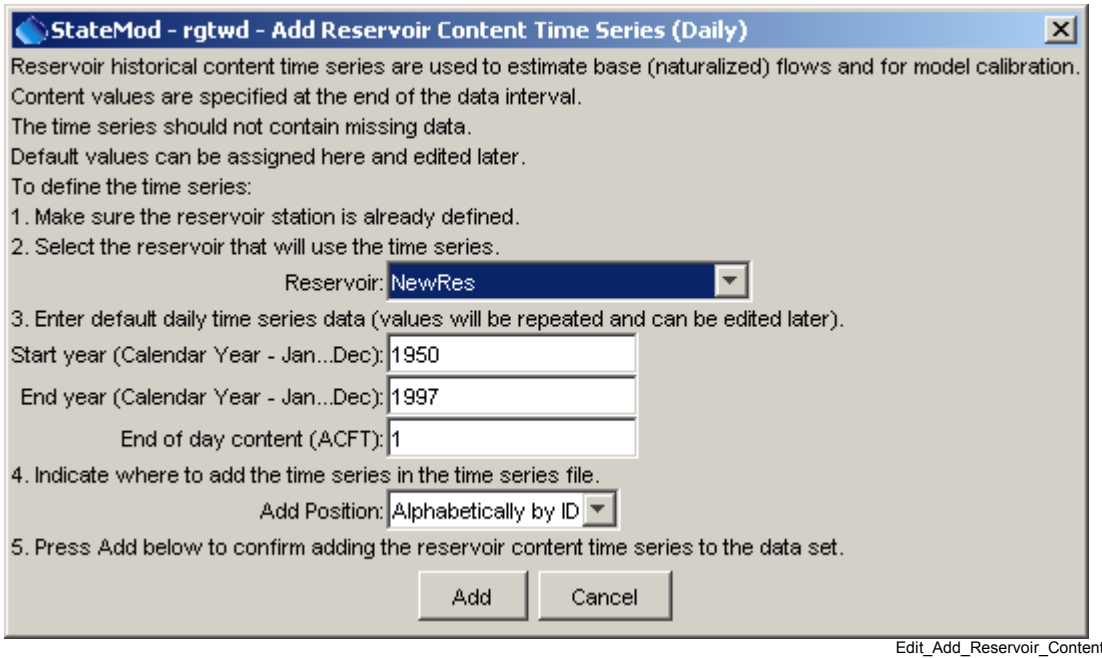
The ***Edit…Add…Reservoir Content End of Month TS (Monthly)…*** menu adds a new or redefines a reservoir end of month content time series (monthly) for an existing reservoir station:




Edit_Add_Reservoir_ContentTS_Monthly

**New Reservoir Content Time Series (Monthly)**

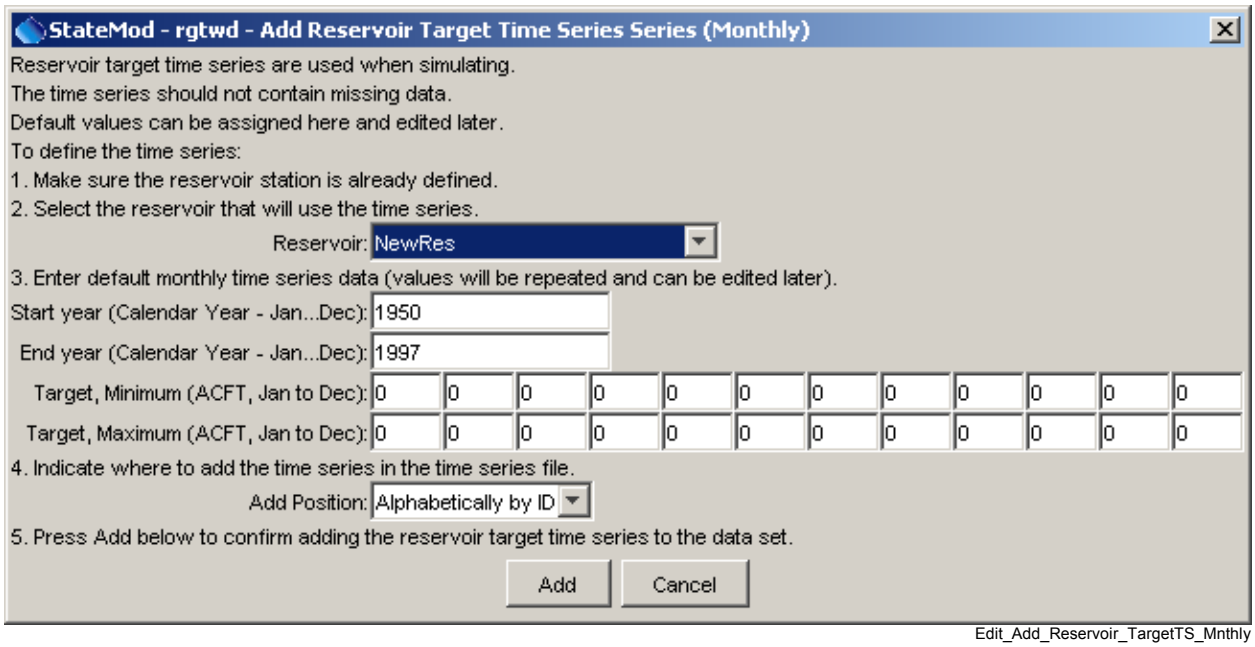
The ***Edit…Add…Reservoir Content End of Day (Daily)…*** menu adds a new or redefines a reservoir end of day content time series (daily) for an existing reservoir station:




Edit_Add_Reservoir_ContentTS_Daily

**New Reservoir Content Time Series (Daily)**

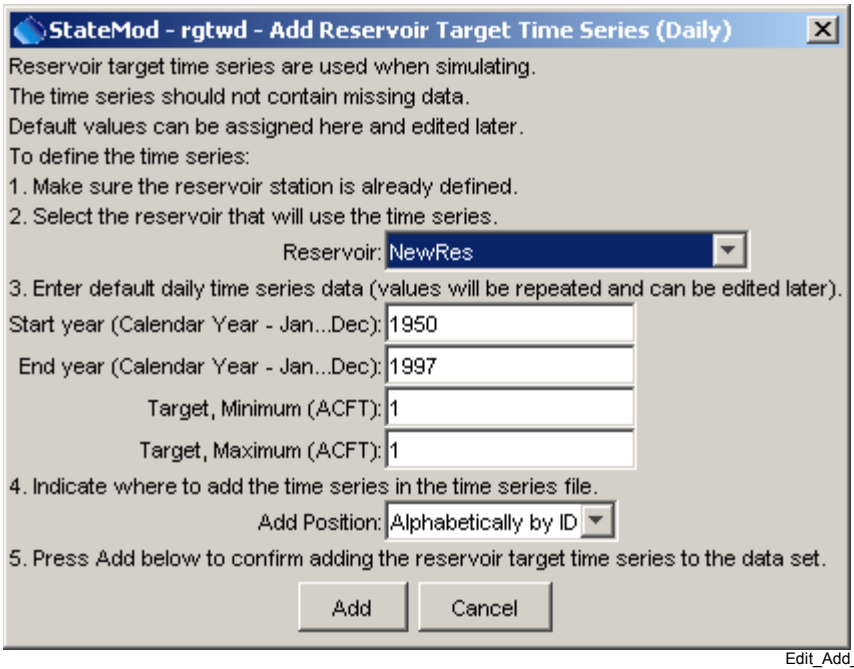
The ***Edit…Add…Target TS (Monthly)…*** menu adds a new or redefines a reservoir target time series (monthly) for an existing reservoir station:




Edit_Add_Reservoir_TargetTS_Mnthly

**New Reservoir Target Time Series (Monthly)**

The ***Edit…Add…Target TS (Daily)…*** menu adds a new or redefines a reservoir target time series (daily) for an existing reservoir station:

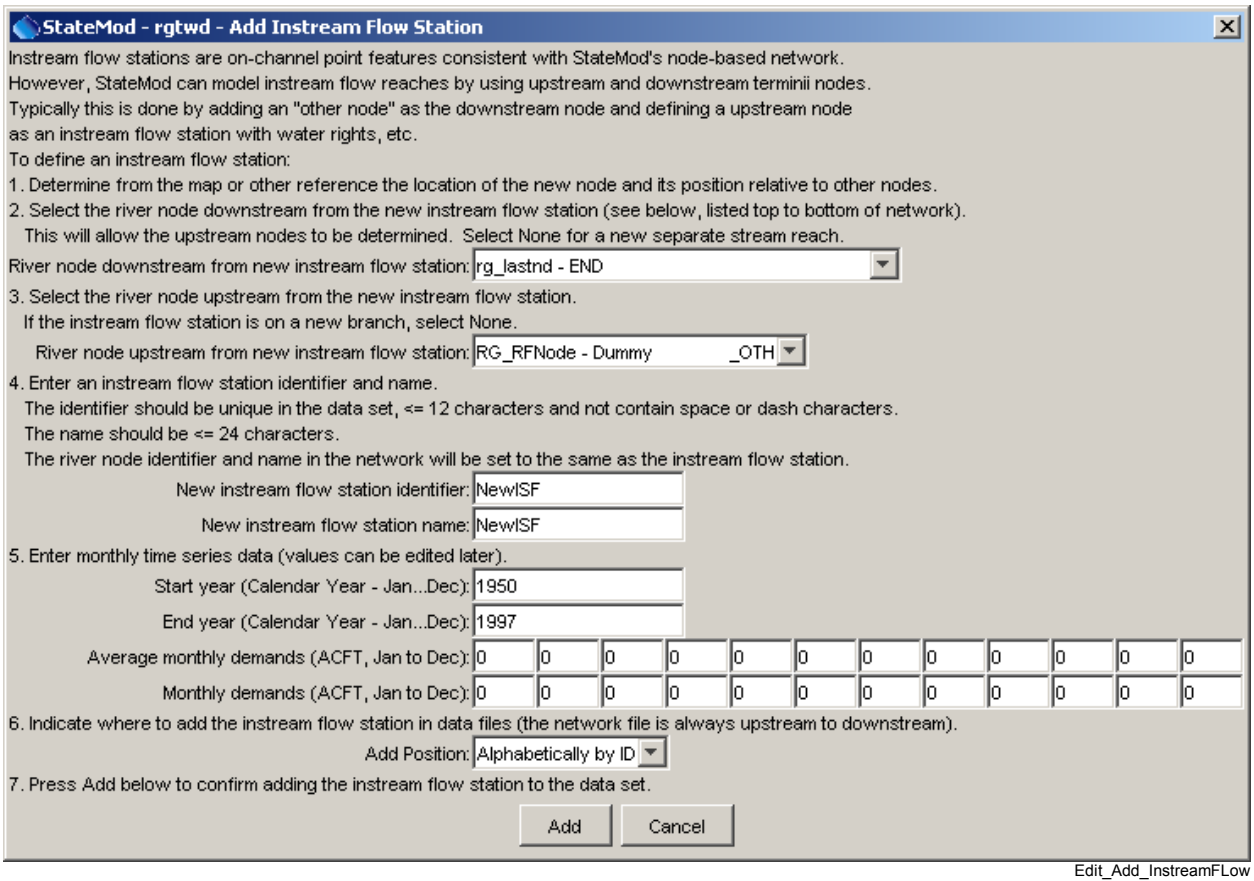



Edit_Add_Reservoir_TargetTS_Daily

**New Reservoir Target Time Series (Daily)**

### 4.9.7 Adding Instream Flow Data

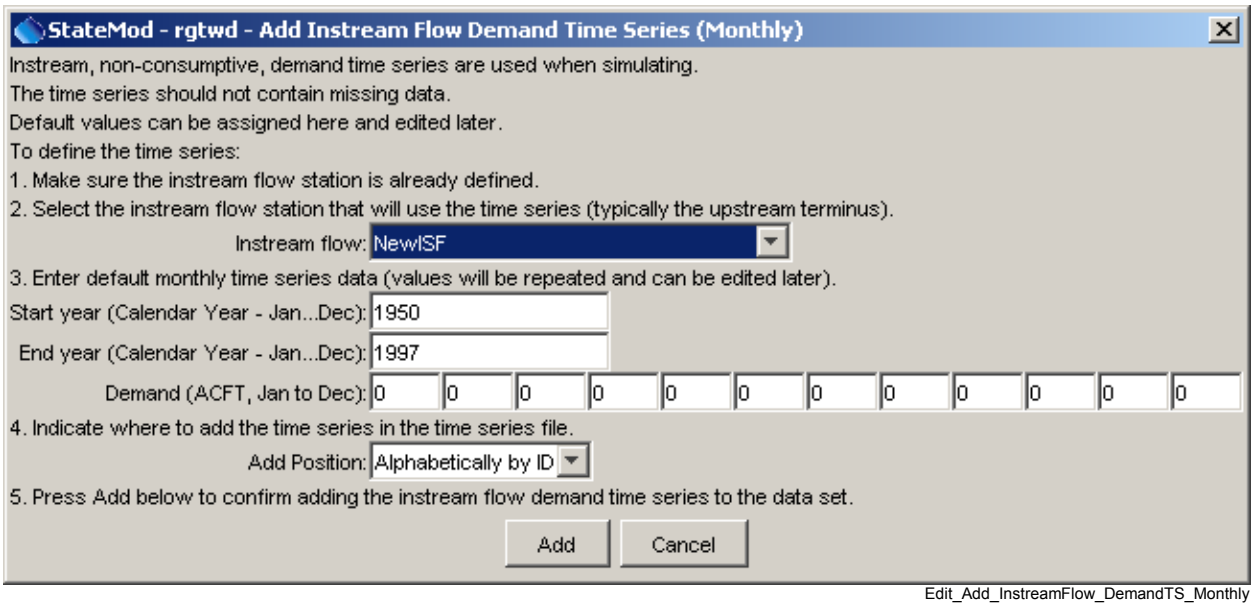
The ***Edit…Add…Instream Flow…*** menu adds a new instream flow station:




Edit_Add_InstreamFLow

**New Instream Flow Station**

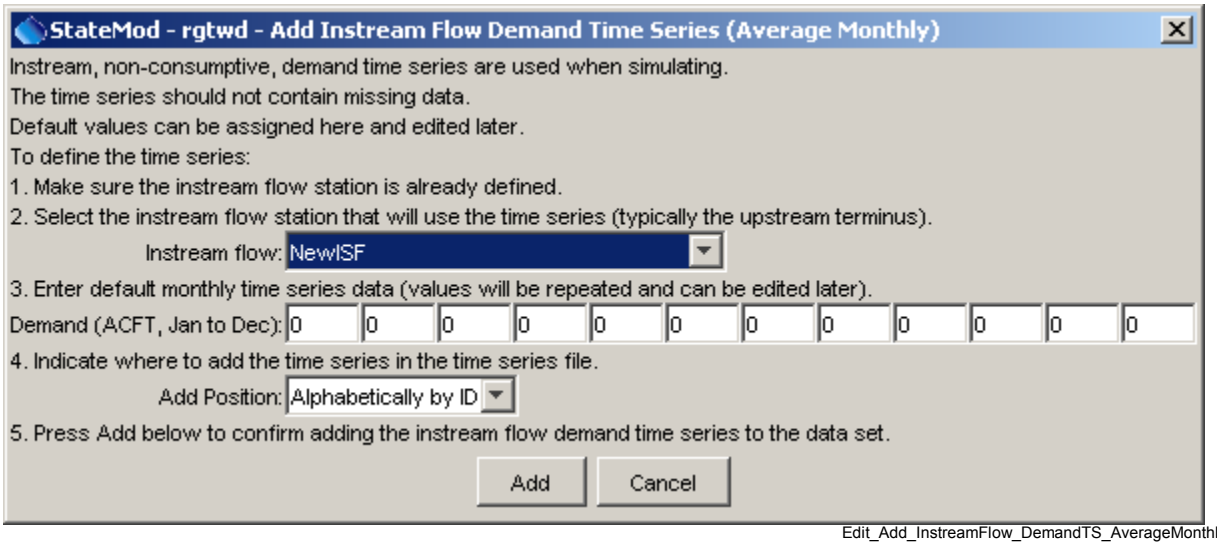
The ***Edit…Add…Instream Flow Demand TS (Monthly)…*** menu adds a new or redefines an instream flow demand time series (monthly) for an existing instream flow station:




Edit_Add_InstreamFlow_DemandTS_Monthly

**New Instream Flow Demand Time Series (Monthly)**

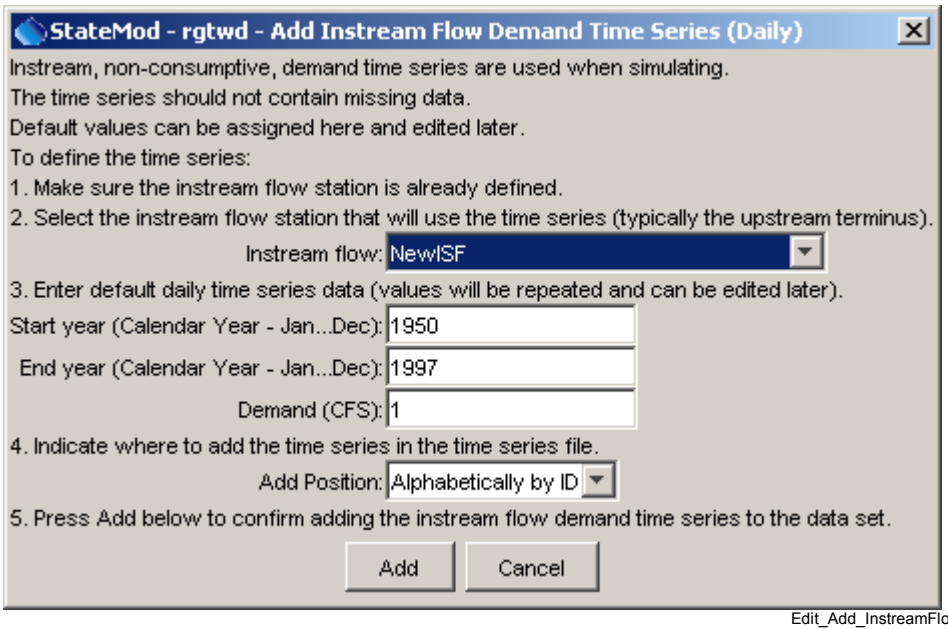
The ***Edit…Add…Instream Flow Demand TS (Average Monthly)…*** menu adds a new or redefines an instream flow demand time series (average monthly) for an existing instream flow station:




Edit_Add_InstreamFlow_DemandTS_AverageMonthly

**New Instream Flow Demand Time Series (Average Monthly)**

The ***Edit…Add…Instream Flow Demand TS (Daily)…*** menu adds a new or redefines an instream flow demand time series (daily) for an existing instream flow station:

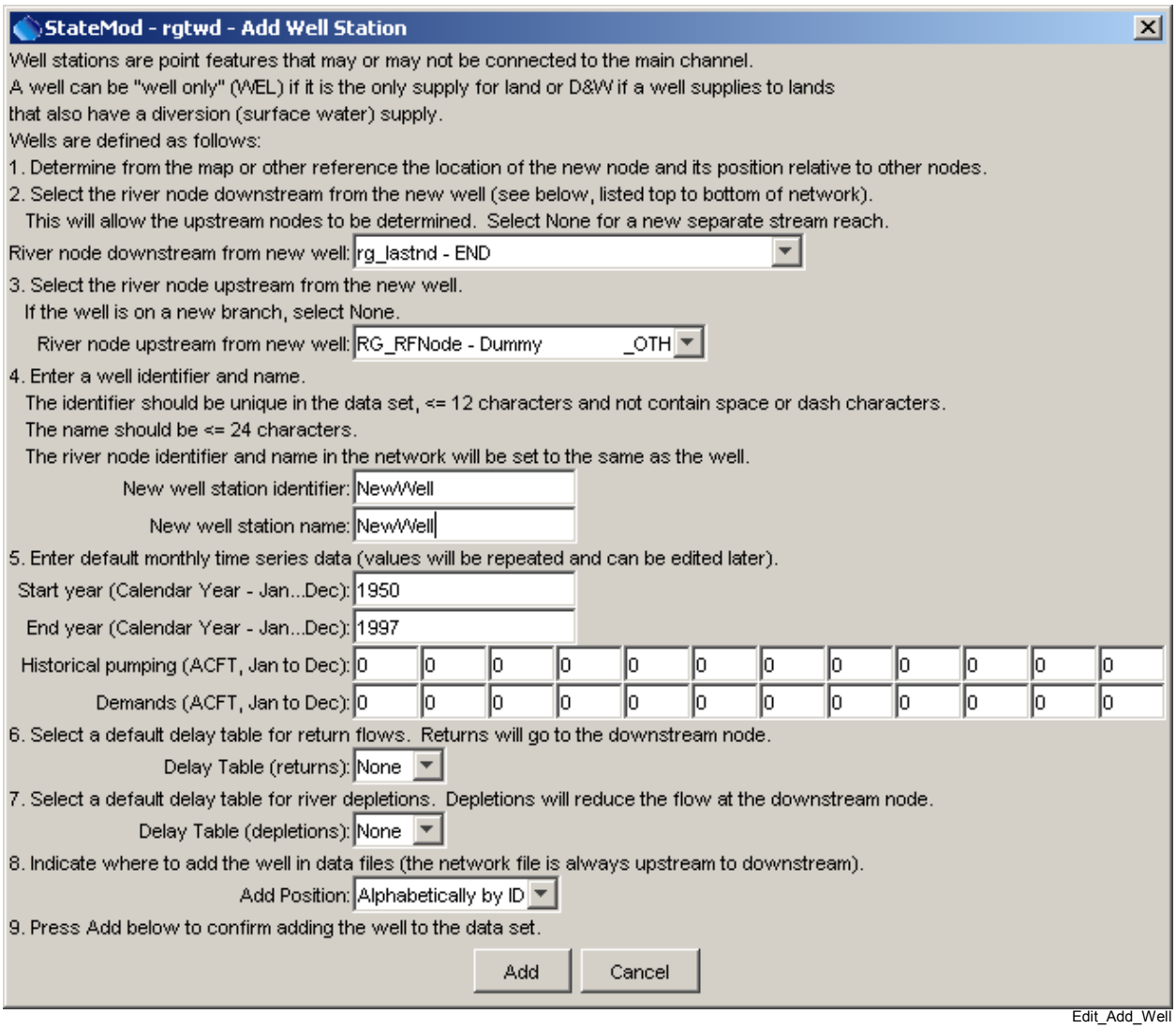



Edit_Add_InstreamFlow_DemandTS_Daily

**New Instream Flow Demand Time Series (Daily)**

### 4.9.8 Adding Well Data

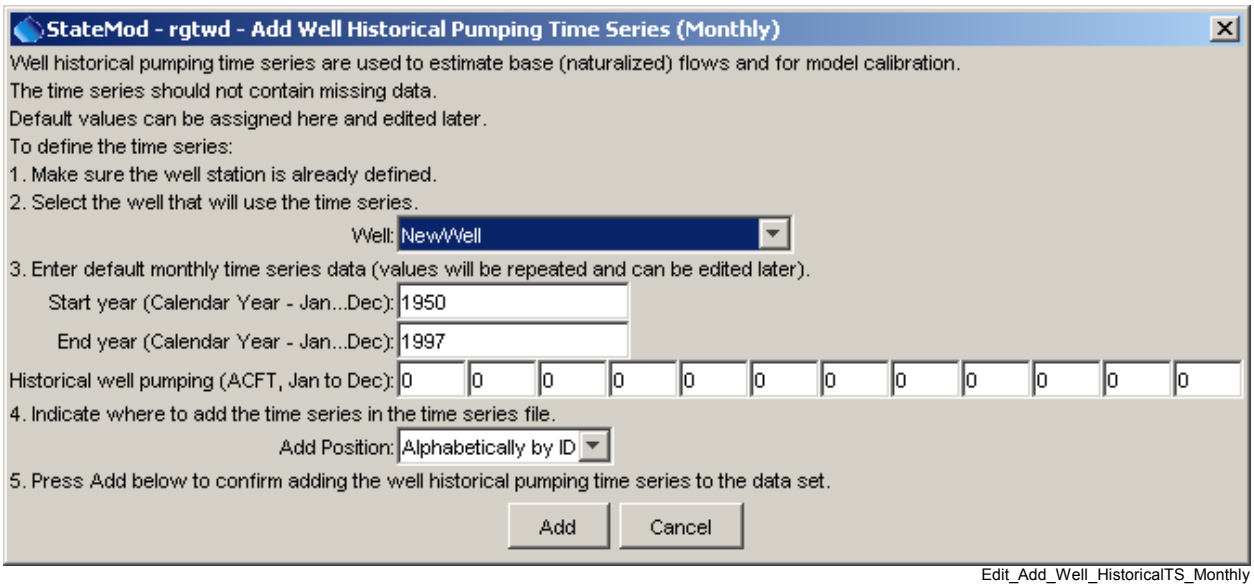
The ***Edit…Add…Well…*** menu adds a new well station:




Edit_Add_Well

**New Well Station**

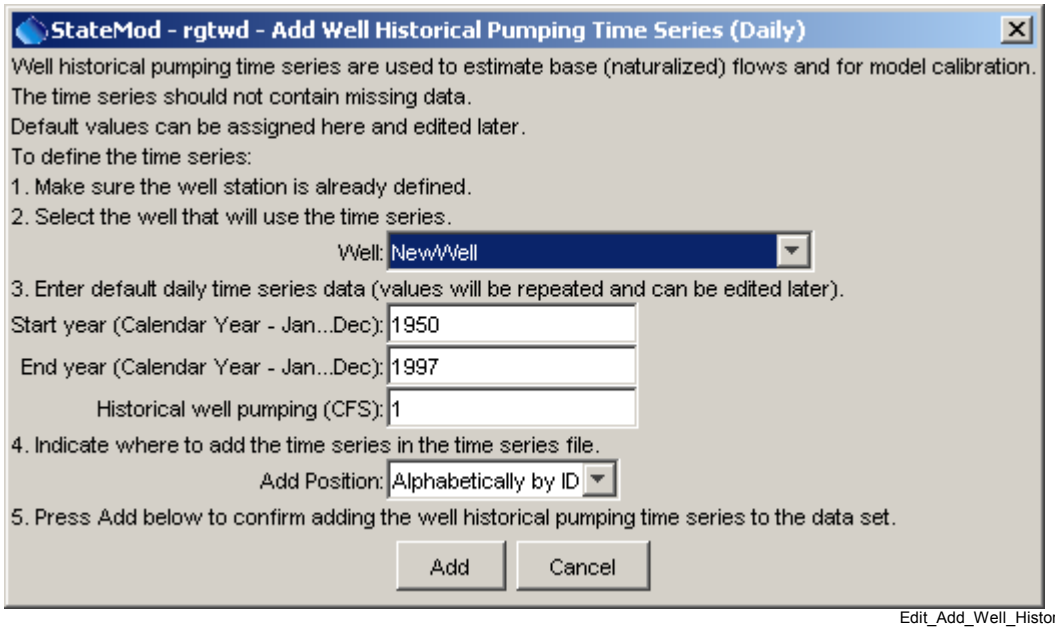
The ***Edit…Add…Well Historical Pumping TS (Monthly)…*** menu adds a new or redefines a well historical pumping time series (monthly) for an existing well station:




Edit_Add_Well_HistoricalTS_Monthly

**New Well Historical Pumping Time Series (Monthly)**

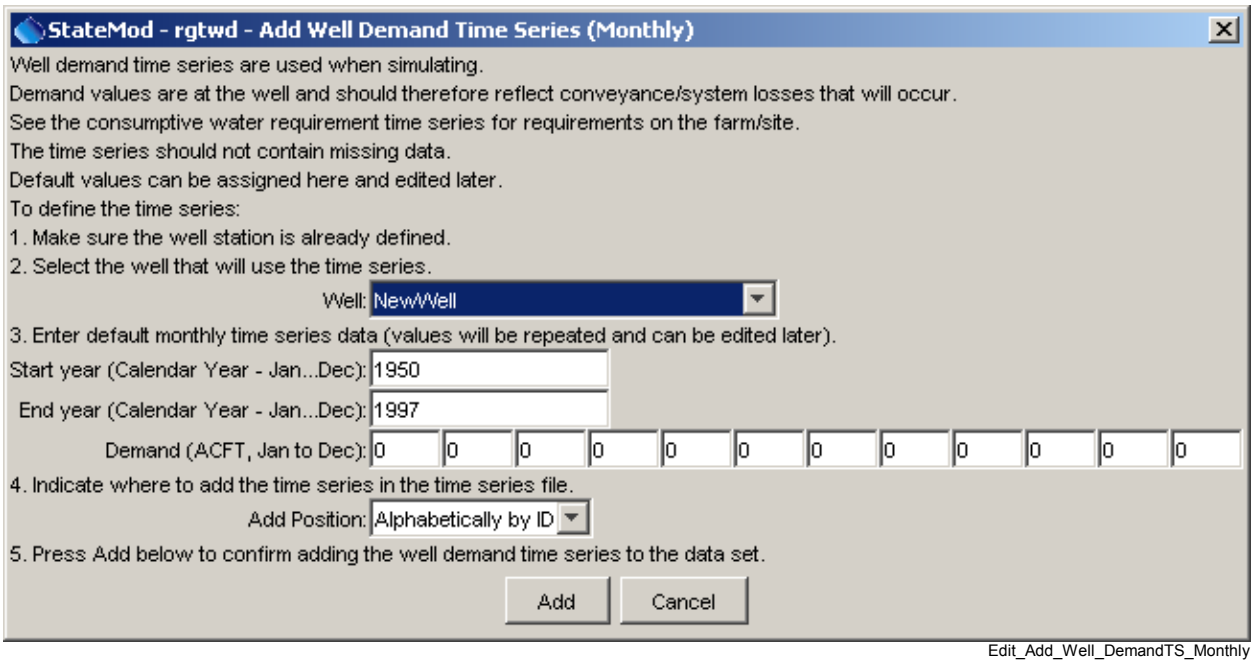
The ***Edit…Add…Well Historical Pumping TS (Daily)…*** menu adds a new or redefines a well historical time series (daily) for an existing well station:




Edit_Add_Well_HistoricalTS_Daily

**New Well Historical Pumping Time Series (Daily)**

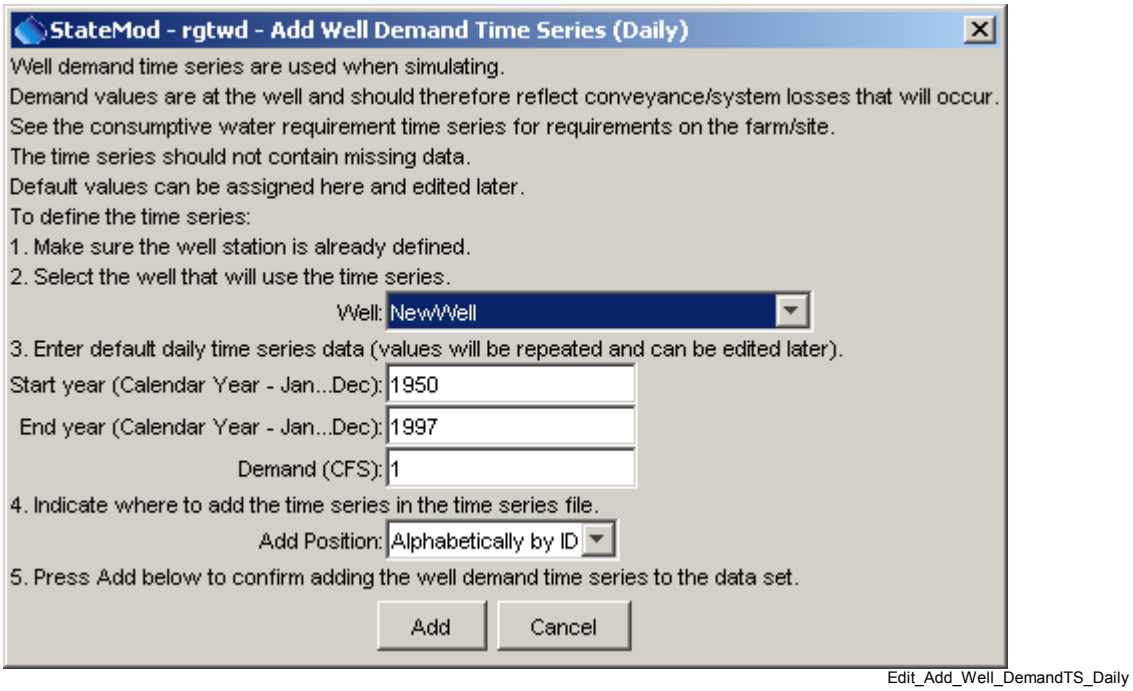
The ***Edit…Add…Well Demand TS (Monthly)…*** menu adds a new or redefines a well demand time series (monthly) for an existing well station:




Edit_Add_Well_DemandTS_Monthly

**New Well Historical Demand Time Series (Monthly)**

The ***Edit…Add…Well Demand TS (Daily)…*** menu adds a new or redefines a well demand time series (daily) for an existing well station:




Edit_Add_Well_DemandTS_Daily

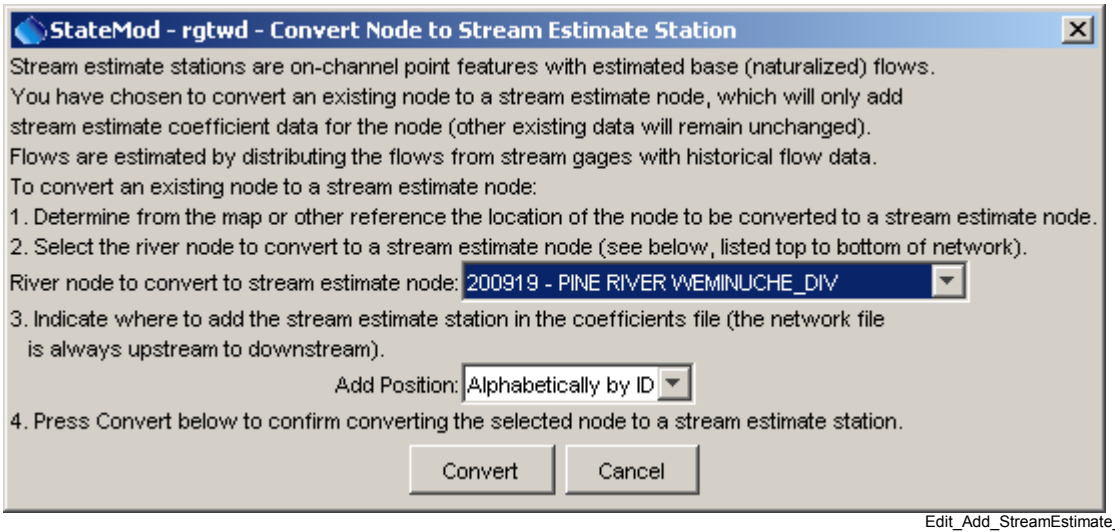
**New Well Historical Demand Time Series (Daily)**

The ***Edit…Add…Well Consumptive Water Requirement TS (Monthly)…*** menu adds a new or redefines a well consumptive water requirement time series (monthly) for an existing well station.

The ***Edit…Add…Well Consumptive Water Requirement TS (Daily)…*** menu adds a new or redefines a well consumptive water requirement time series (daily) for an existing well station.

### 4.9.9 Adding Stream Estimate Data

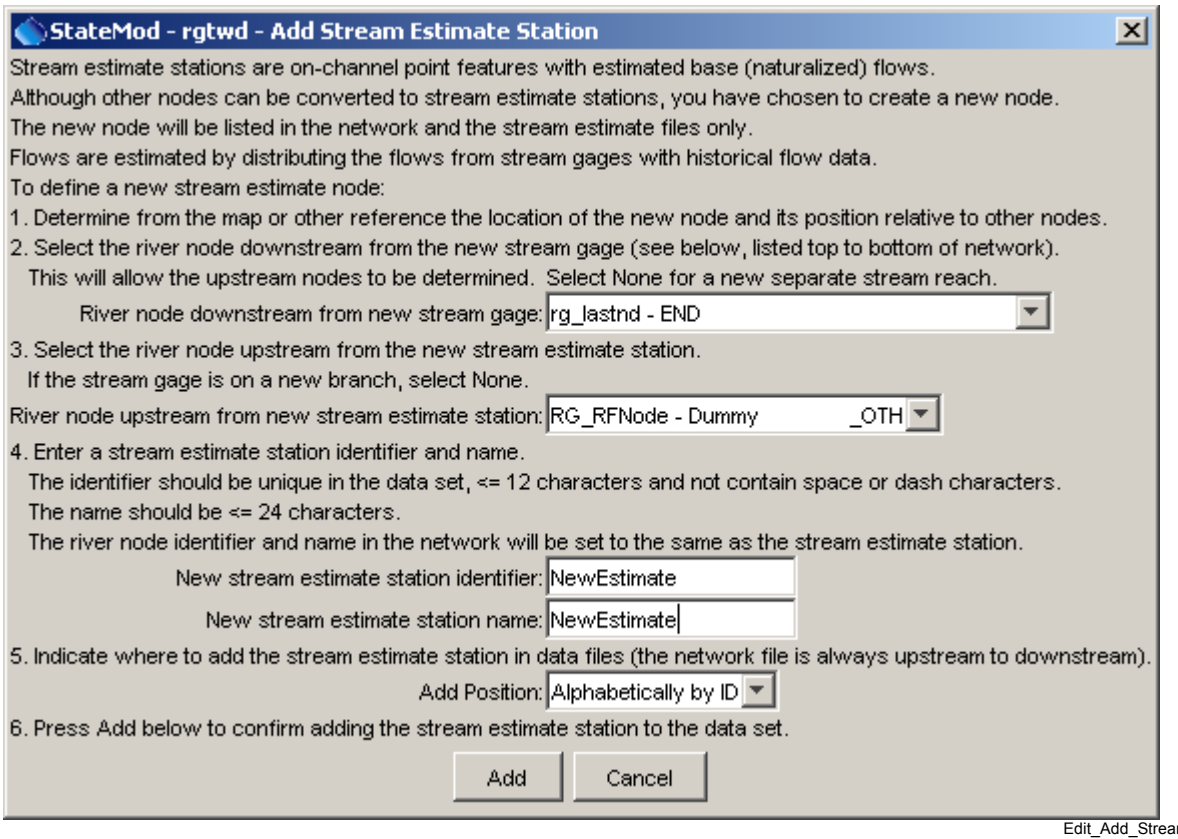
The ***Edit…Add…Stream Estimate Station (Convert Node to Stream Estimate)…*** menu converts an existing station to a stream estimate station:




Edit_Add_StreamEstimate_Convert

**Converting a Station to a Stream Estimate Station**

The ***Edit…Add…Stream Estimate Station (Create New Stream Estimate Node)…*** menu creates a new stream estimate station:

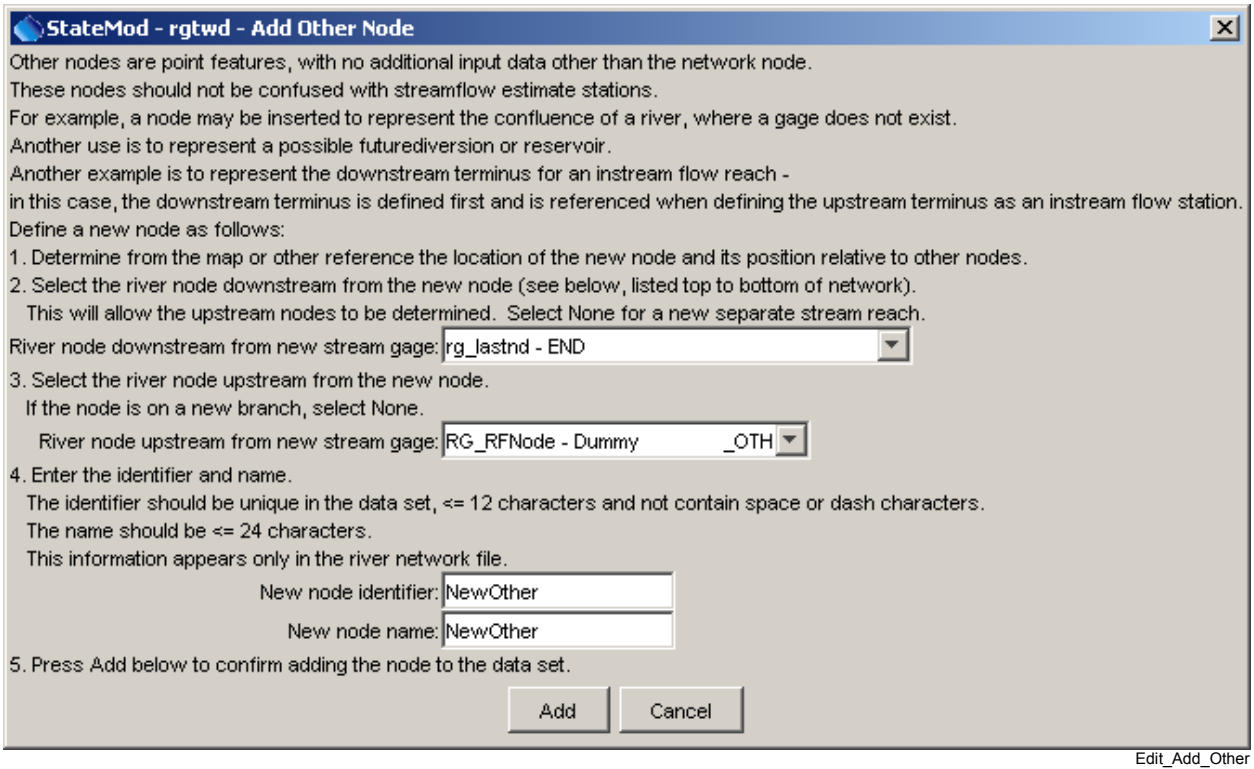



Edit_Add_StreamEstimate

**New Stream Estimate Station**

### 4.9.10 Adding Other Data

The ***Edit…Add…Other Node…*** menu creates a new node in the river network file that is not a station in any other file:

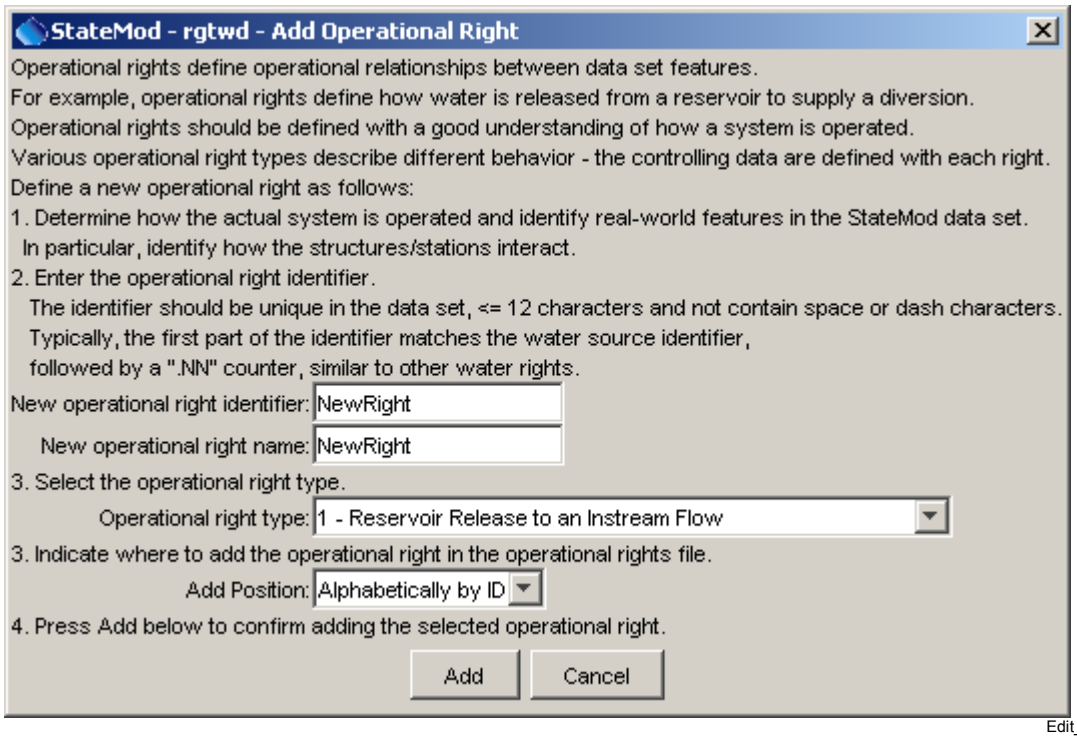



Edit_Add_Other

**New Other (River Node) Station**

### 4.9.11 Adding Operational Right Data

The ***Edit…Add…Operational Right…*** menu creates a new operational right:

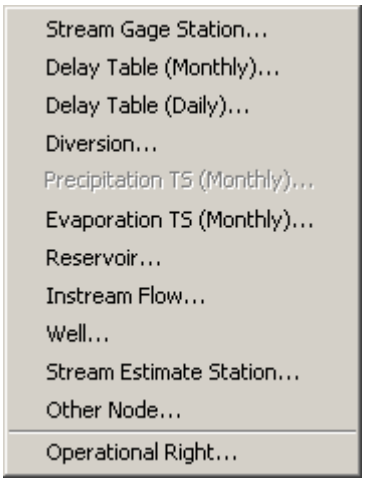



Edit_Add_Other

**New Operational Right**

## 4.10 Editing a Model Network – Deleting Data

Similar to adding data to the data set, deleting data requires using the ***Edit…Delete*** menu to maintain consistency between the model network and data components:




Menu_Edit_Delete

**Edit…Delete Menu**

Features that exist in the network will be deleted, and the corresponding upstream/downstream nodes will be reconnected as appropriate. After deleting the node, the data files should be saved and the StateMod data check should be run. The StateMod GUI will not reconnect return flows, operational rights, or baseflow nodes. Edit windows must be used to adjust these data. Any problems associated with the deletion will be described in the StateMod log file.

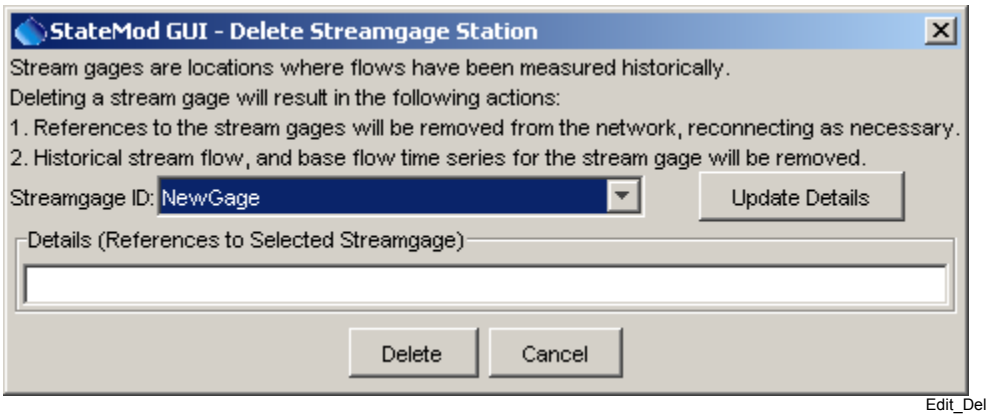
The basic procedure to delete data is as follows:

1. Select the appropriate ***Data…Delete*** menu item.
2. Select the data item to delete. The ***Update Details*** button can be pressed to show information about the data item – this feature is under development and is envisioned to indicate related data that could be impacted by the delete.
3. Confirm the delete by pressing the ***Delete*** button, resulting in data being deleted from one or more data lists in memory. If a station is being deleted, the related time series will also be deleted.
4. If the item that was added corresponds to a node in the network, automatically remove the node from the network and reconnect effected nodes.
5. If appropriate, interactively use the network editor to position the node.

The following sections illustrate features to delete data.

### 4.10.1 Deleting a Stream Gage Data

The ***Edit…Delete…Stream Gage…*** menu deletes a stream gage:




Edit_Delete_StreamGage

**Deleting a Stream Gage Station**

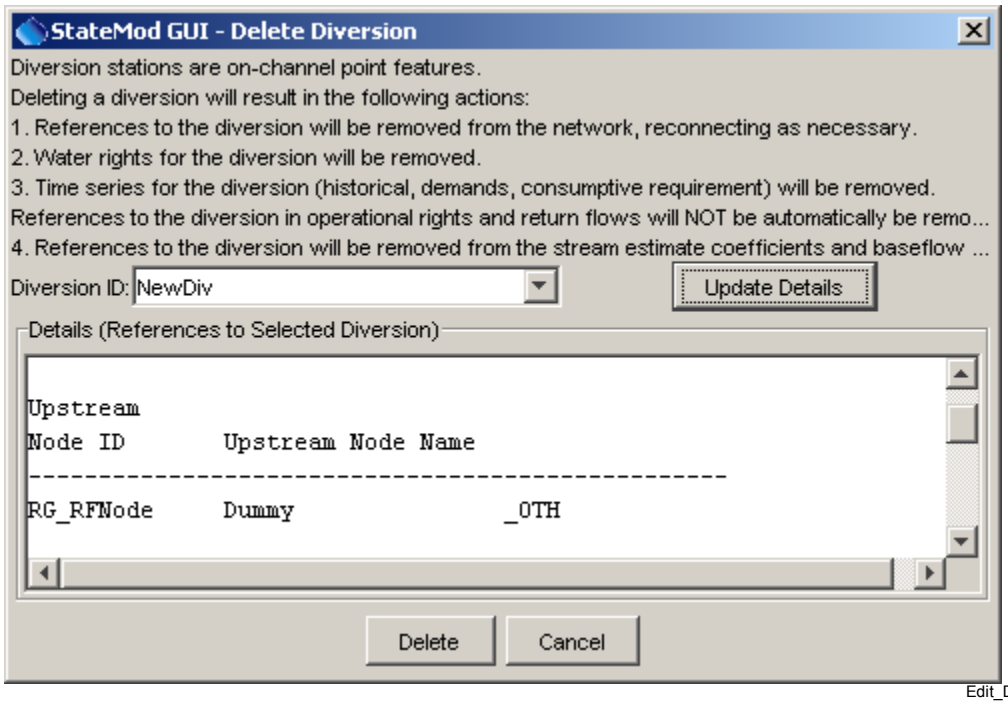
### 4.10.2 Deleting Delay Table Data

The ***Edit…Delete…Delay Table (Monthly)…*** menu deletes a monthly delay table:

The ***Edit…Delete…Delay Table (Daily)…*** menu deletes a daily delay table:

### 4.10.3 Deleting Diversion Data

The ***Edit…Delete…Diversion…*** menu deletes a diversion station:




Edit_Delete_Diversion

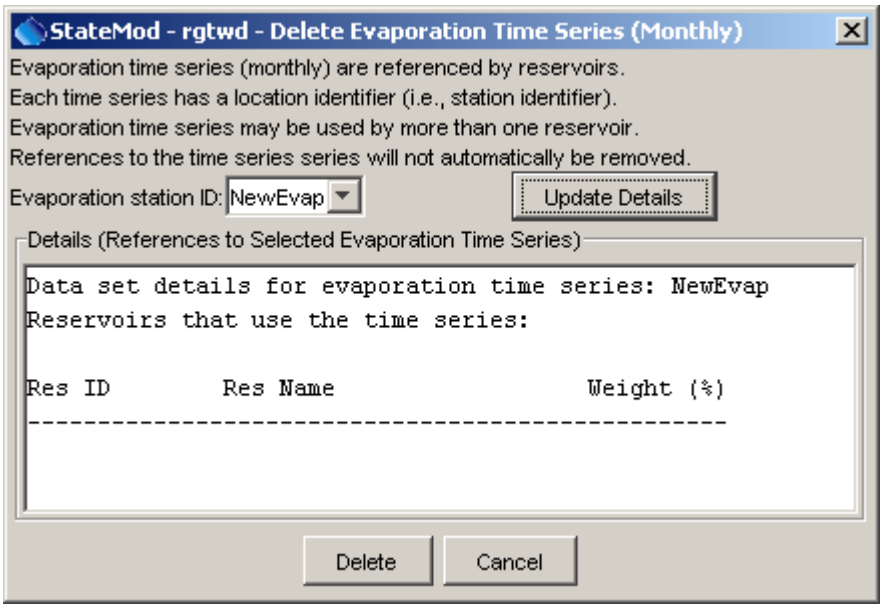
**Deleting a Diversion Station**

### 4.10.4 Deleting Precipitation Data

The ***Edit…Delete…Precipitation TS (Monthly)…*** menu deletes a monthly precipitation time series.

### 4.10.5 Deleting Evaporation Data

The ***Edit…Delete…Evaporation TS (Monthly)…*** menu deletes a monthly evaporation time series:

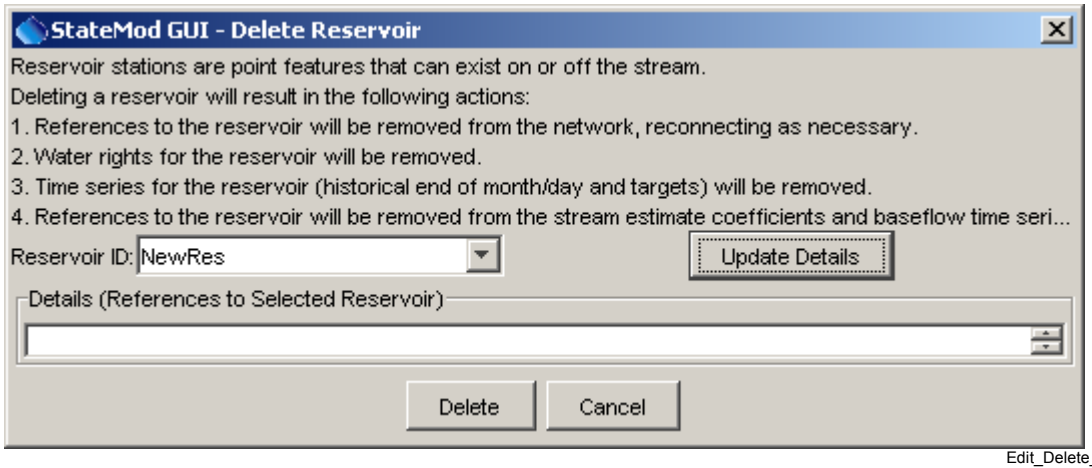



Edit_Delete_Evap

**Deleting an Evaporation Time Series**

### 4.10.6 Deleting Reservoir Data

The ***Edit…Delete…Reservoir…*** menu deletes a reservoir station:

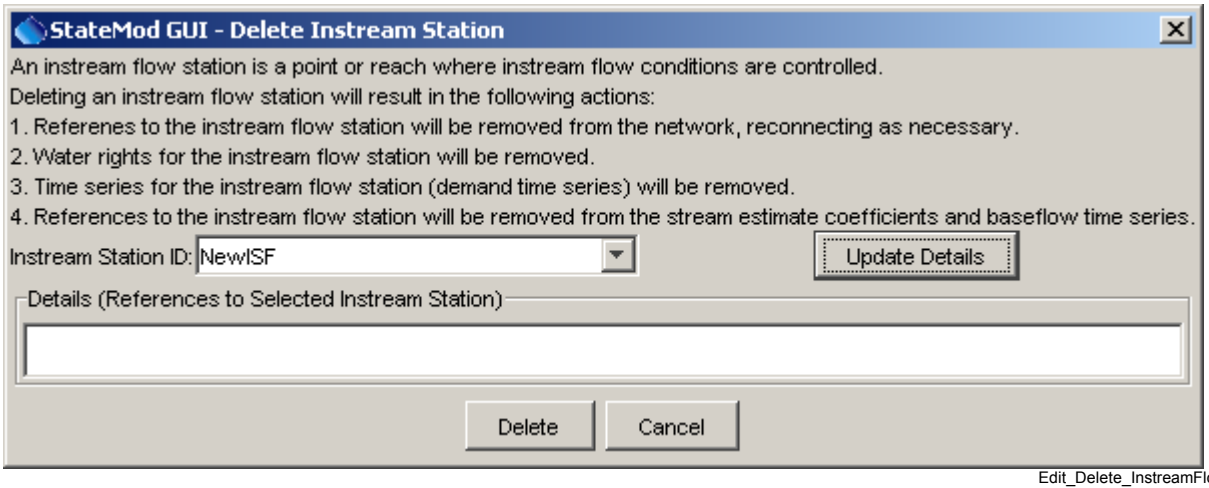



Edit_Delete_Reservoir

**Deleting a Reservoir Station**

### 4.10.7 Deleting Instream Flow Data

The ***Edit…Delete…Instream Flow…*** menu deletes an instream flow station:

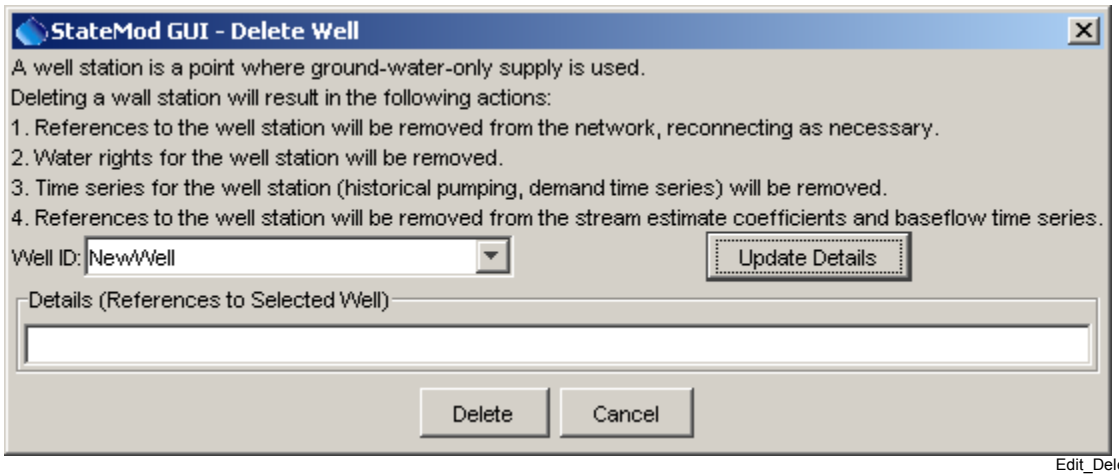



Edit_Delete_InstreamFlow

**Deleting an Instream Flow Station**

### 4.10.8 Deleting Well Data

The ***Edit…Delete…Well…*** menu deletes a well station:

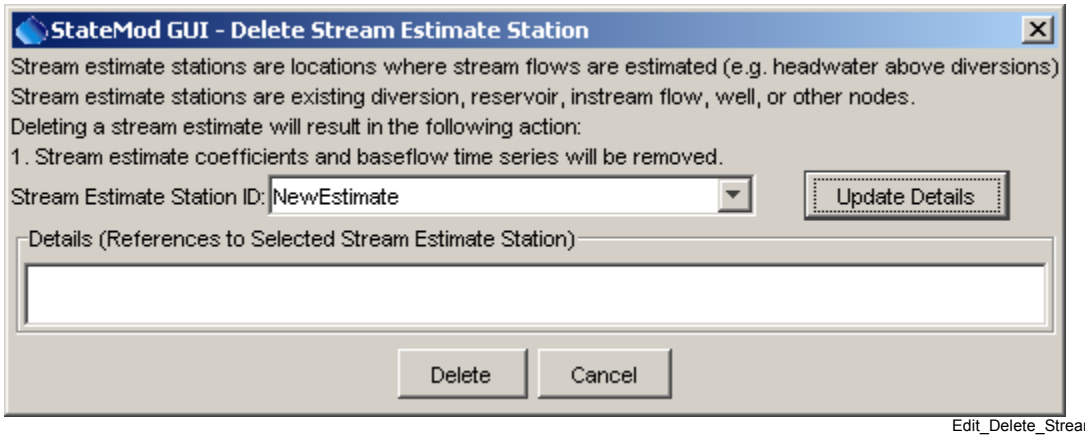



Edit_Delete_Well

**Deleting a Well Station**

### 4.10.9 Deleting Stream Estimate Data

The ***Edit…Delete…Stream Estimate Station…*** menu deletes a stream estimate station:

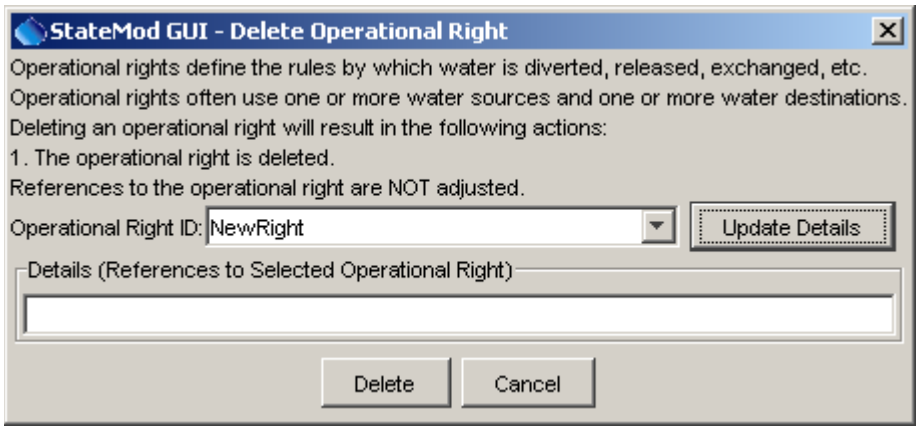



Edit_Delete_StreamEstimate

**Deleting a Stream Estimate Station**

### 4.10.10 Deleting Operational Right Data

The ***Edit…Delete…Operational Right…*** menu deletes an operational right:




Edit_Delete_OpRight

**Deleting an Operational Right**

This page is intentionally blank.

# 5 Viewing and Editing Data

Version 07.04.00, 2013-04-18

The ***Data*** menu allows data sets to be viewed and edited, with data being organized according to the major StateMod data types:


Menu_Data

The ***Data*** menu items are listed in approximately the order of creation, similar to the CDSS StateDMI software. In particular, components that depend on other components are generally listed last.

The primary components each have a main display window, which may display data from one or more StateMod data files. For example, diversion data contains diversion station information, water rights, delay tables, and associated time series. Consequently, there are fewer ***Data*** menu items than there are StateMod files. See the StateMod software documentation for information about the model data. The StateMod GUI allows changes to data but in many cases cannot check for data connections (e.g., it does not know which delay table to use for return flows when adding a new structure). It is therefore important that users understand the implications of editing data and realize that changes in more than one window may be necessary for a working data set. The data windows do not allow adding new model features or deleting existing features. To do so, use the ***Edit*** menu, which enforces consistency between the model network and other files. Each edit feature is described in the following sections.

## 5.1 Control Data

The ***Data…Control*** menu allows viewing/editing of control file information:


Menu_Data_Control

Control data consists of important properties and also organizes the files in the data set. Each control data component is described in the following sections.

### 5.1.1 Response File

The ***Data…Control…Response*** menu displays the response file contents, which lists files being used for the current data set:

StateMod - rgtwd - Response File Contents

To rename a data set component, select a row and either type a new file name or use the Browse button.
If ARE DATA MODIFIED? is YES, data for the component have been modified byt the file has not been written
Consequently, StateMod will not recognize the changes until the data are saved with File...Save.
The file with the original filename will remain even after the new file is saved.
Data set base name (from *.rsp): rgtwd
Data set directory: C:\CDSS_BeforeDVD\statemod\data\rgTWday2

| DATA GROUP | DATA SET COMPONENT | FILE NAME | ARE DATA MODIFIED? |
|---|---|---|---|
| Control Data | Response | rgtwd.rsp | |
| Control Data | Control | rgTWD.ctl | |
| Control Data | Output Request | sam.out | |
| Stream Gage Data | Stream Gage Stations | rgTW.ris | |
| Stream Gage Data | Stream Gage Historical TS (Monthly) | rgTW.rih | |
| Stream Gage Data | Stream Gage Historical TS (Daily) | rgTWD.rhy | |
| Stream Gage Data | Stream Gage Base TS (Monthly) | rgtw.rim | |
| Stream Gage Data | Stream Gage Base TS (Daily) | rgTWD.rid | |
| Delay Table (Monthly) Data | Delay Tables (Monthly) | rgTW.dly | YES |
| Delay Table (Daily) Data | Delay Tables (Daily) | rgTwD.dld | |
| Diversion Data | Diversion Stations | rgTW.dds | |
| Diversion Data | Diversion Rights | rgTW.ddr | |
| Diversion Data | Diversion Historical TS (Monthly) | rgTW.ddh | |
| Diversion Data | Diversion Historical TS (Daily) | rgTWD.dhy | |
| Diversion Data | Diversion Demand TS (Monthly) | rgTW.ddm | |
| Diversion Data | Diversion Demand TS Override (Monthly) | | |
| Diversion Data | Diversion Demand TS (Average Monthly) | | |
| Diversion Data | Diversion Demand TS (Daily) | | |
| Diversion Data | Irrigation Practice TS (Yearly) | rg.tsp | |
| Diversion Data | Consumptive Water Requirement TS (Monthly) | rg.iwr | |
| Diversion Data | Consumptive Water Requirement TS (Daily) | rgTWD.iwd | |
| Diversion Data | Soil Moisture | rg.par | |
| Precipitation Data | Precipitation Time Series (Monthly) | | |
| Evaporation Data | Evaporation Time Series (Monthly) | rgTW.eva | |
| Reservoir Data | Reservoir Stations | rgTW.res | |
| Reservoir Data | Reservoir Rights | rgTW.rer | |
| Reservoir Data | Reservoir Content TS, End of Month (Monthly) | rgTW.eom | |
| Reservoir Data | Reservoir Content TS, End of Day (Daily) | rgTWD.eoy | |
| Reservoir Data | Reservoir Target TS (Monthly) | rgTW.tar | |
| Reservoir Data | Reservoir Target TS (Daily) | | |
| Instream Flow Data | Instream Flow Stations | rgTW.ifs | |
| Instream Flow Data | Instream Flow Rights | rgTW.ifr | |
| Instream Flow Data | Instream Flow Demand TS (Monthly) | rgTW.ifm | |
| Instream Flow Data | Instream Flow Demand TS (Average Monthly) | rgTW.ifa | |
| Instream Flow Data | Instream Flow Demand TS (Daily) | | |
| Well Data | Well Stations | rgTW.wes | |
| Well Data | Well Rights | rgTW.wer | |
| Well Data | Well Historical Pumping TS (Monthly) | rgTW.weh | |
| Well Data | Well Historical Pumping TS (Daily) | rgTWD.why | |
| Well Data | Well Demand TS (Monthly) | rgTW.wem | |
| Well Data | Well Demand TS (Daily) | | |
| Stream Estimate Data | Stream Estimate Stations | rgTW.ris | |
| Stream Estimate Data | Stream Estimate Coefficients | rgTW.rib | |
| River Network Data | River Network | rgTW.rin | |
| River Network Data | Network (Graphical) | rgtw.net | |
| Operational Data | Operational Rights | rgTW.opr | YES |
| Spatial Data | GeoView Project | gis\rgTW_StateMod.gvp | |

Browse... Apply Cancel Close

Data_Control_Response

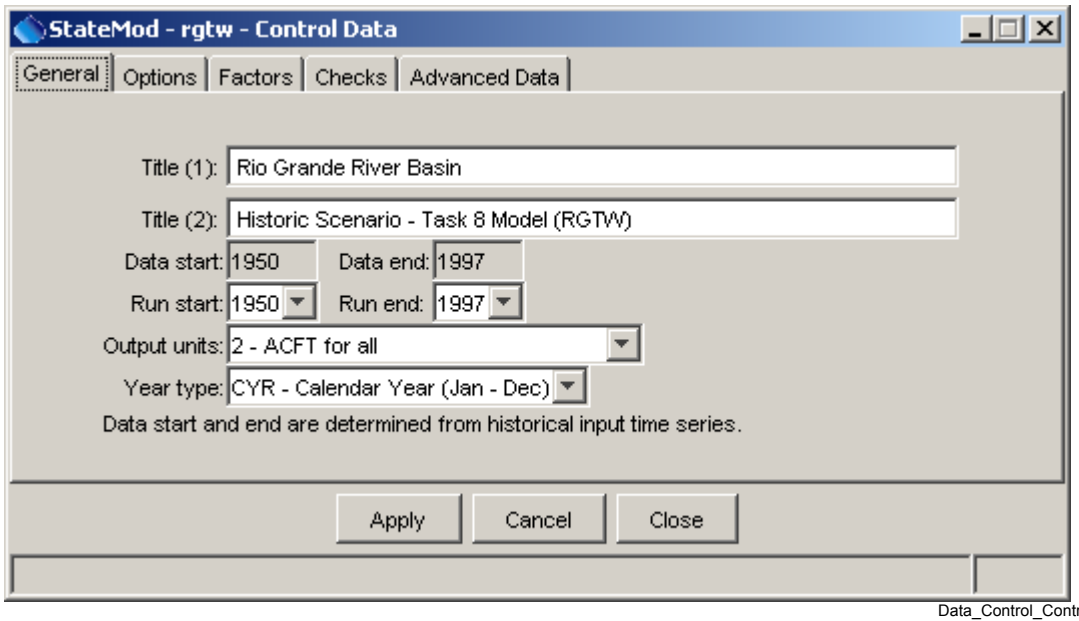
**Response File Data Window**

The response file display shows the primary data group, the specific data set component, and the corresponding file name in the response file. Relative paths are encouraged to promote portability of data sets. The ***ARE DATA MODIFIED?*** column indicates if data files have been modified within the StateMod GUI. If ***YES*** is indicated, then the associated files will need to be saved with ***File…Save*** before the StateMod software will recognize the changes in a run. If model file formats change over time, the data files may automatically be tagged as being modified when read. Saving the files will result in using the new file format.

There is typically no need to modify filenames and consistent filenames are encouraged. The names of files in a data set typically have the same base name, with file extensions following StateMod modeling conventions. In older versions of StateMod, it was often necessary to use an empty "dummy" file in the response file. This is no longer needed and the above figure illustrates how omitting a file name indicates that the data component is not used in the data set.

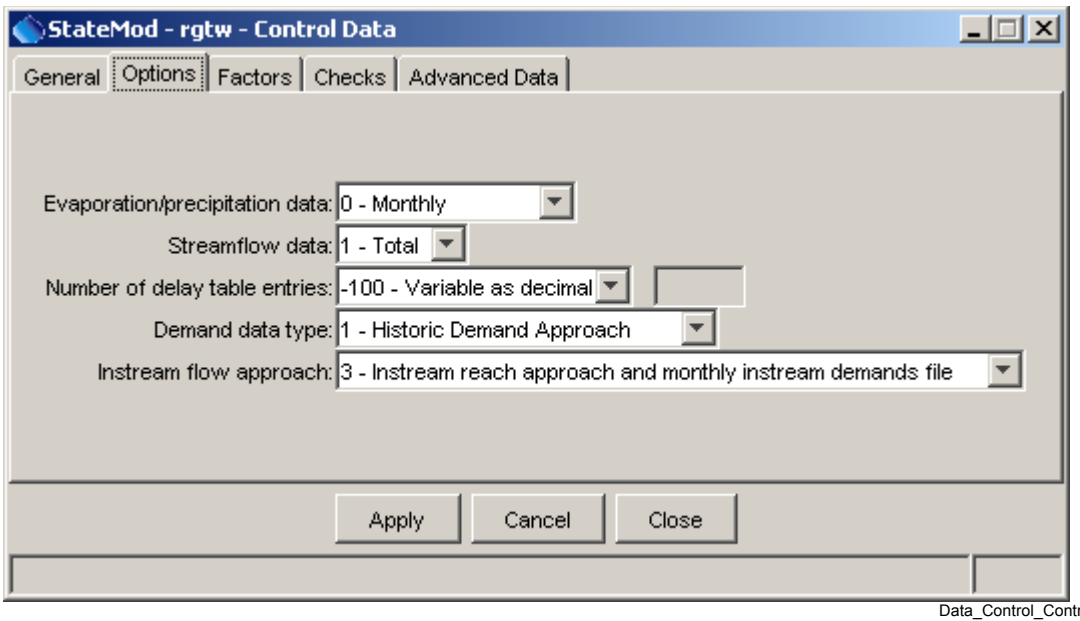
### 5.1.2 Control File

The ***Data…Control...Control*** menu allows viewing/editing the StateMod control data, which consists primarily of string and numeric properties for the data set. The control data are presented using a tabbed panel window, as shown in the following figures.



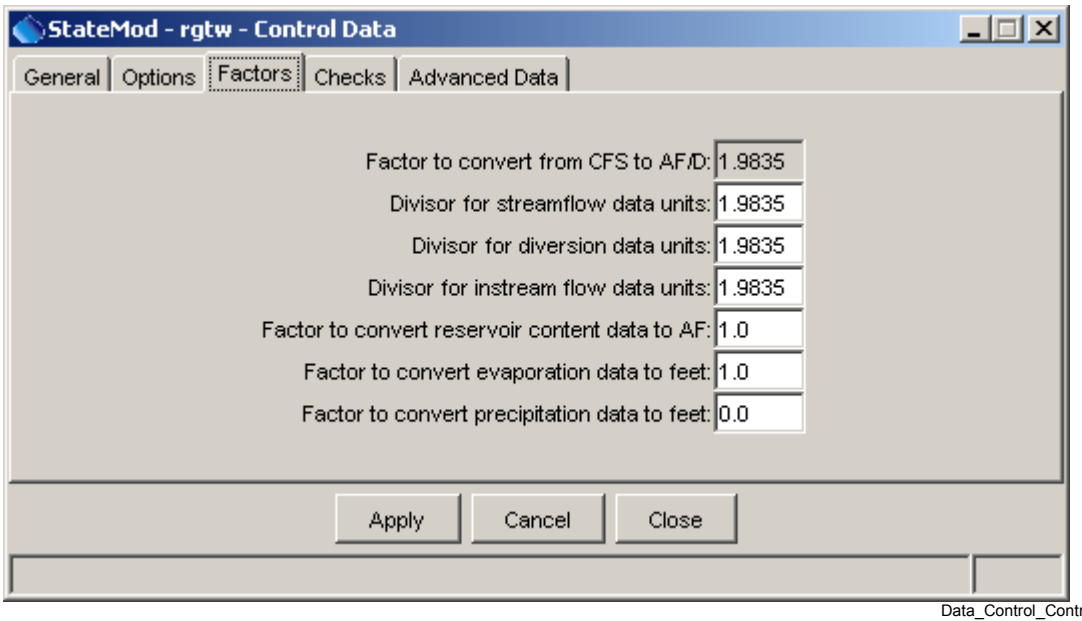

Data_Control_Control_General

**Control Data – General Properties**



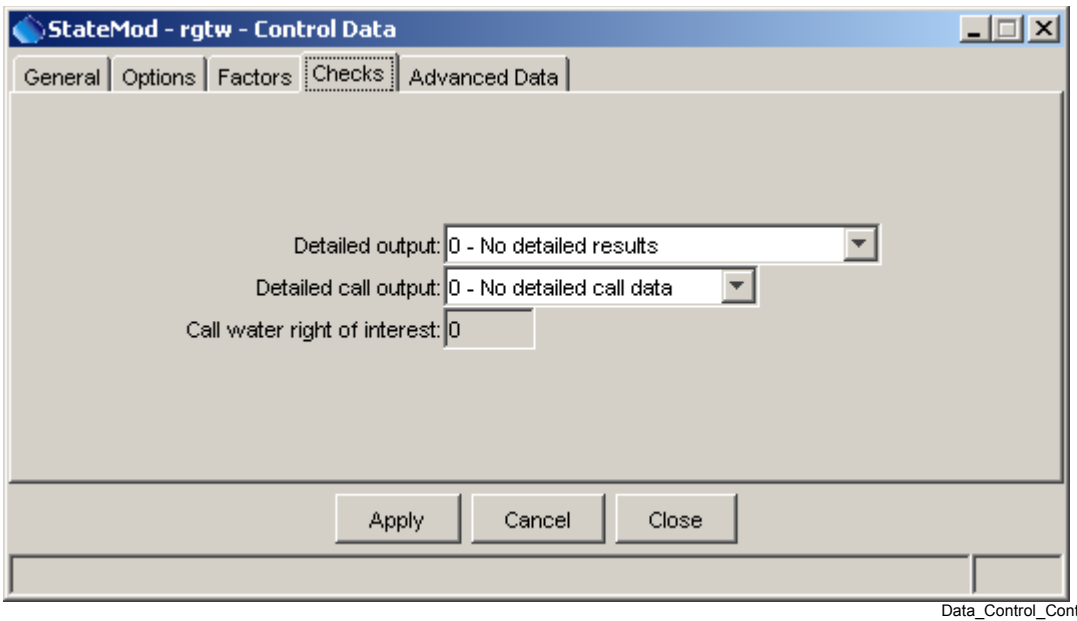

Data_Control_Control_Options

**Control Data – Option Properties**



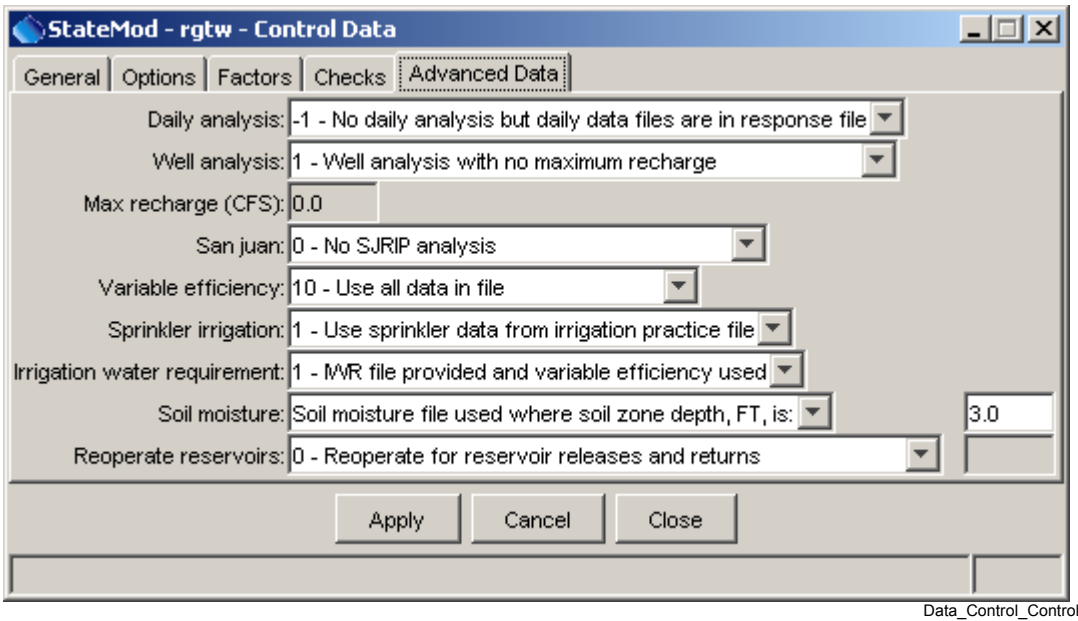

Data_Control_Control_Factors

**Control Data – Conversion Factor Properties**




Data_Control_Control_Checks

**Control Data – Data Check Properties**



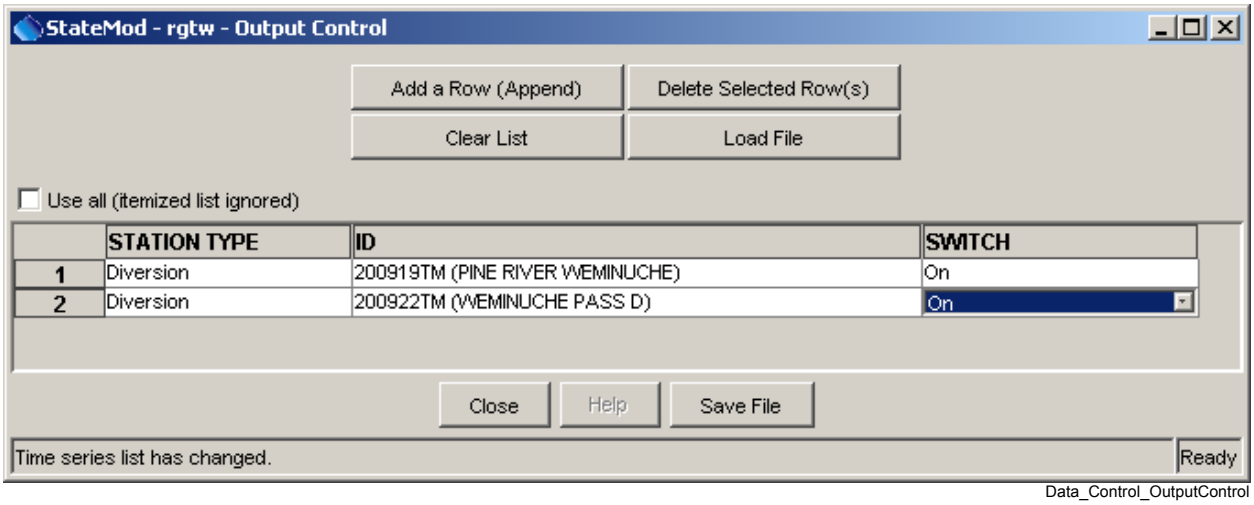

Data_Control_Control_Advanced

**Control Data – Advanced Properties**

Refer to the StateMod documentation for more information about the meaning of control data. Changing the run period of record is one of the more useful features of this window and limits the model output (the graphing tool also uses the run period to determine the period to graph). The StateMod GUI determines the data period of record by evaluating all the input time series files and finding the maximum period of record (if time series are not read when selecting the data set, then only the streamflow time series file is used for the data period). The StateMod GUI shows standard choices for control data and will attempt to pass through unrecognized values. This allows for enhancement to the StateMod software.

### 5.1.3 Output Control File

The ***Data…Control…Output Request*** menu displays a dialog that controls how much output StateMod will produce.




Data_Control_OutputControl

**Output Control Window**

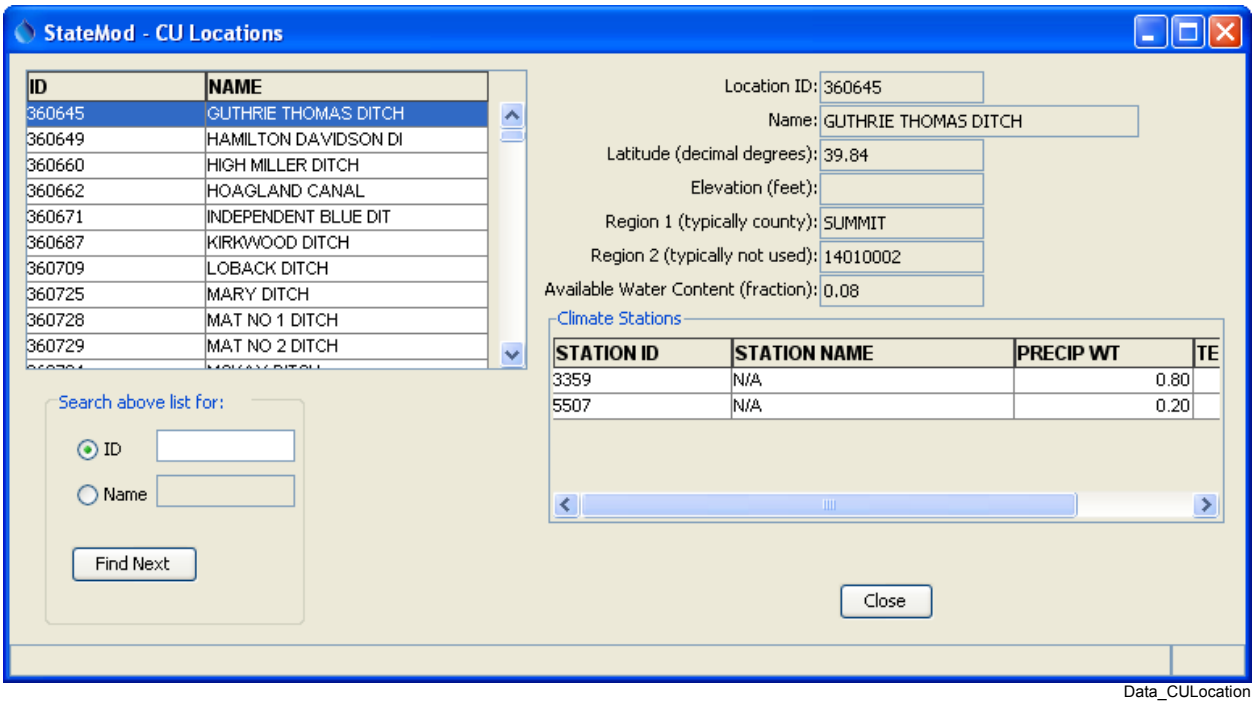
The output control interface edits the StateMod output control file. The purpose of the output control file is to limit the StateMod output to a few structures. This reduces model execution time but also limits the output that can be viewed after a run. The output control file name is usually left constant in the response file, being overwritten as needed. StateMod, when run in data check mode, will automatically create a template output control file with extension *.xou*. If the response file also uses this name, then a custom output control file may be overwritten when StateMod runs in check mode.

The output control window creates a list of stations that will have output generated when the model runs. Stations can be selected either by manually entering the identifiers or relying on the interface to generate the identifiers. An output control file also allows all structures to be output. Selecting the ***Use all*** checkbox turns on this feature and the structures in the list are ignored (but will be included in the output control file for subsequent manipulation of the file).

Adding a row initializes a new row in the list. Selecting a station type results in a selectable list of identifiers, which will be alphabetized and include stations for the selected station type. The switch should be set to `On` but can be set to `Off` to deactivate the item. When finished editing the contents of an output control file, select ***Save File***. A file selector is displayed with a default file extension of *.tpo* (although the file extensions *.out* and *.xou* are also often used). To be recognized by StateMod, this file name should match the output control file in the response file. If necessary, rename the output control file outside the StateMod GUI or read a file and then re-save using a different name.

## 5.2 Consumptive Use Data

The ***Data…Consumptive Use*** menu displays the consumptive use locations and associated data:



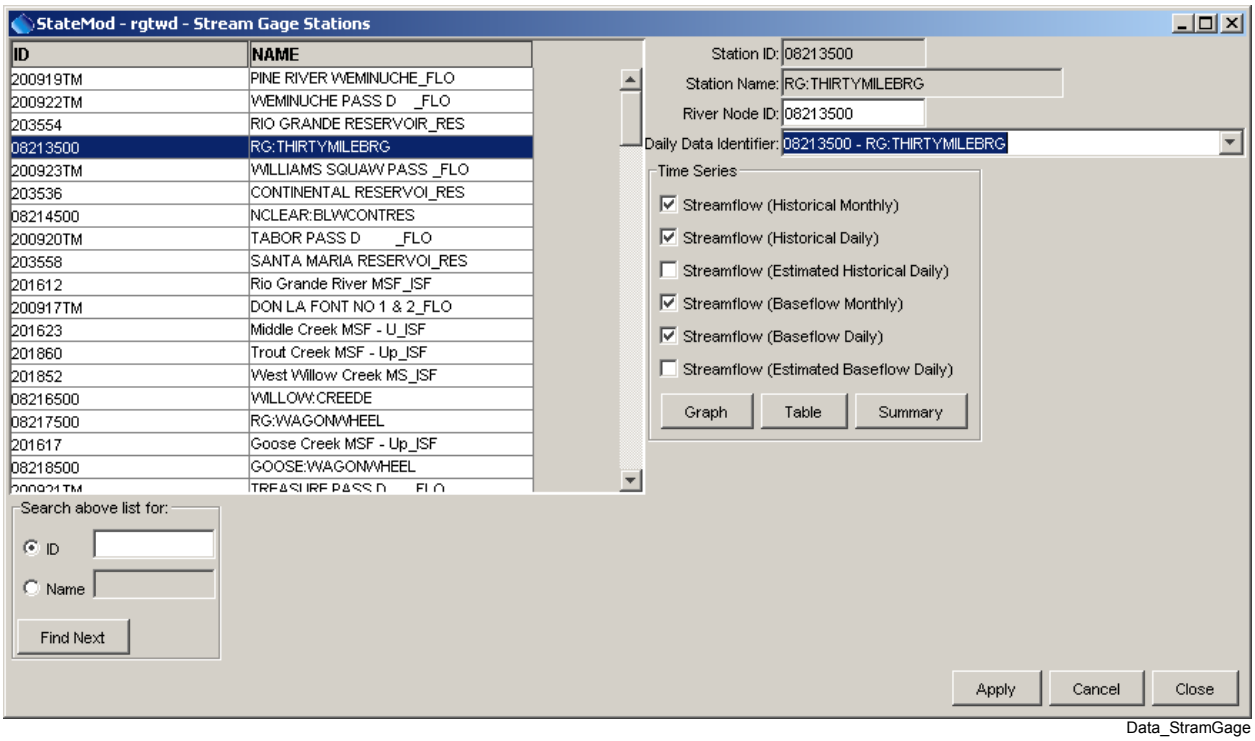

Data_CULocation

**Consumptive Use Locations Data Window**

Consumptive use locations correspond to locations that supply consumptive use data (e.g., irrigation water requirement) data. Currently the data are read-only and the irrigation water requirement time series are not accessible from the window.

## 5.3 Stream Gage Data

The ***Data…Stream Gage*** menu displays the stream (river) gage stations:



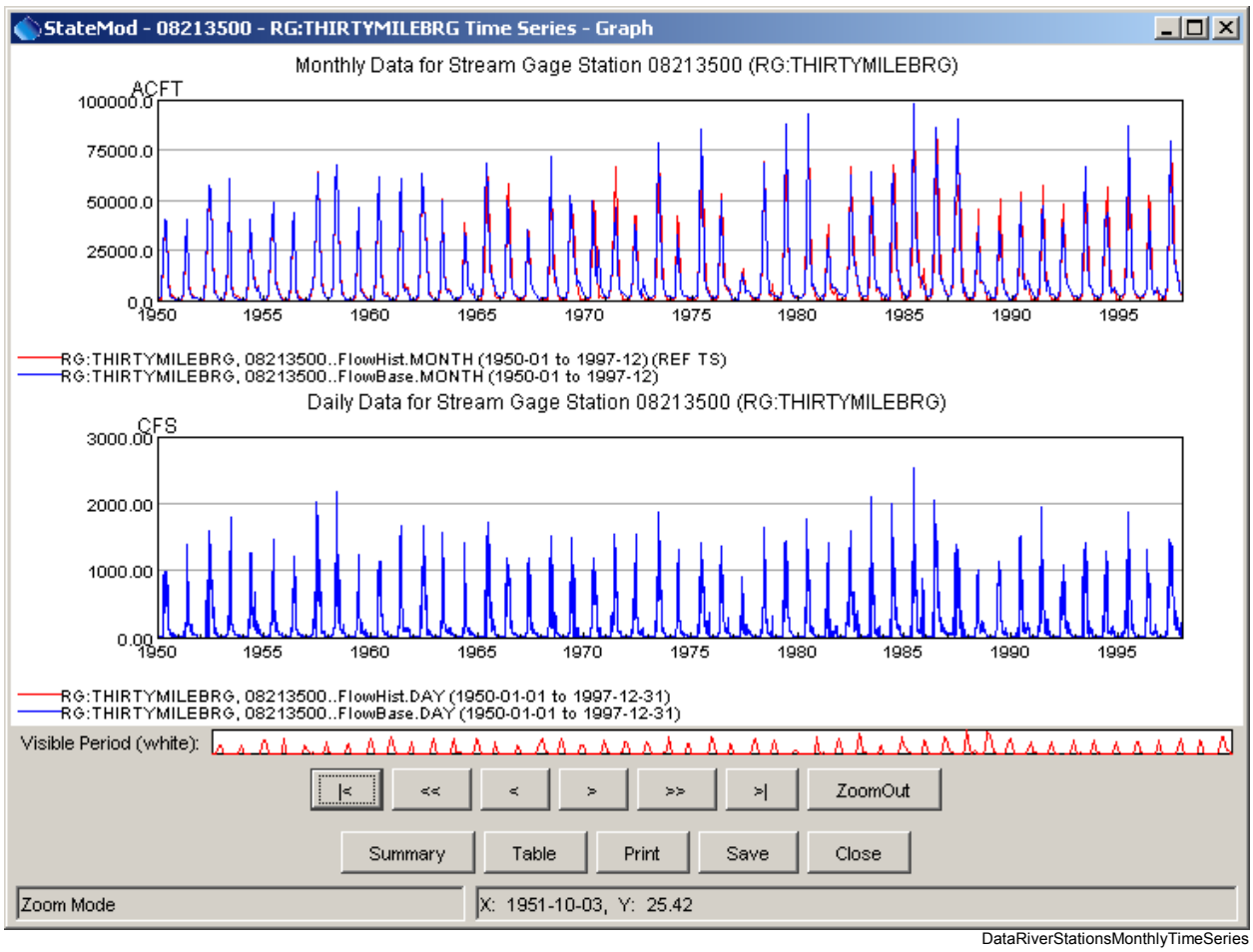

Data_StramGage

**River Stations Data Window**

River stations consist of points with historical streamflow data (the alternative being stream estimate stations where streamflow is estimated using baseflow coefficients – see **Section 5.12**). Selecting a station from the list on the left will display the station data on the right side of the window. The station data shown in the upper right can be edited and saved.

Use the ***Edit*** menu to add or delete a station.

A list of available input time series is shown and can be selected for display with the ***Graph***, ***Table***, and ***Summary*** buttons. If the data set does not contain certain time series, then checkboxes will be disabled. For example, a streamflow gage may have historical time series data and baseflow time series, which is created by running StateMod in baseflow mode. All available time series are shown on a single "page", with separate graphs used to group similar data intervals, as shown in the following example:




DataRiverStationsMonthlyTimeSeries

**River Station Monthly Time Series Graph**

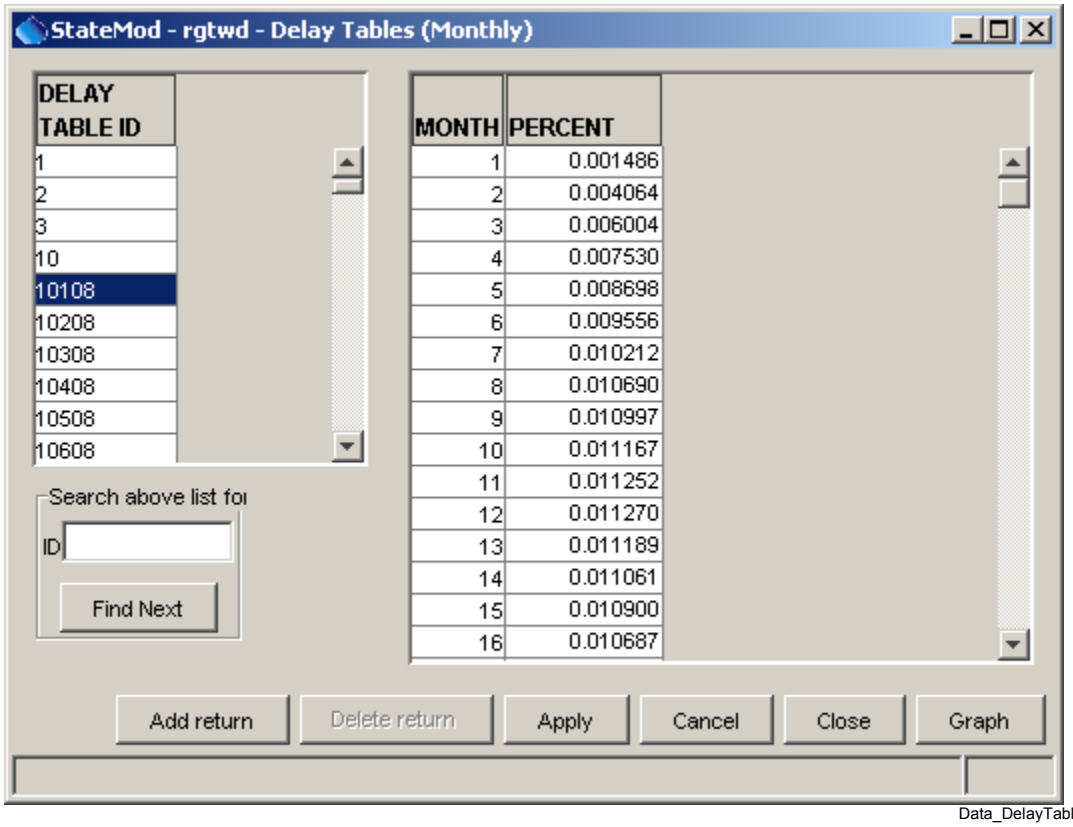
The data types for the time series are consistent with the nomenclature used in the ***Results…Graphing Tool*** menu, where possible. The ***Summary*** button can be used to display a text summary of the time series. The ***Table*** button can be used to view the time series in tabular form. Refer to the **TSView Time Series Viewing Tools Appendix** for more information about the graphing tools.

## 5.4 Delay Table Data

The ***Data…Delay Tables*** menu displays delay table information.

### 5.4.1 Delay Tables (Monthly)

The ***Data…Delay Tables…Monthly*** menu displays the list of monthly delay tables, which are used by diversion stations and well stations when indicating return flows or depletions.




Data_DelayTables_Monthly

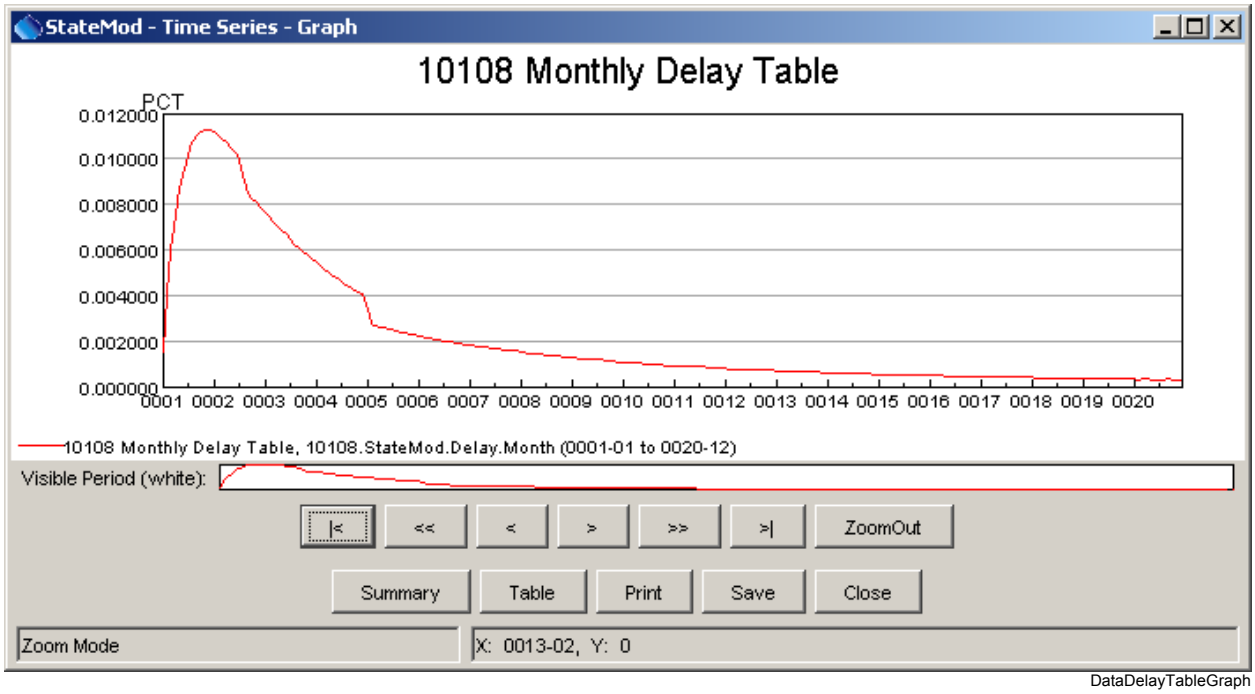
**Delay Table Data (Monthly)**

Delay table information can be specified as percent or fraction (decimal), depending on the *interv* control file parameter. To add a new delay table entry, select the ***Add return*** button. This will add a row to the display with default data above the highlighted row. If the last row is highlighted and a row is added, a prompt will allow adding the row above or below the last row. Fill in the data as appropriate. To delete a delay table entry, select a row and then press the ***Delete return*** button.

Care should be taken when modifying delay tables because delay tables can be used by more than one station. It may be necessary to modify the diversion or well station delay table assignments.

Use the ***Edit*** menu to add or delete a delay table.

Pressing the ***Graph*** button displays the delay table as shown in the following figure, treating the data as a time series where the initial year is one (the following example shows a delay table that spans twenty years):

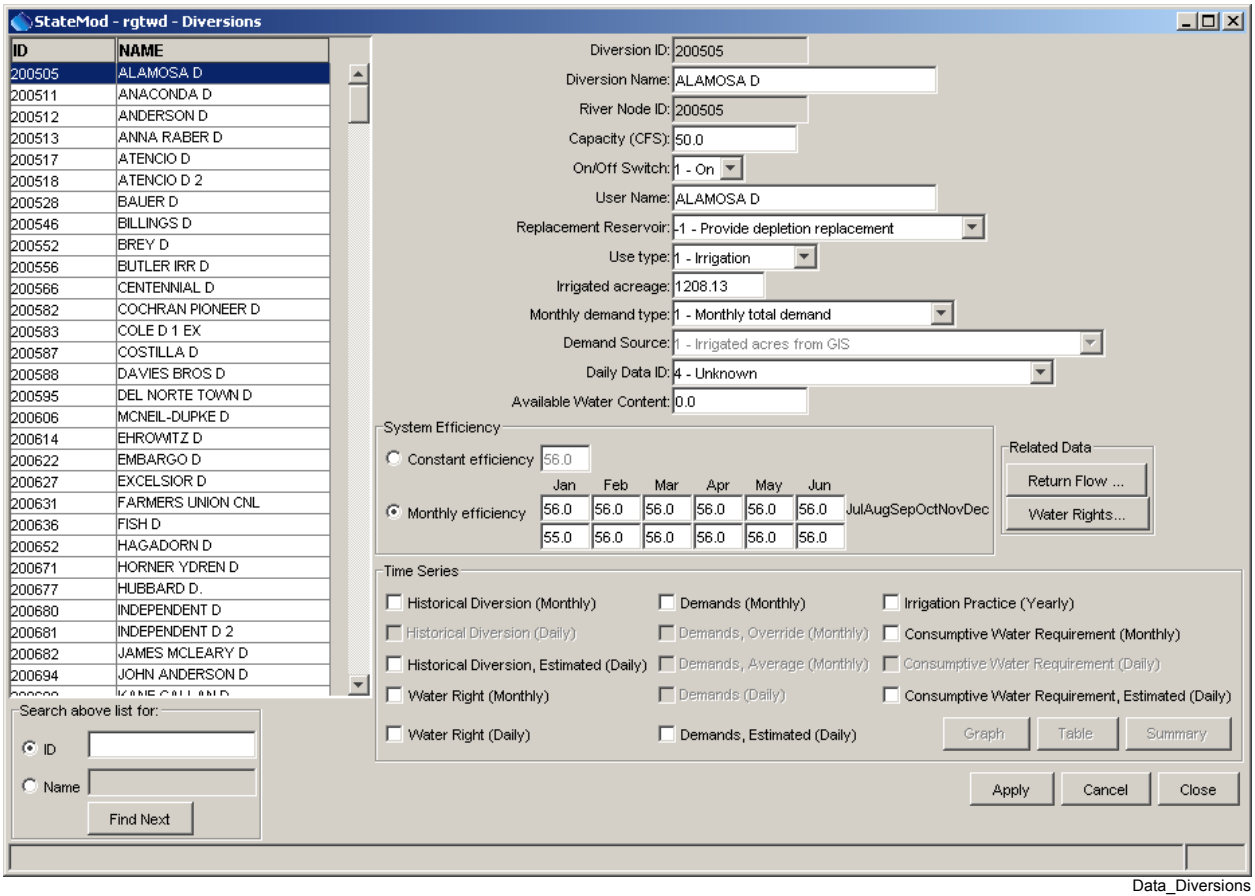



DataDelayTableGraph

**Monthly Delay Table Graph**

## 5.5 Diversion Data

The ***Data…Diversions*** menu displays information for diversions. The primary data component is diversion stations, and secondary components are water rights, historical time series, demand time series, return flows (delay table assignments).



Data_Diversions

**Diversion Data Window**

All diversion stations in the data set are listed on the left side of the window. Selecting a diversion from the list displays that diversion station's information in the window. Secondary data are displayed in additional windows accessed via buttons. Access to other data is disabled if the data were not read or are not a part of the data set. All data are editable except for the identifier and river node identifier, which are referenced in the network and other data. Press the ***Apply*** button after making changes. Changes are also applied if the ***Clos***e button is pressed. Use the main ***Edit*** menu to add or delete diversion stations and optionally the secondary data.

The list of diversions can be search by entering the identifier or name in the appropriate search boxes located below the list. Any number of characters can be entered in the search box. Press ***Enter*** to perform the case-insensitive search, starting at the top of the list. The ***Find Next*** button, when pressed, will find the next station that matches the information.

The diversion efficiency is displayed for each month in the year. If the diversion has a constant efficiency, the same value is displayed in each of the twelve monthly fields. Variable efficiencies, if

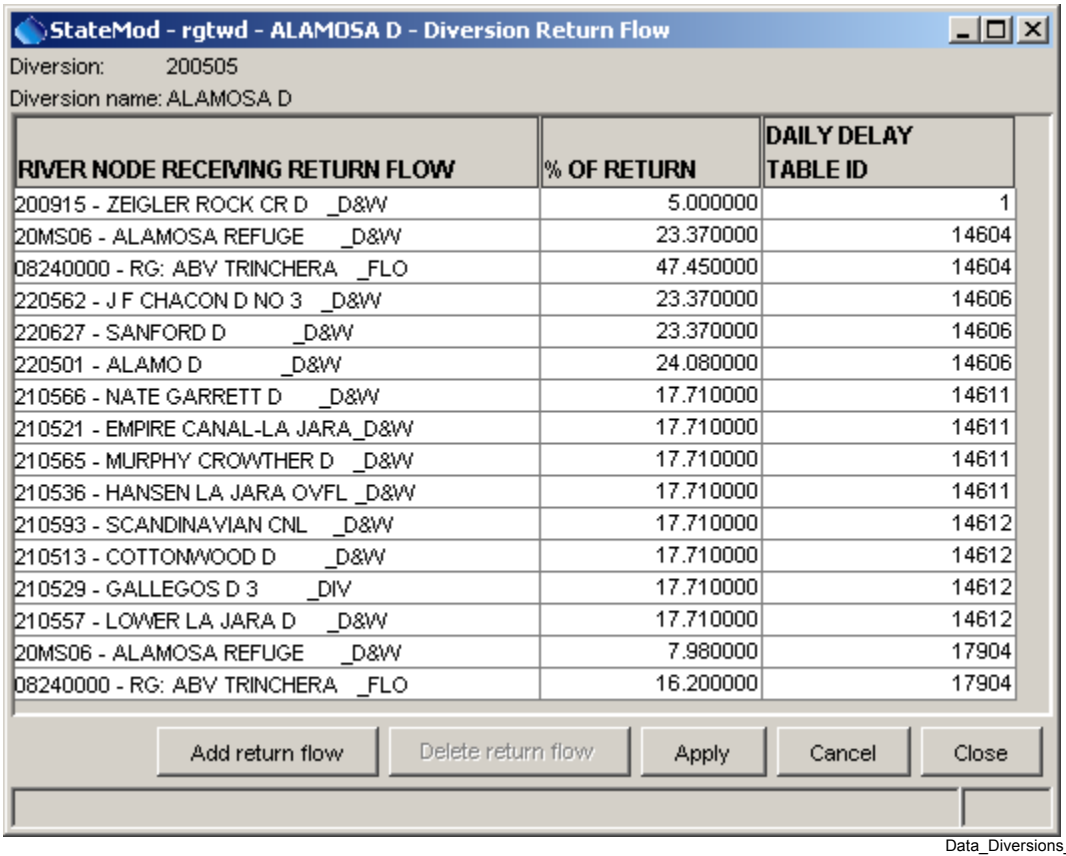
used in modeling, are not shown (only the average efficiencies shown in the diversion station file are shown).

The ***Daily Data ID*** is used to associate a daily time series with a diversion station. Changes to this value take effect when either a different diversion in the list is chosen or when ***Enter*** is pressed in the ***Daily Data ID*** text field. The following options are available:

- If the ***Daily Data ID*** exactly matches the ***Diversion ID***, the pattern and values are the same.
- If the ***Daily Data ID*** is "0", the pattern and values are again the same but are the average daily values, calculated using the monthly time series.
- If the ***Daily Data ID*** does not match the ***Diversion ID*** and is not "0", the pattern time series corresponding to the ***Daily Data ID*** is displayed as-is. However, the values time series is calculated using the pattern and known monthly totals. The daily time series monthly total should agree with the monthly time series, but the distribution should correspond with the pattern. More than one diversion may reference the same pattern. For that reason, care should be taken when changing the pattern itself.
- Additional options may be available from StateMod. Refer to the StateMod software documentation.

### 5.5.1 Diversion Station Return Flows

Diversion station return flows (delay table assignments) are displayed by pressing the ***Return Flow*** button in the main diversions window:



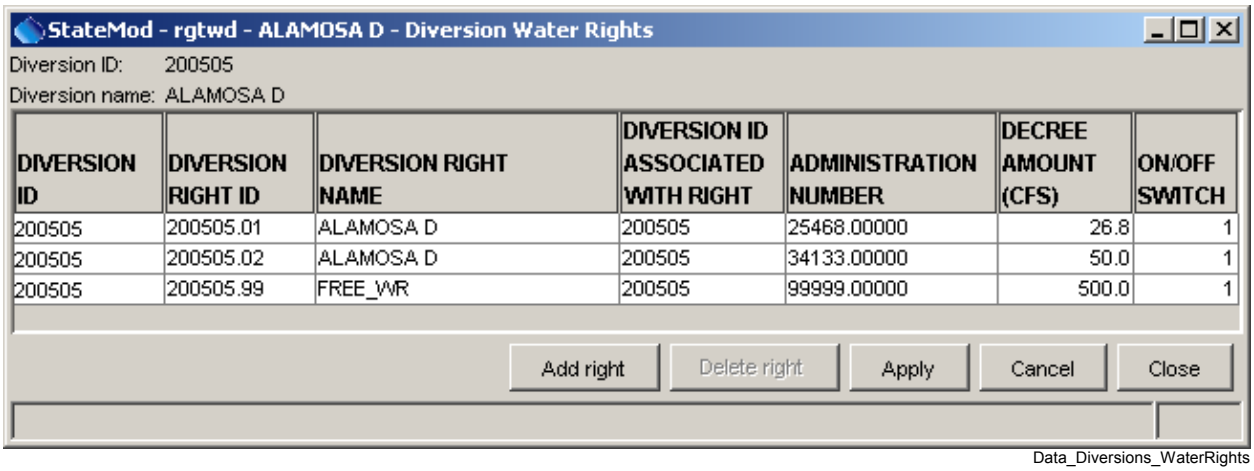

Data_Diversions_ReturnFlow

**Diversion Return Flow Data**

Press the ***Add return flow*** button to add a row to the display. Fill in the information as appropriate, using the choices that are provided. To delete a return flow, select a row and press the ***Delete return flow*** button. Press the ***Apply*** button to update the reservoir station data. The ***Close*** button will apply changes and close the window.

### 5.5.2 Diversion Water Rights

The water rights that apply to the diversion can be viewed by pressing the ***Water Rights*** button in the diversion data window, resulting in a display as shown in the following figure:



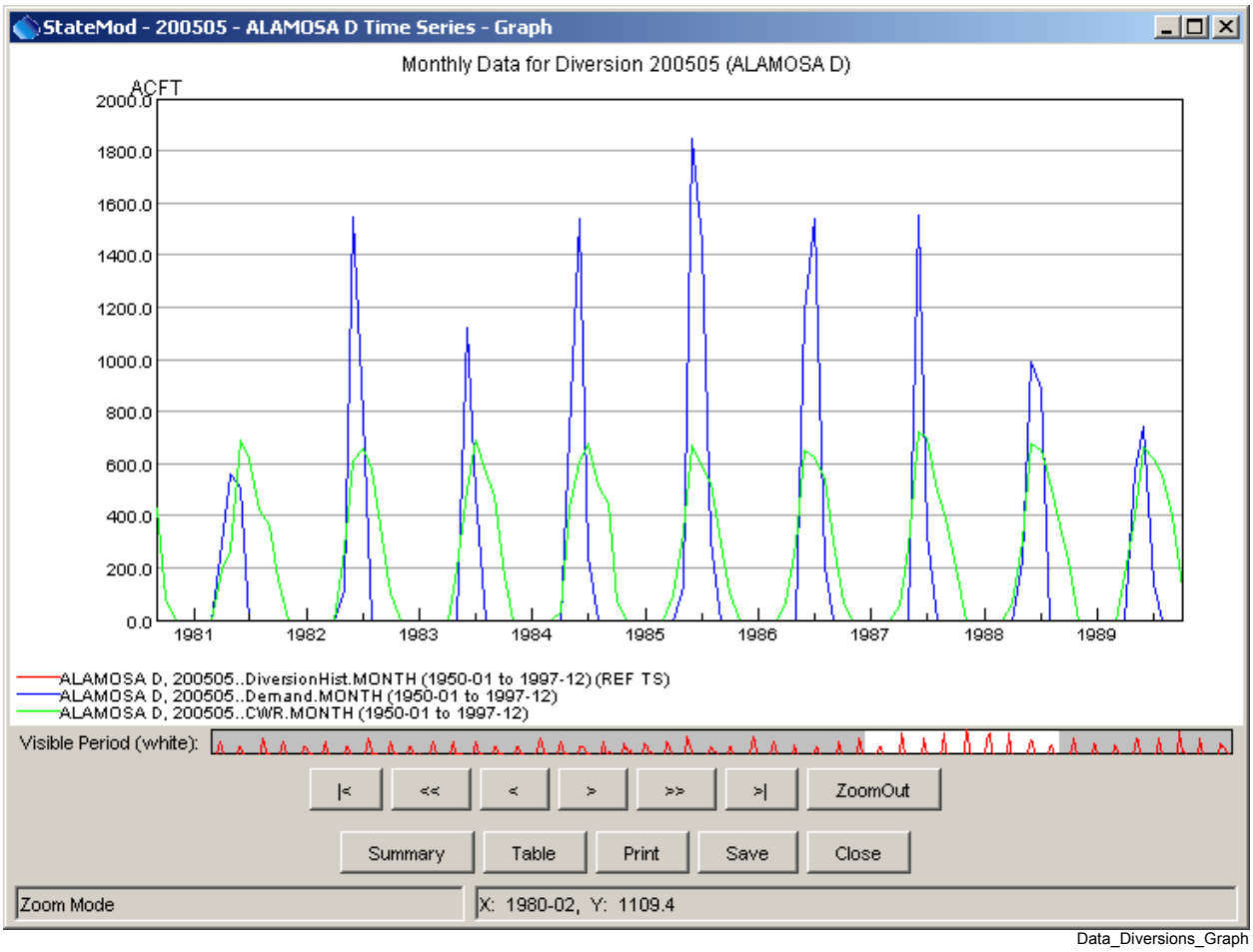

Data_Diversions_WaterRights

**Diversion Water Rights Data Window**

To add a water right, press ***Add right***. This will add a row with default information, which should be updated as appropriate. To delete a water right, select the row and press ***Delete right***. Press the ***Apply*** button to update the reservoir station data. The ***Close*** button will apply changes and close the window.

### 5.5.3 Diversion Time Series

The bottom of the main diversions window lists all time series associated with diversion stations. The checkboxes next to time series are enabled according to the data that are available for the currently selected diversion station. To view time series, select one or more time series and press the ***Graph***, ***Table***, or ***Summary*** buttons. The following figure illustrates a graph of monthly data:




Data_Diversions_Graph

**Diversion Time Series (Monthly)**

The data types for the time series are consistent with the nomenclature used in the ***Results…Graphing Tool*** menu, where possible. The ***Summary*** button can be used to display a text summary of the time series. The ***Table*** button can be used to view the time series in tabular form. Refer to the **TSView Time Series Viewing Tools Appendix** for more information about the graphing tools.

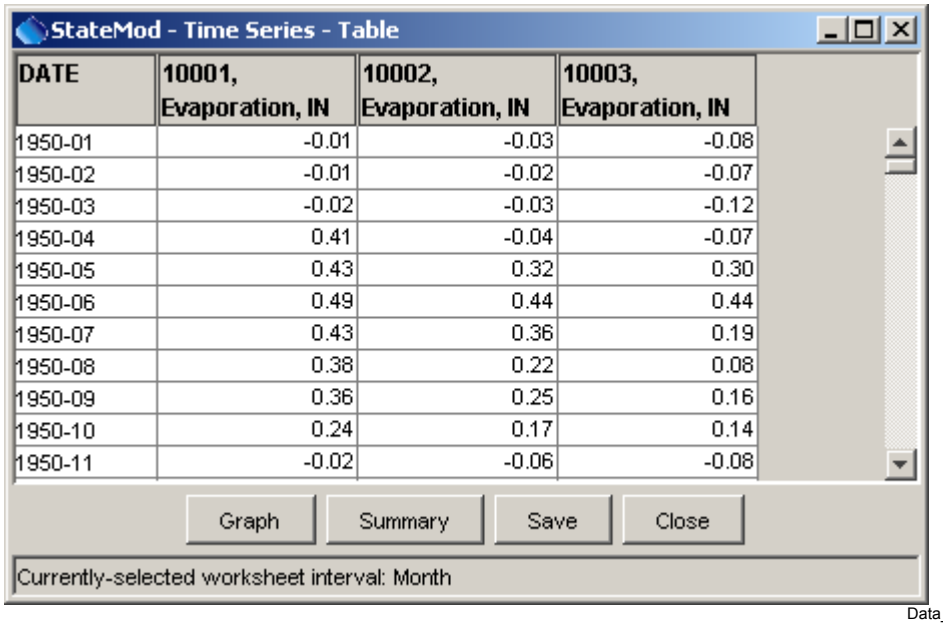
## 5.6 Precipitation Data

The ***Data…Precipitation…*** menu displays monthly precipitation data (average annual precipitation currently cannot be displayed). Precipitation data consist of precipitation time series and are used to compute net reservoir evaporation. Frequently, net evaporation is pre-calculated and is provided in the evaporation data. Refer to the next section for information about viewing evaporation data. The display features for precipitation and evaporation data are similar.

## 5.7 Evaporation Data

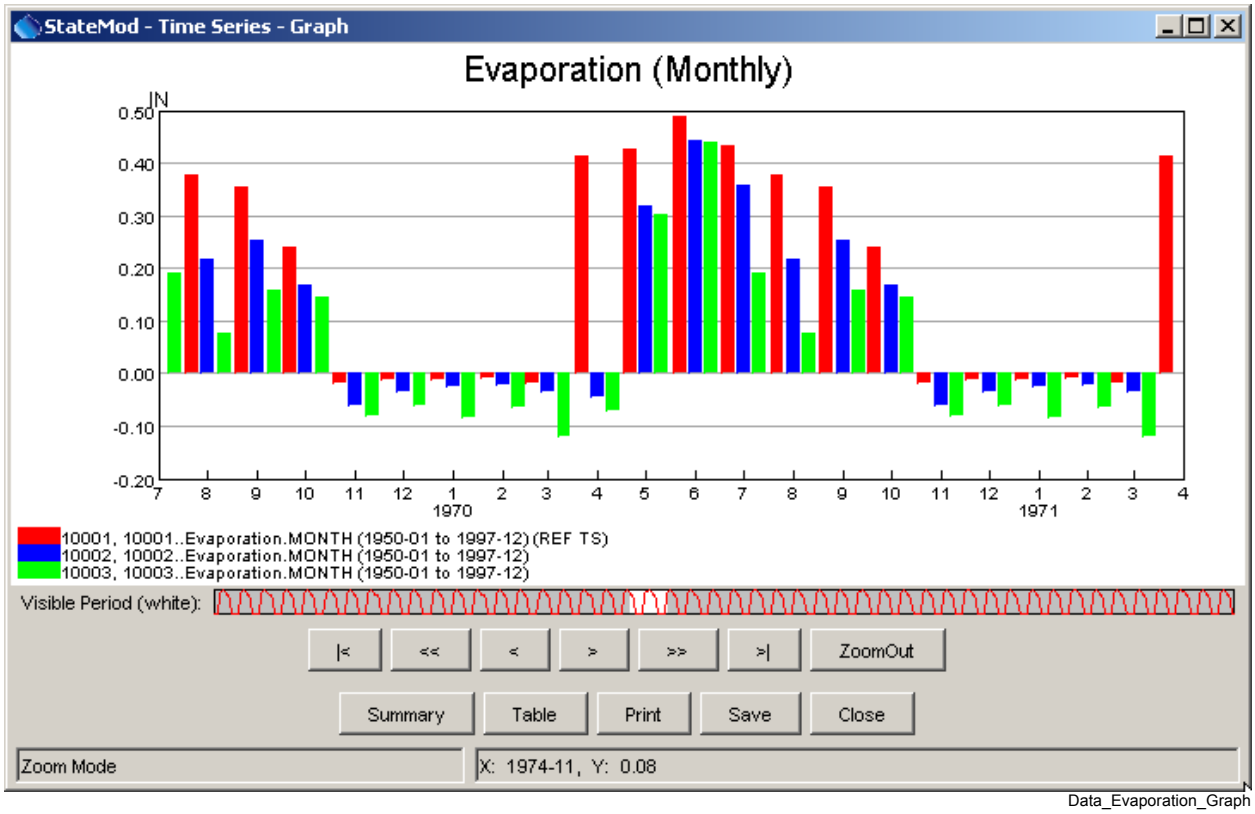
The ***Data…Evaporation…*** menu displays evaporation data. Evaporation data consist of evaporation time series and are used to compute net reservoir evaporation (based on reservoir surface area). Frequently, pan evaporation and precipitation time series are analyzed external to StateMod to produce a net evaporation time series (e.g., see the TSTool software). In this case, no precipitation time series will be provided to StateMod and only the evaporation time series will be used. The identifiers for the climate

stations in the time series files are referenced in reservoir station data. The StateMod GUI currently displays all monthly evaporation time series in a table (average annual evaporation currently cannot be displayed), from which a graph can be displayed. Enhancements are being considered to facilitate selecting specific time series for graphs.




Data_Evaporation_Graph

**Evaporation Time Series Table**

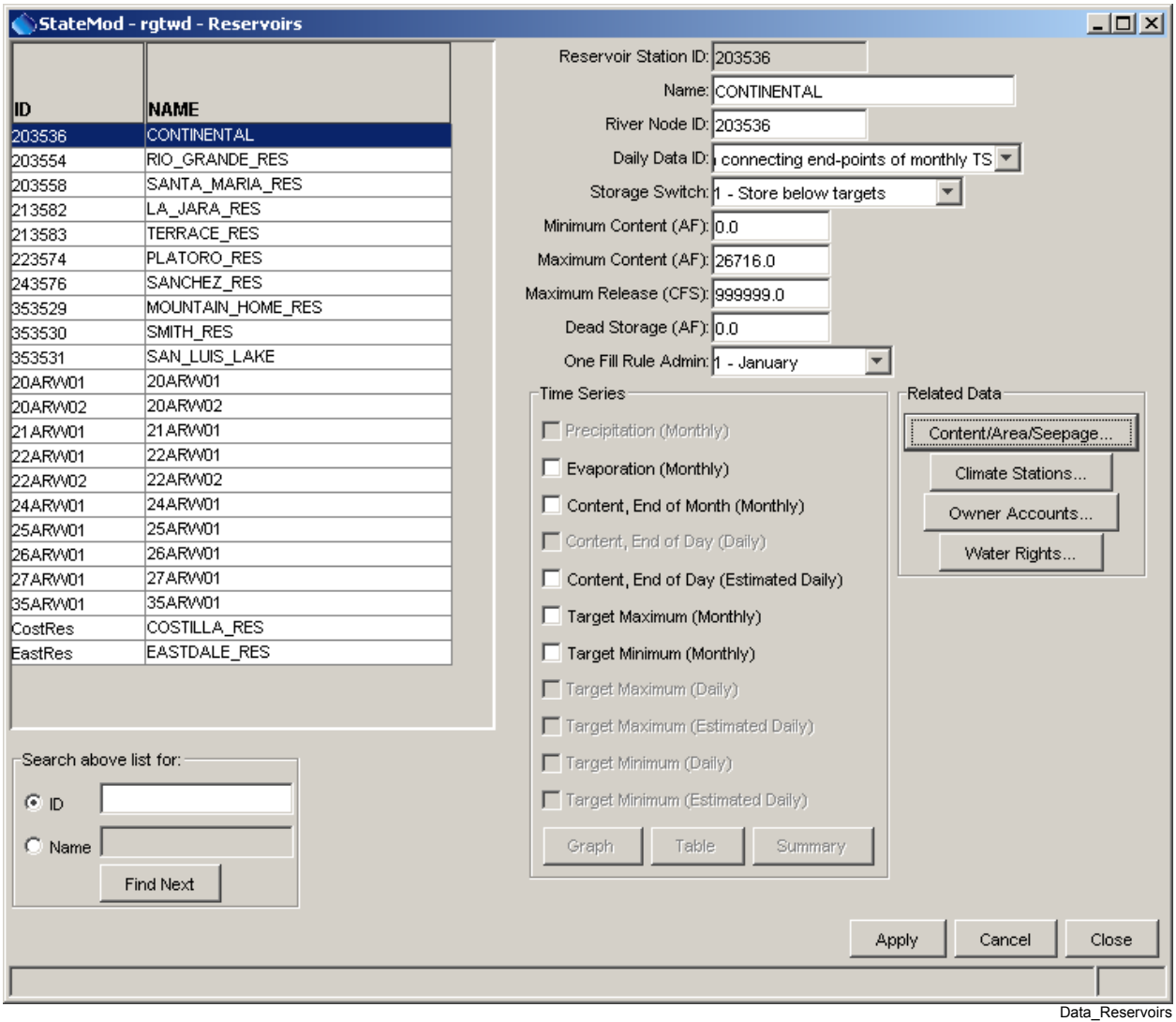



Data_Evaporation_Graph

**Monthly Evaporation Time Series Graph**

## 5.8 Reservoir Data

The ***Data…Reservoirs*** menu displays information for reservoirs. The primary data component is reservoir stations, and secondary components are water rights, historical time series and target time series.




Data_Reservoirs

**Reservoir Data**

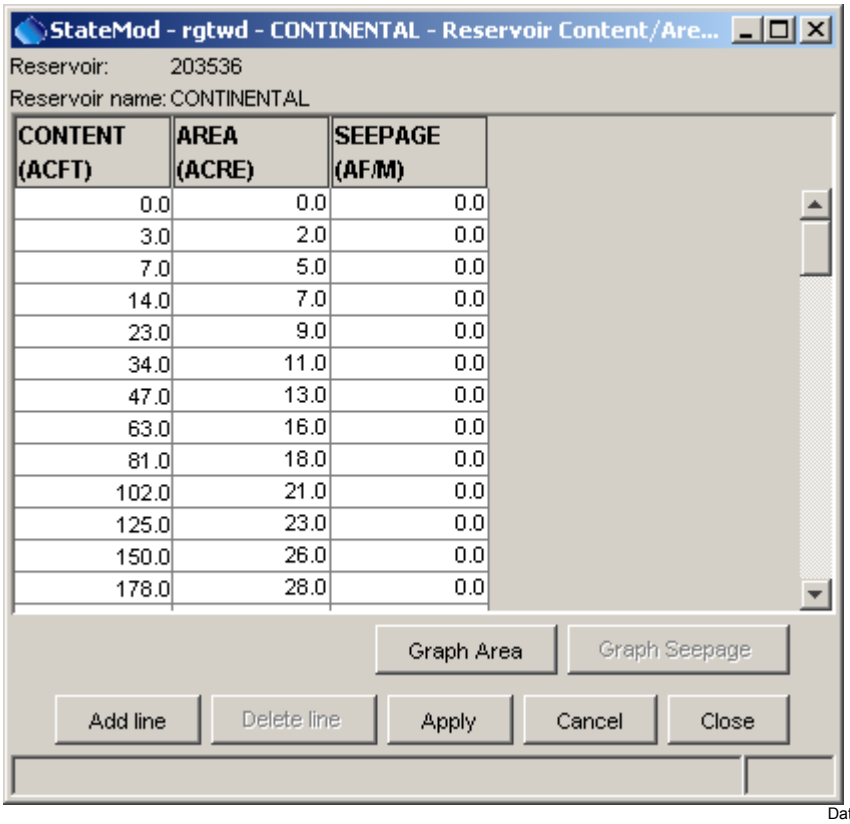
All reservoir stations in the data set are listed on the left side of the window. Selecting a reservoir from the list displays that reservoir station's information in the window. Secondary data are displayed in additional windows accessed via buttons. Access to other data is disabled if the data were not read or are not a part of the data set. All data are editable except for the identifier, which is referenced in the network and other data. Press the ***Apply*** button after making changes. Changes are also applied if the ***Close*** button is pressed. Use the main ***Edit*** menu to add or delete reservoir stations and optionally the secondary data.

To search for a particular station, enter the identifier or name in the appropriate search boxes located below the list. Any number of characters can be entered in the search box. Press ***Enter*** to perform the

case-insensitive search, starting at the top of the list. The ***Find Next*** button, when pressed, will find the next station that matches the information.

### 5.8.1 Reservoir Content/Area/Seepage

Selecting the ***Content/Area/Seepage*** button in the main reservoirs window displays the reservoir's content/area/seepage curve information, as shown below:

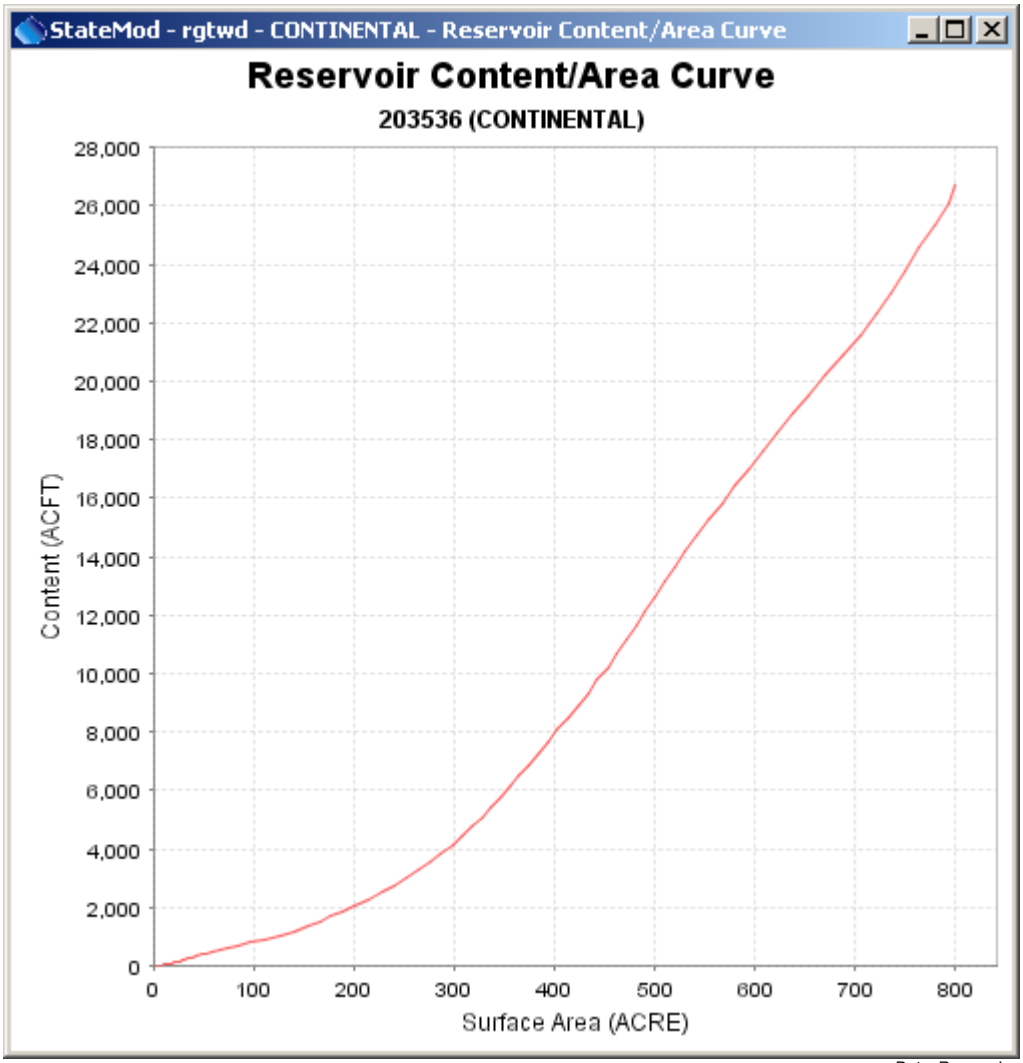



Data_ReservoirsArea_Content

**Reservoir Area Capacity Content Table**

The ***Add line*** button adds a row at the bottom of the table. The ***Delete line*** button will delete the selected row. Information should be added to the table in ascending order. Zero and high point values should be included to bound interpolations. For example, include a very large content value with an area and seepage value that are only slightly larger than the next to last row. Select ***Apply*** to apply the changes to the reservoir data. The ***Close*** button will apply the changes and close the window.

If a range of data values is available, the data can be graphed. Seepage values are often not supplied and therefore the graph button is disabled. The following figure illustrates a reservoir content versus area graph displayed with the ***Graph Area*** button:



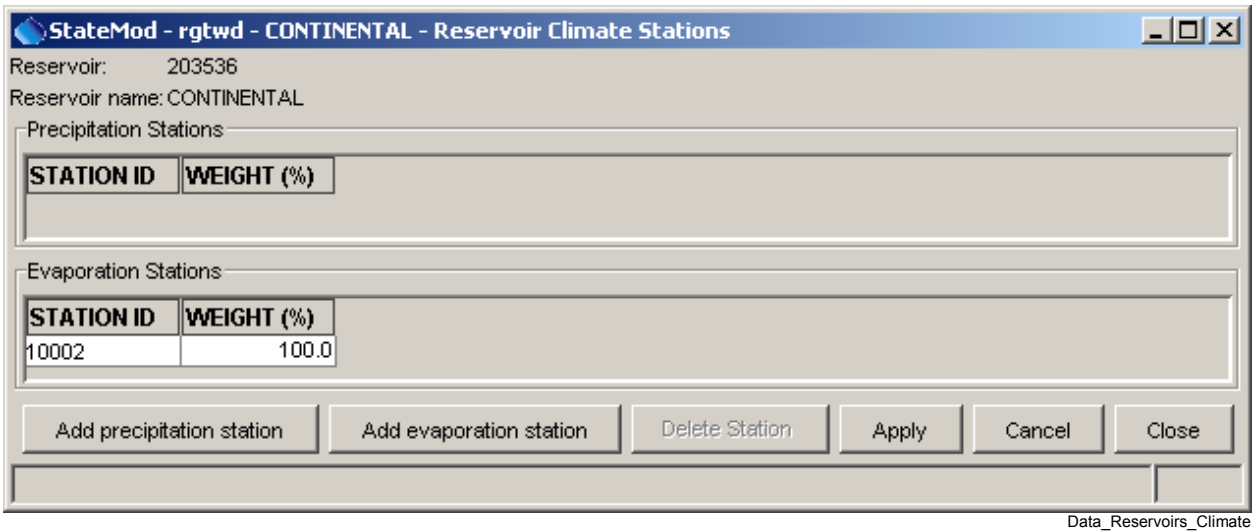

Data_Reservoirs_Content_Graph

**Reservoir Content/Area Graph**

In the above graph, the very large bounding content value is omitted. Right click to print or save the graph to an image. When printing, select the page layout in the first dialog and the printer in the second dialog. Drawing a box on the window will also zoom the graph. Drag the mouse vertically to zoom out.

### 5.8.2 Reservoir Climate Station Assignment

Selecting the ***Climate Stations*** button in the main reservoirs window displays the climate stations that supply precipitation and evaporation time series for the reservoir, as shown below:




Data_Reservoirs_Climate

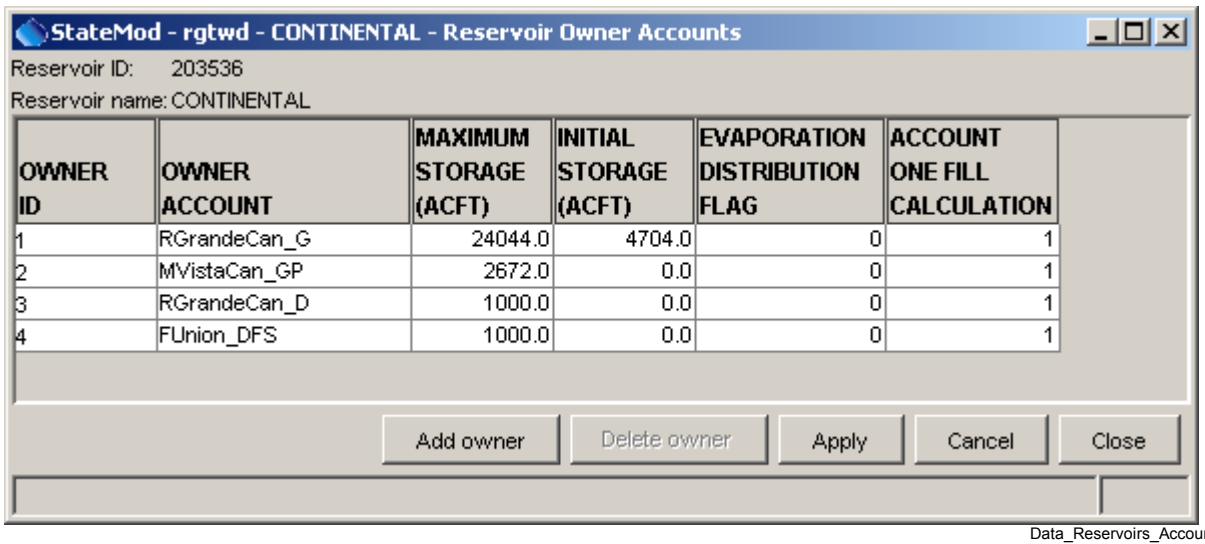
**Reservoir Climate Station Assignments**

The climate station identifiers correspond to the identifiers in the precipitation and evaporation time series. The above example illustrates that no precipitation time series are used and therefore the evaporation time series contain net evaporation data.

The ***Add precipitation station*** and ***Add evaporation station*** buttons will add a row at the bottom of the appropriate table. The ***Delete Station*** button will delete the selected row. Select ***Apply*** to apply the changes to the reservoir data. The ***Close*** button will apply the changes and close the window.

### 5.8.3 Reservoir Accounts

Selecting the ***Owner Accounts*** button in the main reservoirs window displays the accounts associated with the reservoir, as shown in the following figure.



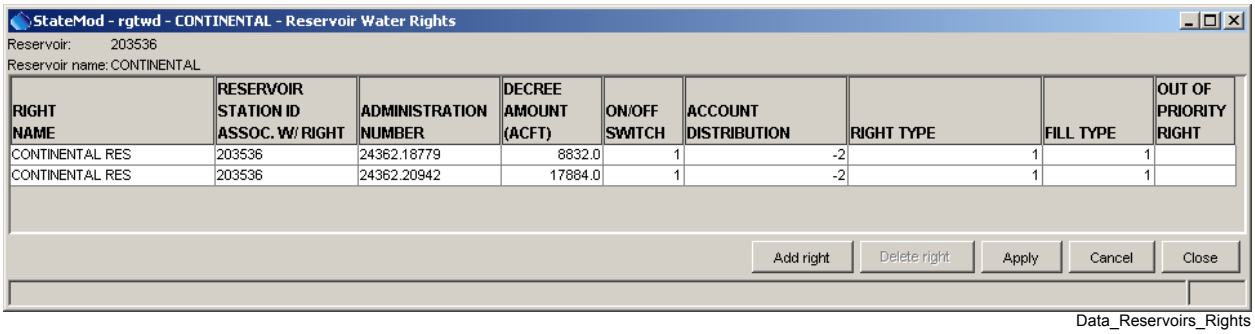

Data_Reservoirs_Accounts

**Reservoir Owner Accounts Data Window**

The ***Add owner*** button will add a row at the bottom of the table. The ***Delete owner*** button will delete the selected row. Select ***Apply*** to apply the changes to the reservoir data. The ***Close*** button will apply the changes and close the window.

### 5.8.4 Reservoir Water Rights

Selecting the ***Water Rights*** button in the main reservoirs window displays the water rights associated with the reservoir, as shown in the following figure:



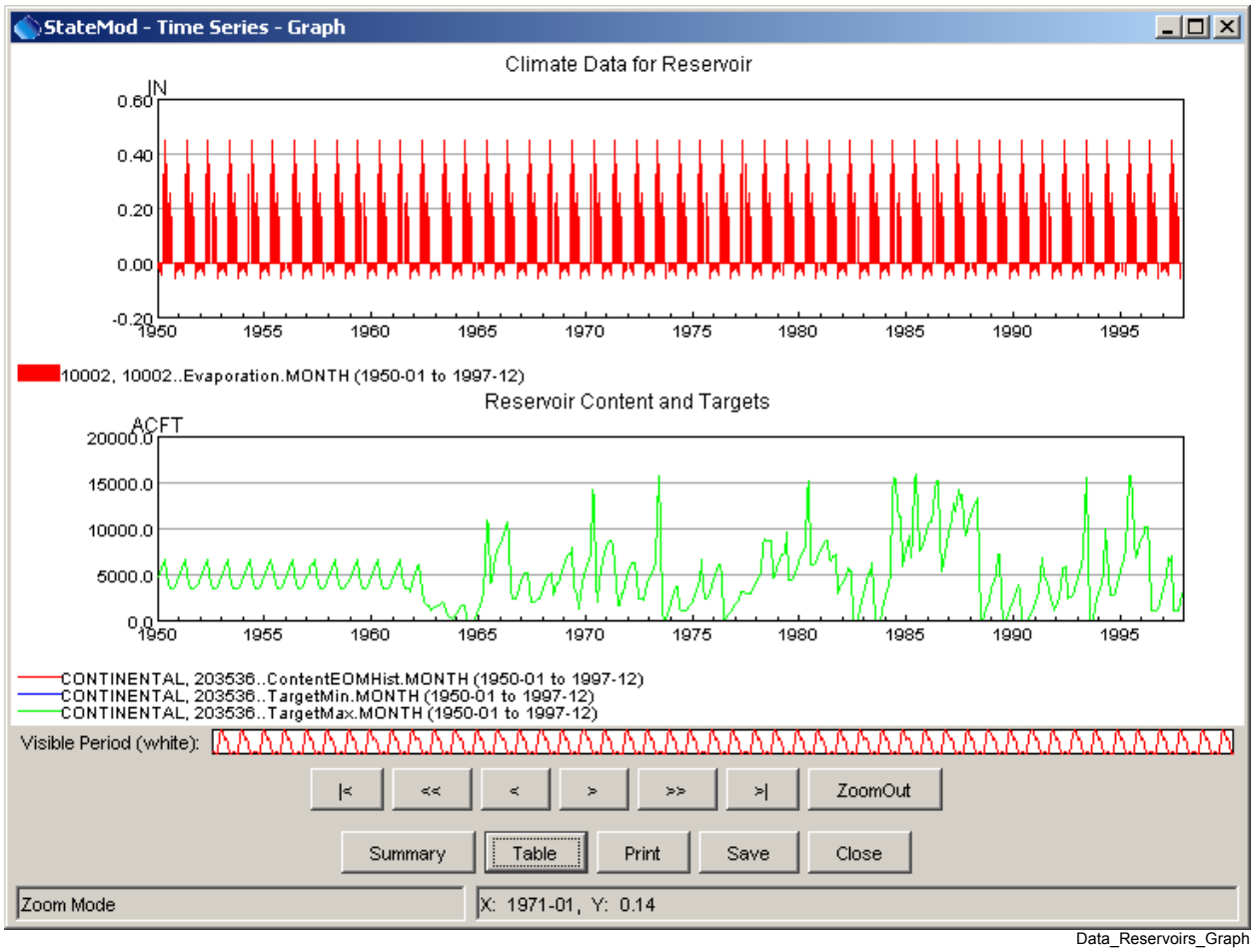

Data_Reservoirs_Rights

**Reservoir Water Rights**

The ***Add right*** button will add a row at the bottom of the table, using default values that should be changed to appropriate values. The ***Delete right*** button will delete the selected row. Select ***Apply*** to apply the changes to the reservoir data. The ***Close*** button will apply the changes and close the window.

### 5.8.5 Reservoir Time Series

The bottom of the main reservoirs window lists all time series associated with reservoir stations. The checkboxes next to time series are enabled according to the data that are available for the currently selected reservoir station. To view time series, select one or more time series and press the ***Graph***, ***Table***, or ***Summary*** buttons. The following figure illustrates a graph of monthly data:



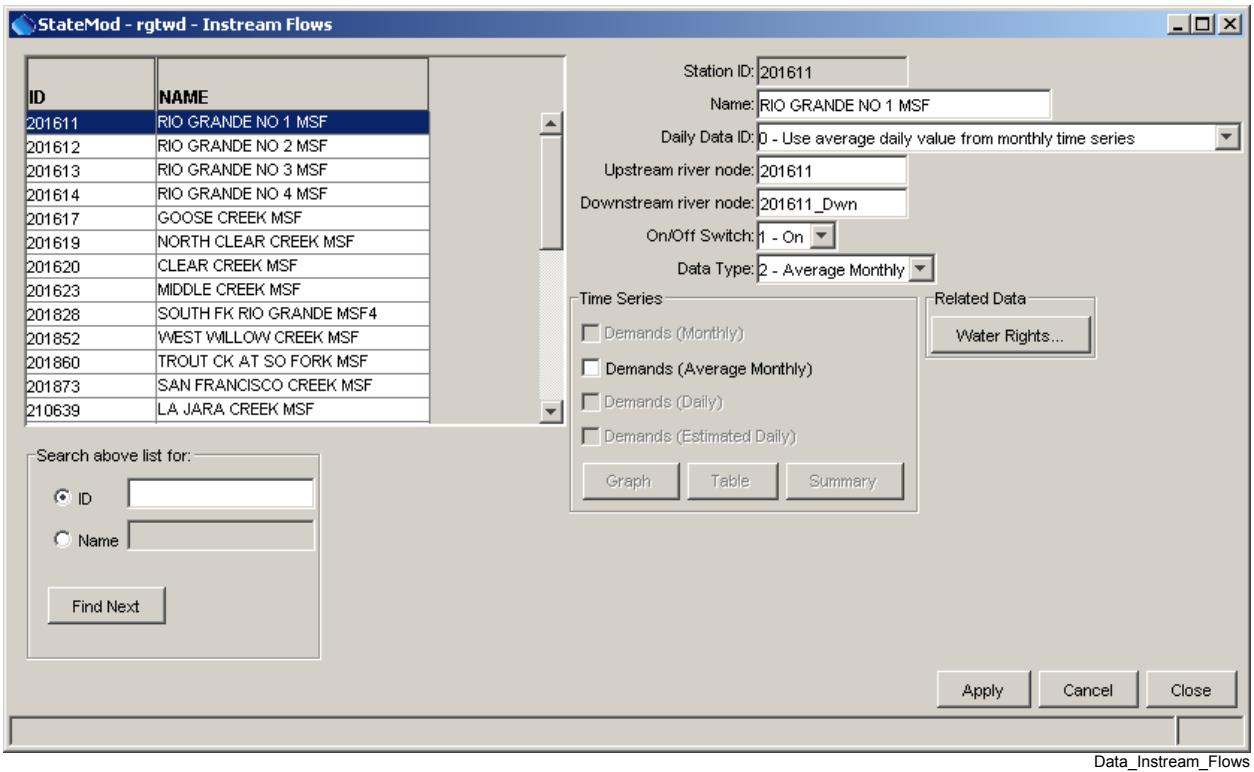

Data_Reservoirs_Graph

**Reservoir Time Series Graph (Monthly)**

The data types for the time series are consistent with the nomenclature used in the ***Results…Graphing Tool*** menu, where possible. The ***Summary*** button can be used to display a text summary of the time series. The ***Table*** button can be used to view the time series in tabular form. Refer to the **TSView Time Series Viewing Tools Appendix** for more information about the graphing tools.

## 5.9 Instream Flow Data

The ***Data…Instream Flows*** menu displays information for instream flow reaches. The primary data component is instream flow stations, and secondary components are water rights and demand time series.



Data_Instream_Flows

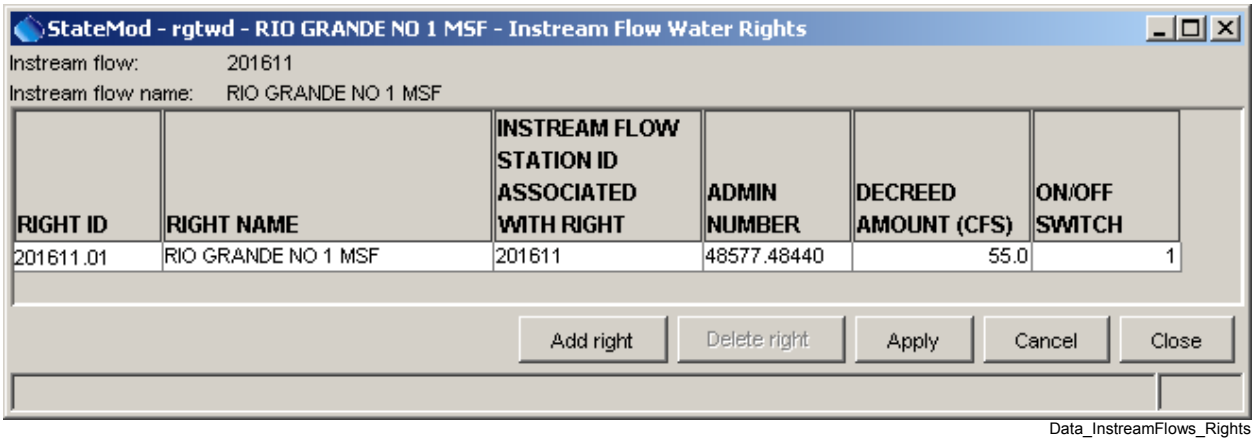
**Instream Flows Data**

All instream flow stations in the data set are listed on the left side of the window. Selecting an instream flow from the list displays that instream flow's information in the window. The instream flow station information is displayed in the main instream flows window and the remaining data are displayed in secondary windows accessed via buttons. Access to other data is disabled if the data were not read or are not a part of the data set. All data are editable except for the identifier, which are referenced in the network and other data. Press the ***Apply*** button after making changes. Changes are also applied if the ***Close*** button is pressed. Use the main ***Edit*** menu to add or delete instream flow stations and optionally the secondary data.

To search for a particular station, enter the identifier or name in the appropriate search boxes located below the list. Any number of characters can be entered in the search box. Press ***Enter*** to perform the case-insensitive search, starting at the top of the list. The ***Find Next*** button, when pressed, will find the next station that matches the information.

### 5.9.1 Instream Flow Rights

Selecting the ***Water Rights*** button in the main instream flows window displays the water rights associated with the instream flow station, as shown in the following figure:



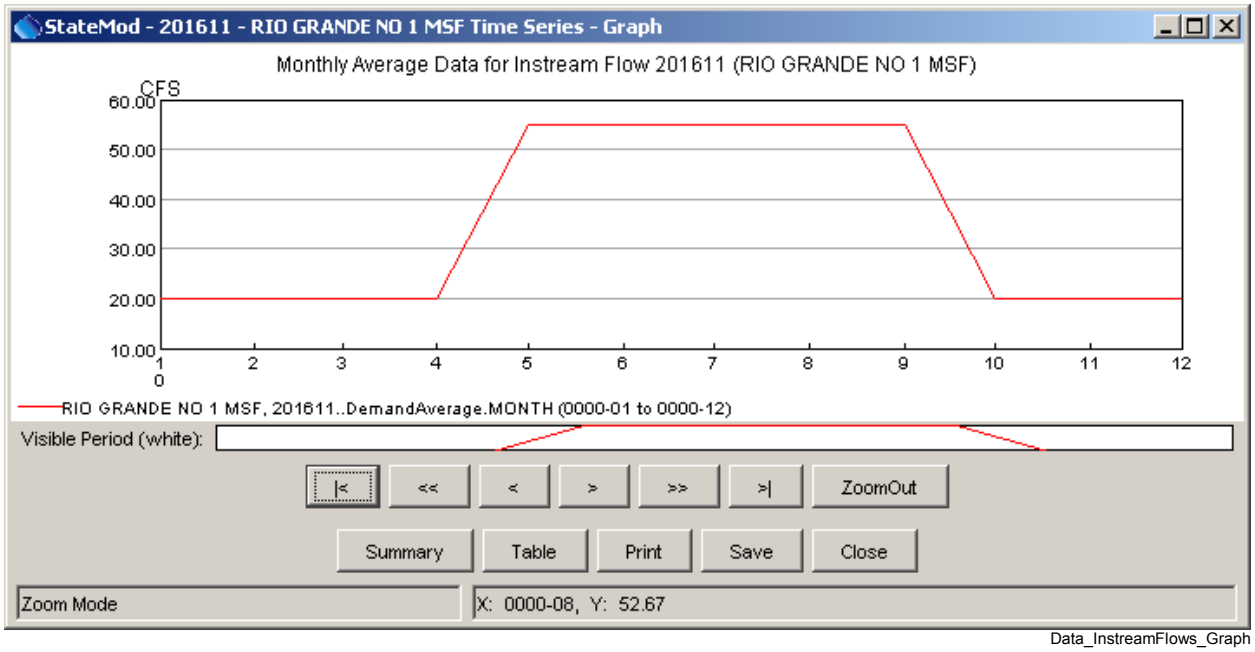

Data_InstreamFlows_Rights

**Instream Flow Water Rights**

The ***Add right*** button will add a row at the bottom of the table, using default values that should be changed to appropriate values. The ***Delete right*** button will delete the selected row. Select ***Apply*** to apply the changes to the reservoir data. The ***Close*** button will apply the changes and close the window.

### 5.9.2 Instream Flow Time Series

The bottom of the main instream flows window lists all time series associated with instream flow stations. The checkboxes next to time series are enabled according to the data that are available for the currently selected instream flow station. To view time series, select one or more time series and press the ***Graph***, ***Table***, or ***Summary*** buttons. The following figure illustrates a graph of average monthly demand data:



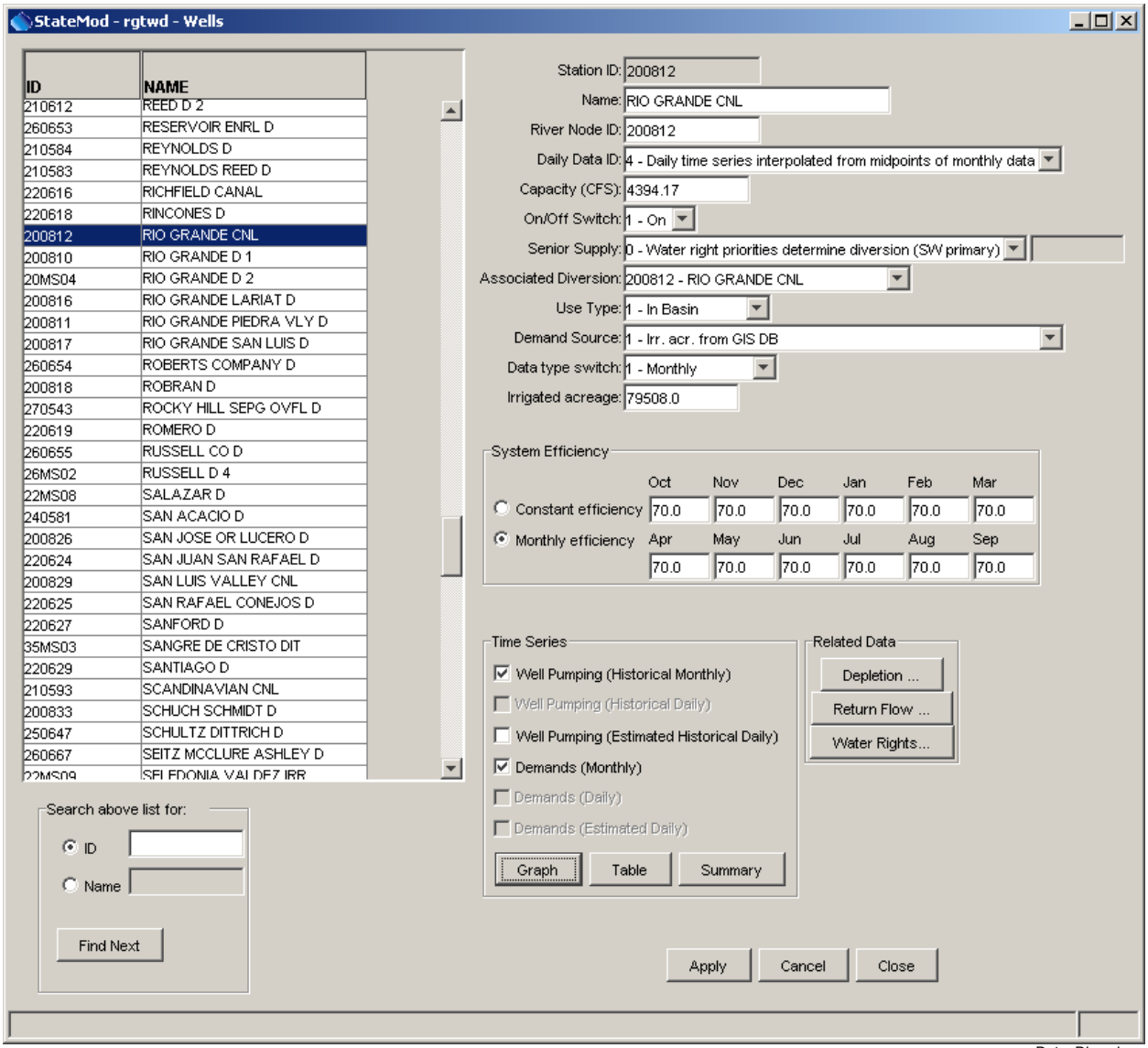

Data_InstreamFlows_Graph

**Instream Flow Average Monthly Demand Graph**

The data types for the time series are consistent with the nomenclature used in the ***Results…Graphing Tool*** menu, where possible. The ***Summary*** button can be used to display a text summary of the time series. The ***Table*** button can be used to view the time series in tabular form. Refer to the **TSView Time Series Viewing Tools Appendix** for more information about the graphing tools.

## 5.10 Well Data

The ***Data…Wells*** menu displays information for wells. The primary data component is well stations, and secondary components are water rights, historical time series, demand time series, depletions, and return flows (delay table assignments). A well station can be a single physical well, or a group of wells treated as a single well for modeling purposes. Additionally, wells can be associated with a diversion station, in which case the identifier for the well typically matches the diversion station and the ***Associated Diversion*** is specified.



Data_Diversions

**Diversion Data Window**

All well stations in the data set are listed on the left side of the window. Selecting a well from the list displays that well's information in the window. The well station information is displayed in the main wells window and the remaining data are displayed in secondary windows accessed via buttons. Access to other data is disabled if the data were not read or are not a part of the data set. All data are editable except for the identifier, which is referenced in the network and other data. Press the ***Apply*** button after making changes. Changes are also applied if the ***Close*** button is pressed.

Use the main ***Edit*** menu to add or delete well stations and optionally the secondary data.

To search for a particular station, enter the identifier or name in the appropriate search boxes located directly below the list. Any number of characters can be entered in the search box. Press ***Enter*** to perform the case-insensitive search, starting at the top of the list. The ***Find Next*** button, when pressed, will find the next station that matches the information.

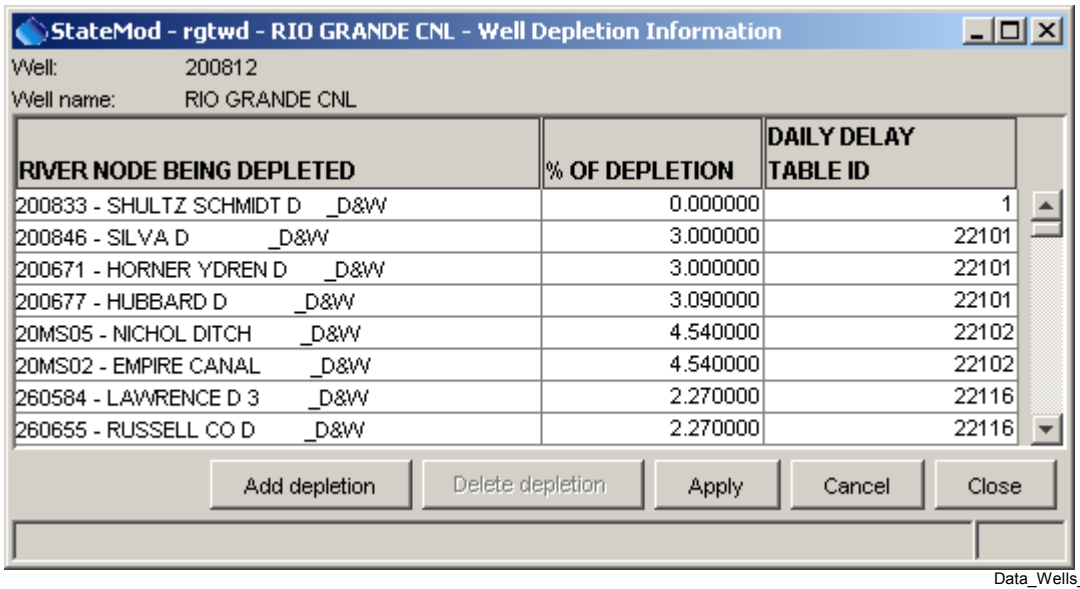
The well efficiency is displayed for each month in the year. If the well has a constant efficiency, the same value is displayed in each of the twelve monthly fields. If variable efficiencies are used in modeling they are not reflected in this display (only the average efficiencies shown in the well station file are shown).

The ***Daily Data ID*** is used to associate a daily time series with this well. Changes to this value take effect when either a different well in the list is chosen or when ***Enter*** is pressed in the ***Daily Data ID*** text field. The following options are available:

- If the ***Daily Data ID*** exactly matches the well ***Station ID***, the pattern and values are the same.
- If the ***Daily Data ID*** is "0", the pattern and values are again the same but are the average daily values, calculated using the monthly time series.
- If the ***Daily Data ID*** does not match the well ***Station ID*** and is not "0", the pattern time series corresponding to the ***Daily Data ID*** is displayed as-is. However, the values time series is calculated using the pattern and known monthly totals. The daily time series monthly total should agree with the monthly time series, but the distribution should correspond with the pattern. More than one well may reference the same pattern. For that reason, care should be taken when changing the pattern itself.
- Additional options may be available from StateMod. Refer to the StateMod software documentation.

### 5.10.1 Well Station Depletions

Well station depletions are displayed by pressing the ***Depletion*** button in the main wells window:



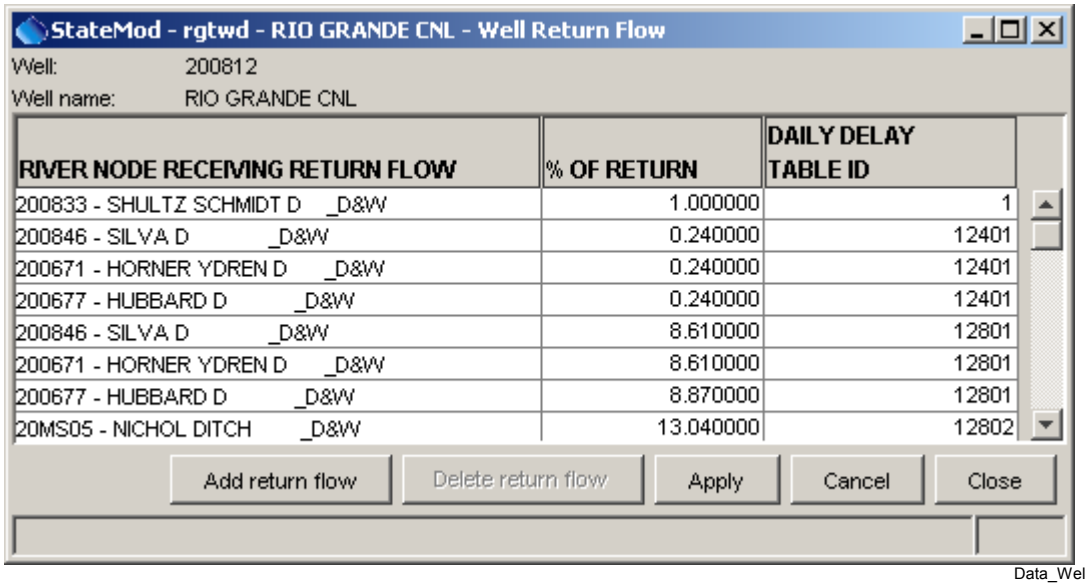

Data_Wells_Depletions

**Well Depletion Data**

Press the ***Add depletion*** button to add a row to the display. Fill in the information as appropriate, using the choices that are provided. To delete a return flow, select a row and press the ***Delete depletion*** button. Select ***Apply*** to apply the changes to the well data. The ***Close*** button will apply the changes and close the window.

### 5.10.2 Well Station Return Flows

Well station return flows (delay table assignments) are displayed by pressing the ***Return Flow*** button in the main wells window:



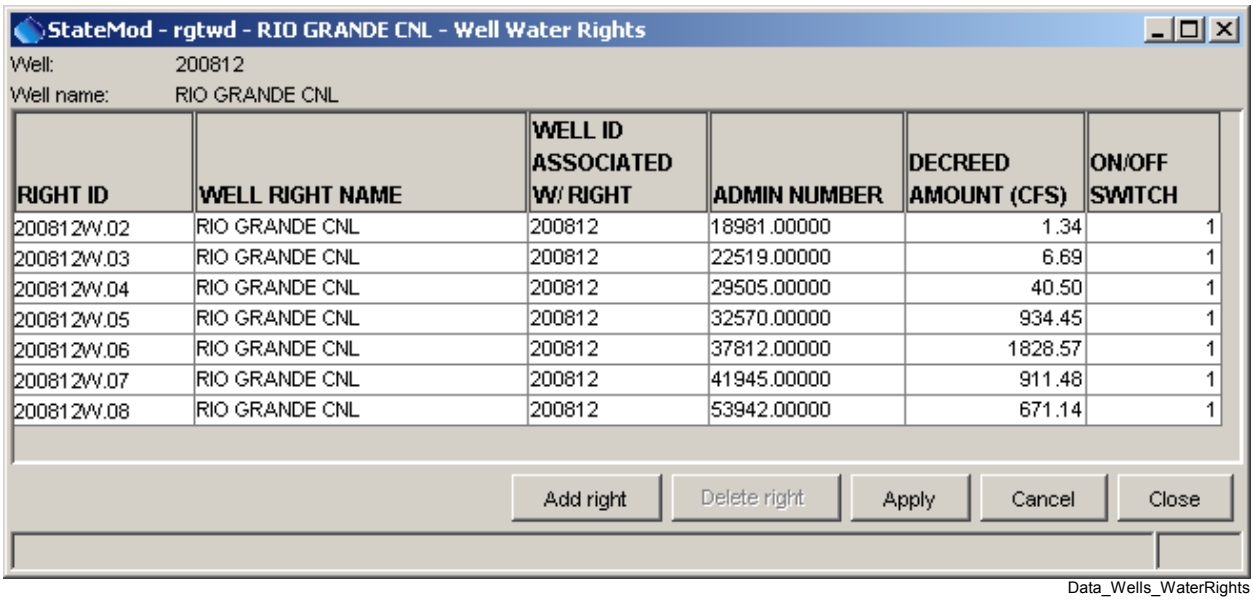

Data_Wells_Returns

**Well Return Flow Data**

Press the ***Add return flow*** button to add a row to the display. Fill in the information as appropriate, using the choices that are provided. To delete a return flow, select a row and press the ***Delete return flow*** button. Select ***Apply*** to apply the changes to the well data. The ***Close*** button will apply the changes and close the window.

### 5.10.3 Well Water Rights

The water rights that apply to the well can be viewed by pressing the ***Water Rights*** button in the wells data window, resulting in a display as shown in the following figure:



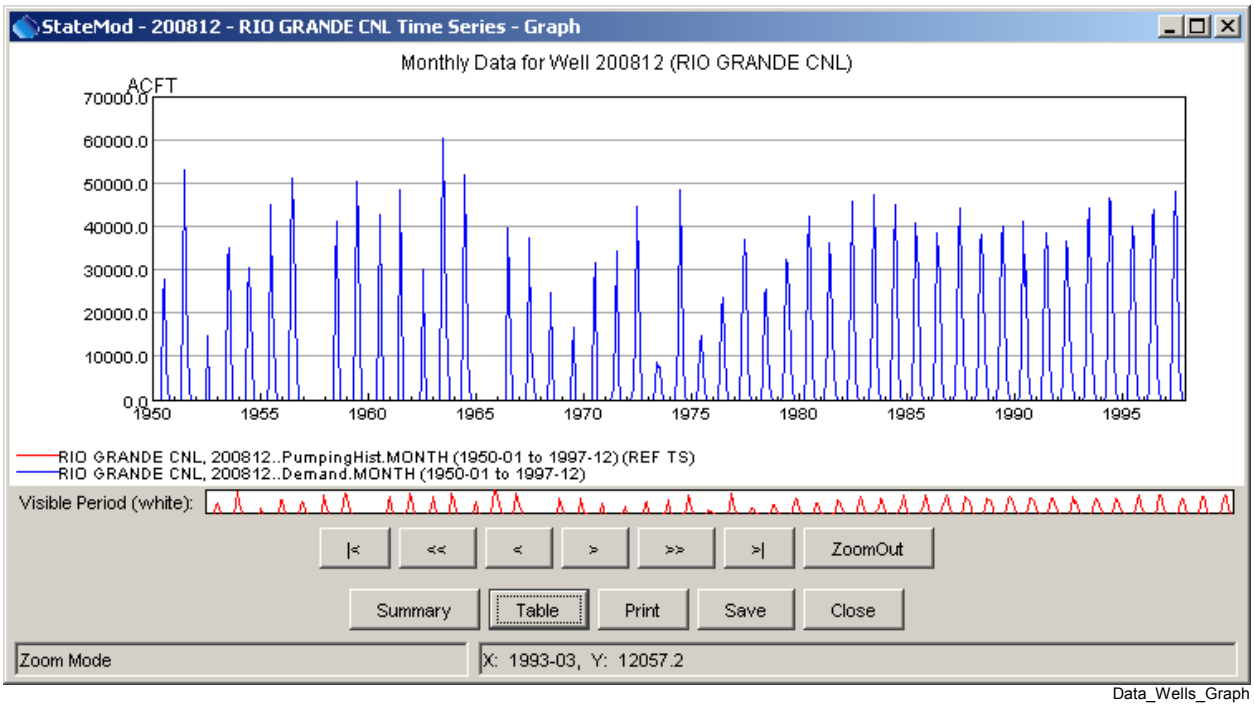

Data_Wells_WaterRights

**Well Water Rights Data Window**

To add a water right, press ***Add right***. This will add a row with default information, which should be updated as appropriate. To delete a water right, select the row and press ***Delete right***. Select ***Apply*** to apply the changes to the well data. The ***Close*** button will apply the changes and close the window.

### 5.10.4 Well Time Series

The bottom of the main wells window lists all time series associated with well stations. The checkboxes next to time series are enabled according to the data that are available for the currently selected well station. To view time series, select one or more time series and press the ***Graph***, ***Table***, or ***Summary*** buttons. The following figure illustrates a graph of monthly data:



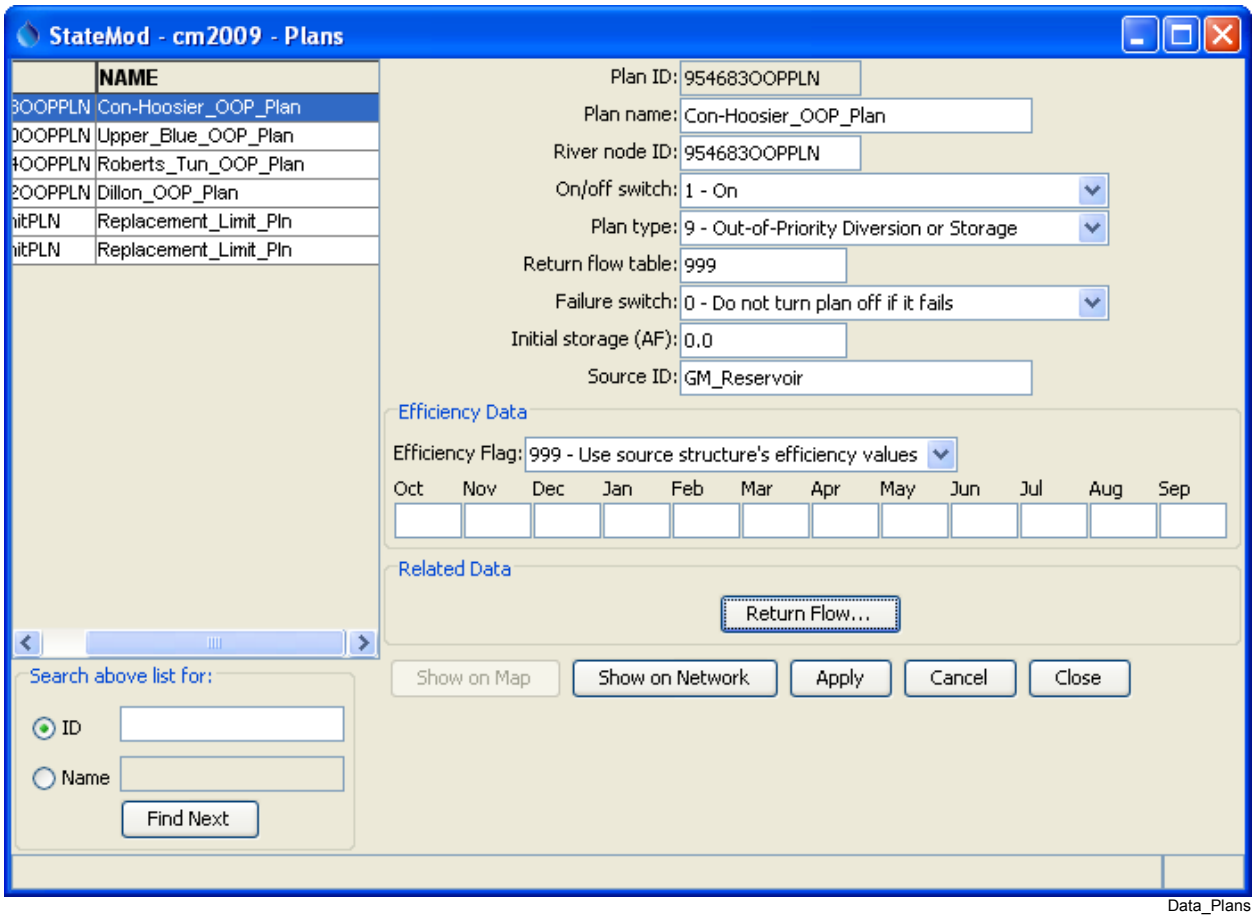

Data_Wells_Graph

**Well Time Series (Monthly)**

The data types for the time series are consistent with the nomenclature used in the ***Results…Graphing Tool*** menu, where possible. The ***Summary*** button can be used to display a text summary of the time series. The ***Table*** button can be used to view the time series in tabular form. Refer to the **TSView Time Series Viewing Tools Appendix** for more information about the graphing tools.

## 5.11 Plan Station Data

The ***Data…Plans*** menu displays a data window for plan stations. Plan stations are representations of system operational features that do not fall into other model node types. For example, an augmentation plan involving recharge can be represented as a plan. See the StateMod documentation for details about plans.



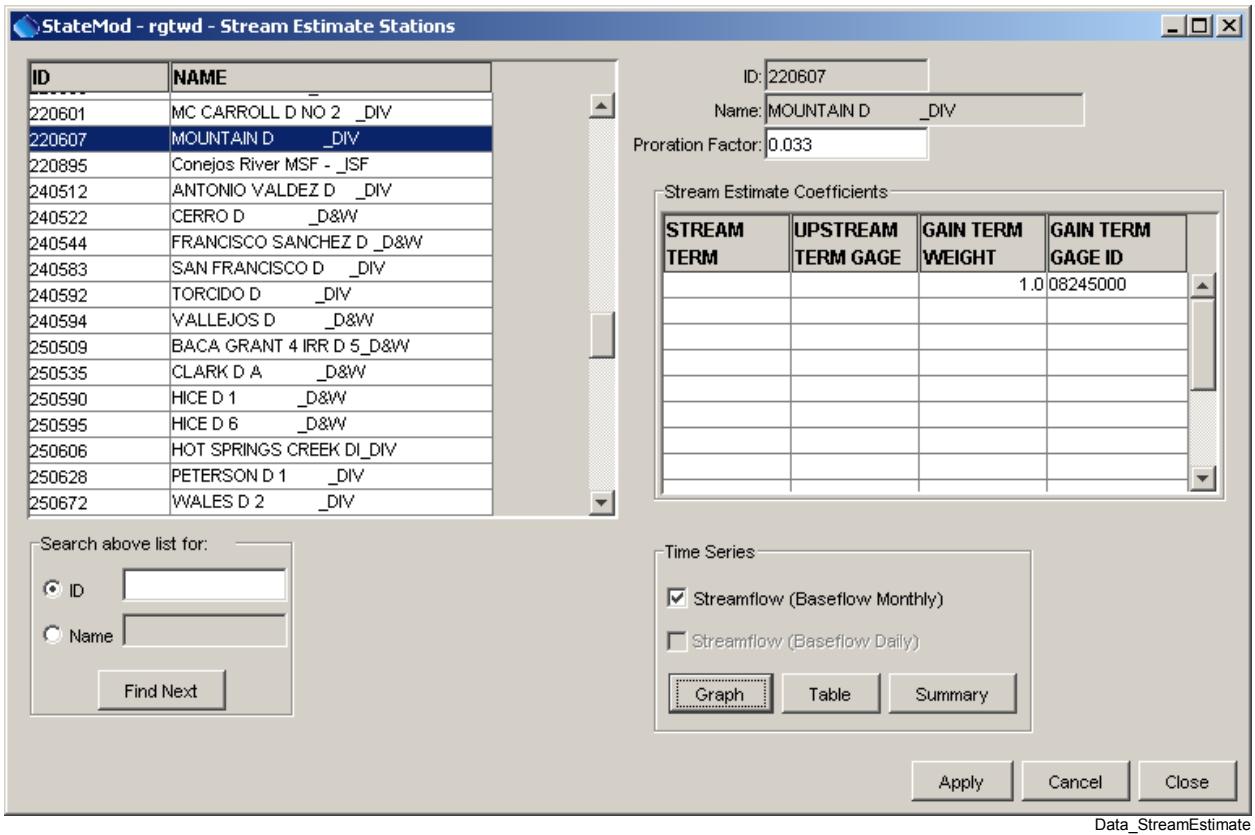
Data_Plans

**Plan Station Data**

The available plan station nodes are shown in the list on the left side of the window. When a node is selected, data associated with the plan are shown on the right side of the window.

## 5.12 Stream Estimate Data

The ***Data…Stream Estimate*** menu displays a data window for stream estimate stations. Stream estimate stations are locations where historical streamflow time series are not available (as opposed to stream gage stations discussed in **Section 5.2**). Such points are often needed in headwater basins in order to insert streamflow into the model.



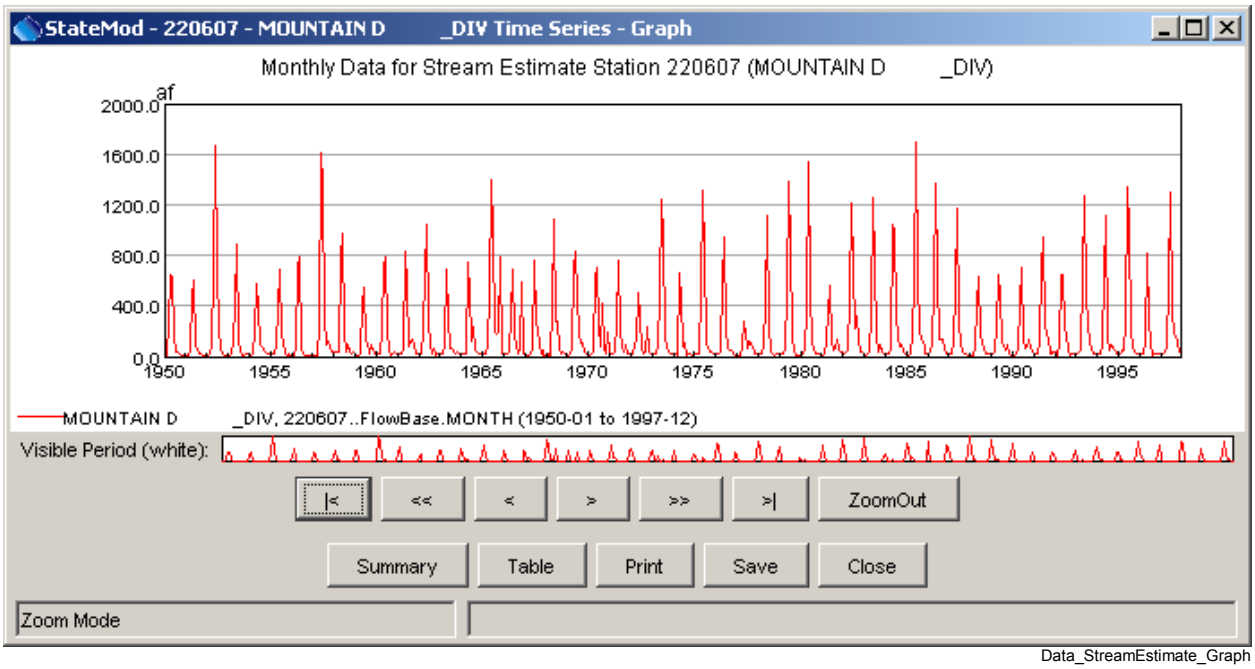
Data_StreamEstimate

**Stream Estimate Data**

The available baseflow nodes are shown in the list on the left side of the window. When a node is selected, the coefficients used to estimate data are shown on the right side of the window. Baseflow coefficient information is not trivial to specify and should be done carefully. See the StateMod software documentation and the StateDMI software documentation for more information.

### 5.12.1 Stream Estimate Time Series

The bottom of the main stream estimate stations window lists all time series associated with stream estimate stations. The checkboxes next to time series are enabled according to the data that are available for the currently selected stream estimate station. To view time series, select one or more time series and press the ***Graph***, ***Table***, or ***Summary*** buttons. The following figure illustrates a graph of monthly data:



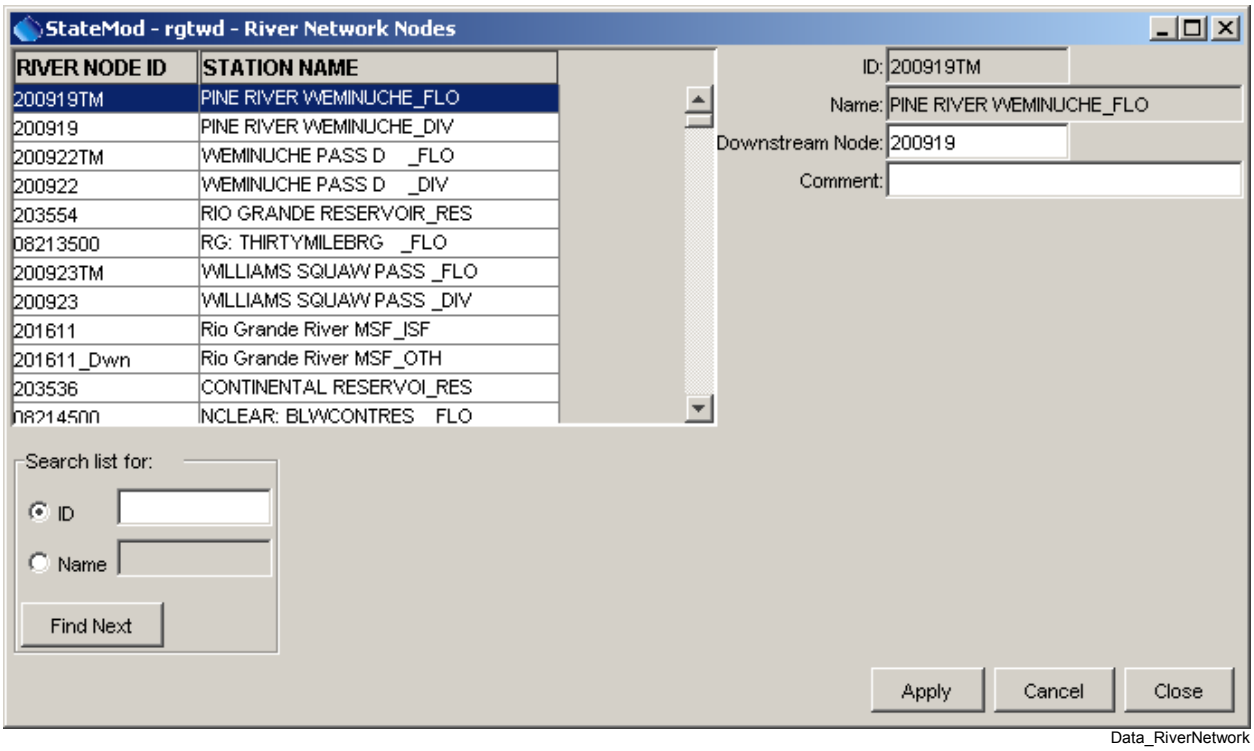

Data_StreamEstimate_Graph

**Stream Estimate Time Series (Monthly)**

The data types for the time series are consistent with the nomenclature used in the ***Results…Graphing Tool*** menu, where possible. The ***Summary*** button can be used to display a text summary of the time series. The ***Table*** button can be used to view the time series in tabular form. Refer to the **TSView Time Series Viewing Tools Appendix** for more information about the graphing tools.

## 5.13 River Network Data

The ***Data…River Network*** menu displays the river network data:




**River Network Data**

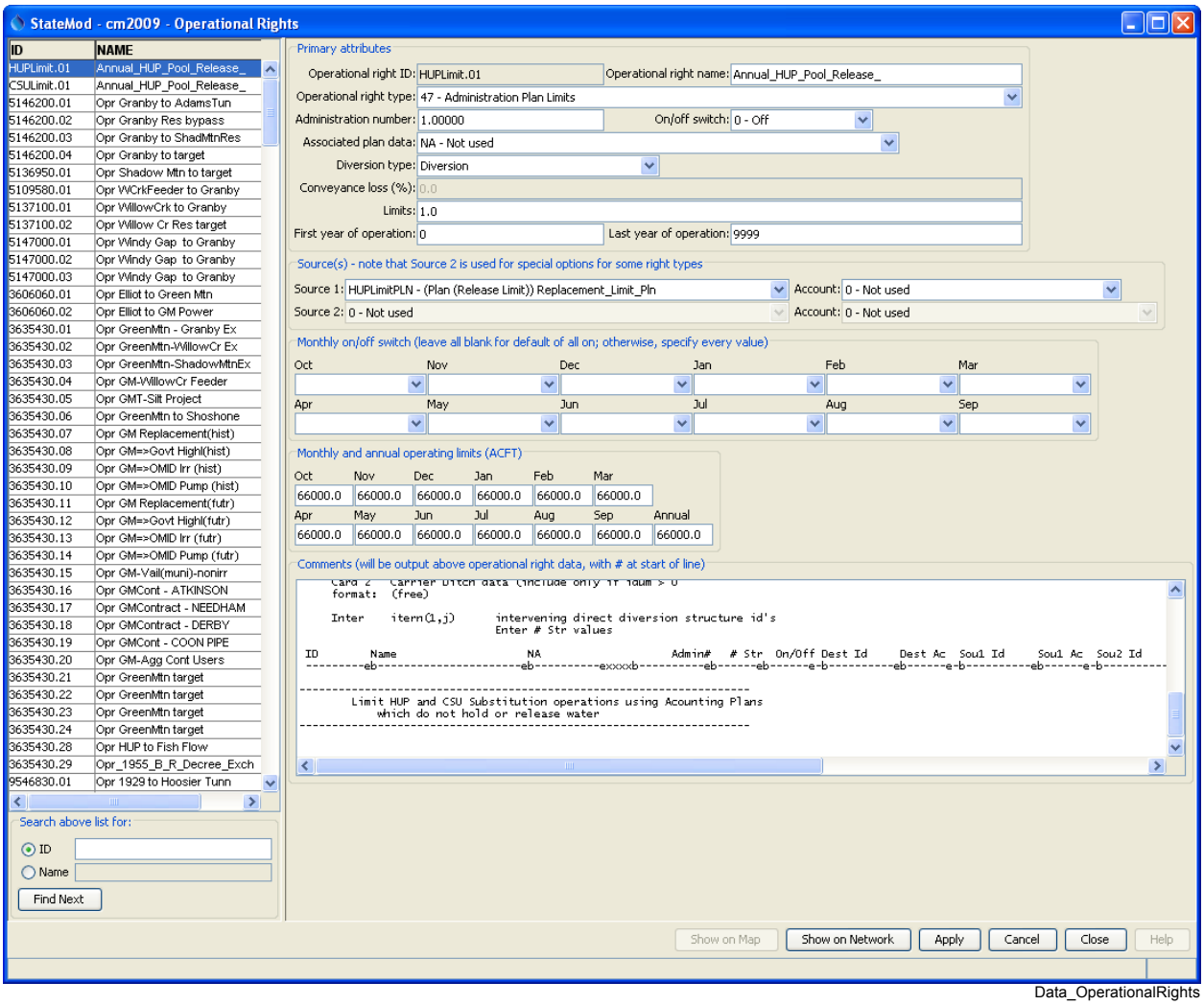
The river network data describes the connectivity of the river network, in particular by indicating the downstream station for each station in the network. In general, it should not be edited because doing so may break data links in memory (e.g., changing a river node identifier in the river network requires that the node be changed in any other data that uses the old identifier; the StateMod GUI will not do this automatically). Instead, nodes should be added and deleted using the main ***Edit*** menu. StateMod reports results at river nodes and it may be appropriate to add a river node that is not in any other station list. It is reasonable to edit the description in the river network window.

To search for a particular station, enter the identifier or name in the appropriate search boxes located directly below the list. Any number of characters can be entered in the search box. Press ***Enter*** to perform the case-insensitive search, starting at the top of the list. The ***Find Next*** button, when pressed, will find the next station that matches the information.

If modeling in Colorado's Decision Support Systems (CDSS), the StateMod network file is typically created using the StateDMI software. Other files are created by determining station lists from the network.

## 5.14 Operational Rights Data

The ***Data…Operational Rights*** menu displays operational rights data.




Data_OperationalRights

**Operational Rights Data**

Operational rights data control operating policies for reservoirs and other structures. This file often is edited by hand during initial data set development. The StateMod GUI attempts to display all operational right types; however, there are a number of limitations that will require additional resources to resolve and using the GUI to edit rights is discouraged at this time:

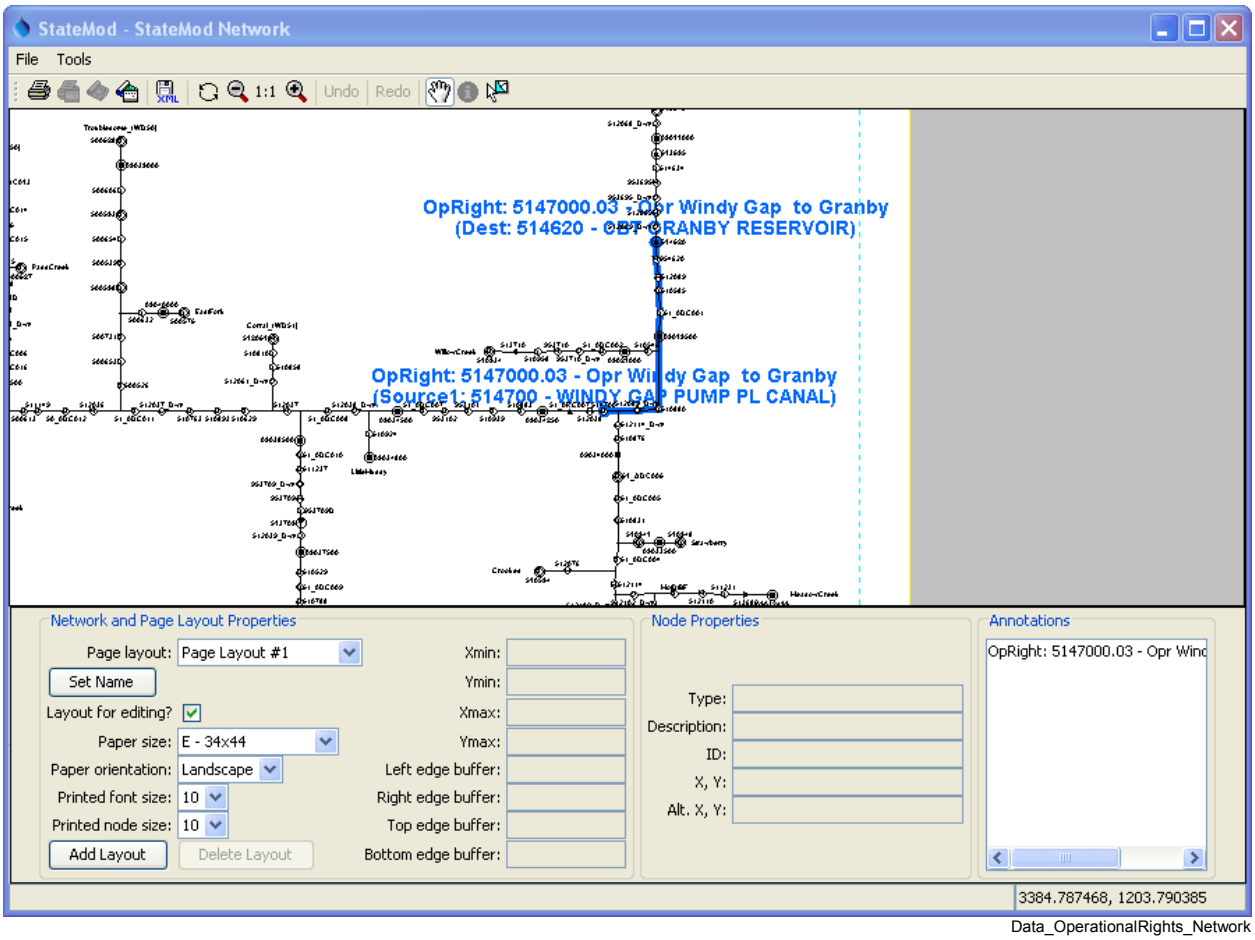
1. The StateMod model software has features that have not been considered by the GUI.
2. The operational rights are complex and difficult to treat generically. The GUI window has been segmented for major operational right data; however, the documentation for operational rights and use in practice needs to be verified against GUI features.
3. Handling of comments in the operational rights file is problematic. Additional standardization on the file format and modeling practices is needed.
4. Right types that are not understood are displayed as text.

Different areas of the window will be enabled, depending on the right type. See the StateMod software documentation for more information about operational rights.

All data are editable except for the identifier and name, which are referenced in other data. Press the ***Apply*** button after making changes. Changes are also applied if the ***Close*** button is pressed. Use the main ***Edit*** menu to add or delete operational rights.

To search for a particular right, enter the identifier or name in the appropriate search boxes located directly below the list. Any number of characters can be entered in the search box. Press ***Enter*** to perform the case-insensitive search, starting at the top of the list. The ***Find Next*** button, when pressed, will find the next station that matches the information.

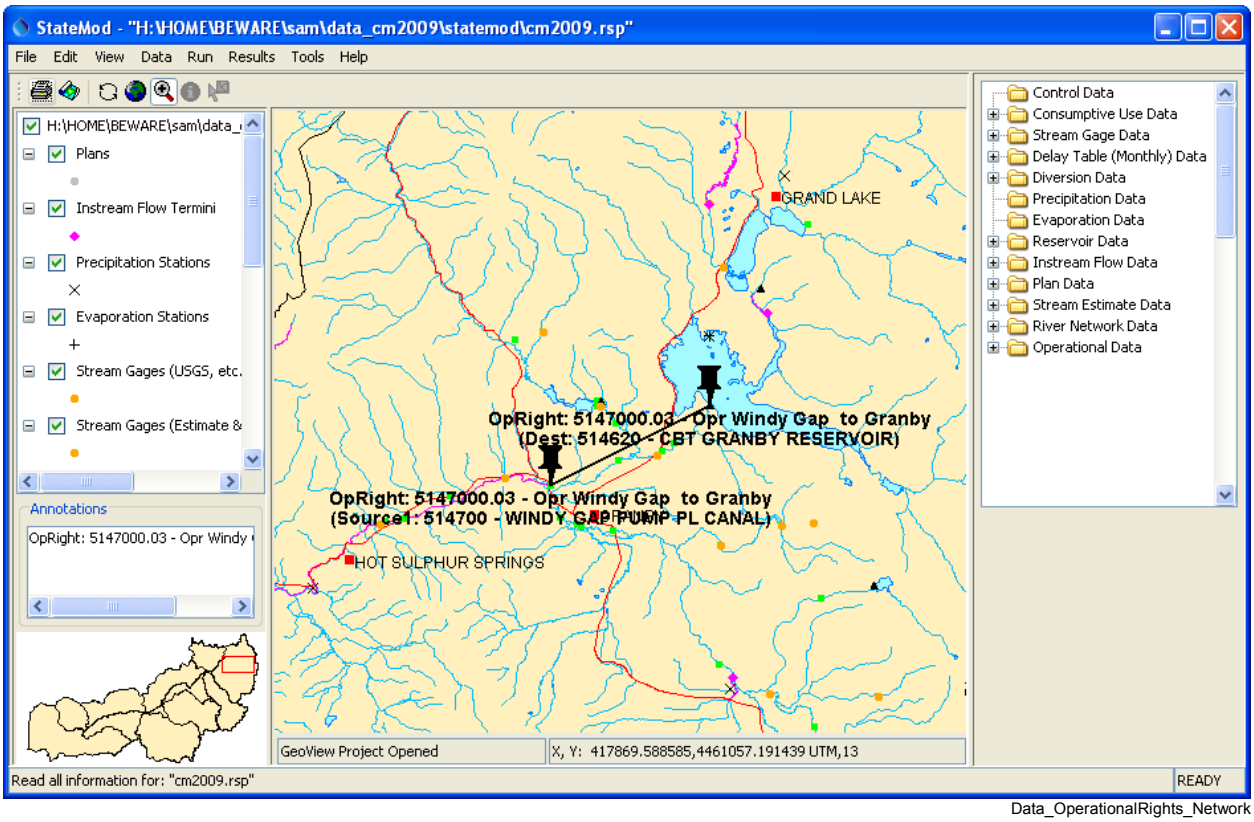
The ***Show on Network*** button is enabled if the operational right source and destination identifiers can be determined. Pressing the button annotates the network with information about the operational right, as shown in the following figure. The intervening flow path is highlighted, although in some cases wet water will not actually flow along the path for the operational right. Right click on the ***Annotations*** area in the lower right to clear the annotations. The annotations will not be saved in the (**.net*) file if the network is saved.




Data_OperationalRights_Network

**Operational Right Annotated on the Network**

The ***Show on Map*** button is enabled if the operational right source and destination location information can be determined. Pressing the button annotates the map with information about the operational right, using push pin symbols. The locations are connected with a straight line for visualization purposes. Right click on the ***Annotations*** area in the lower left to clear the annotations.



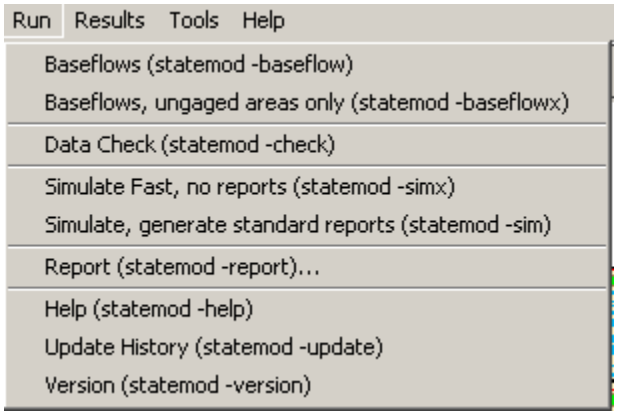

Data_OperationalRights_Network

**Operational Right Annotated on the Map**

# 6 Running StateMod

Version 06.03.02, 2006-03-04, Color, Acrobat Distiller

The ***Run*** menu runs the StateMod model, based on its main run modes, as listed in the menu:




Menu_Run

**Run Menu**

The StateMod GUI runs StateMod by starting a separate StateMod process. StateMod console output is read and displayed it in the GUI. The StateMod exit status is detected by a STOP code printed to the console.

The StateMod GUI automatically provides the StateMod software with appropriate command line parameters in most cases, requiring no further input from users. The exception is that when running in report mode, additional information will be requested to select the report type, as discussed below.

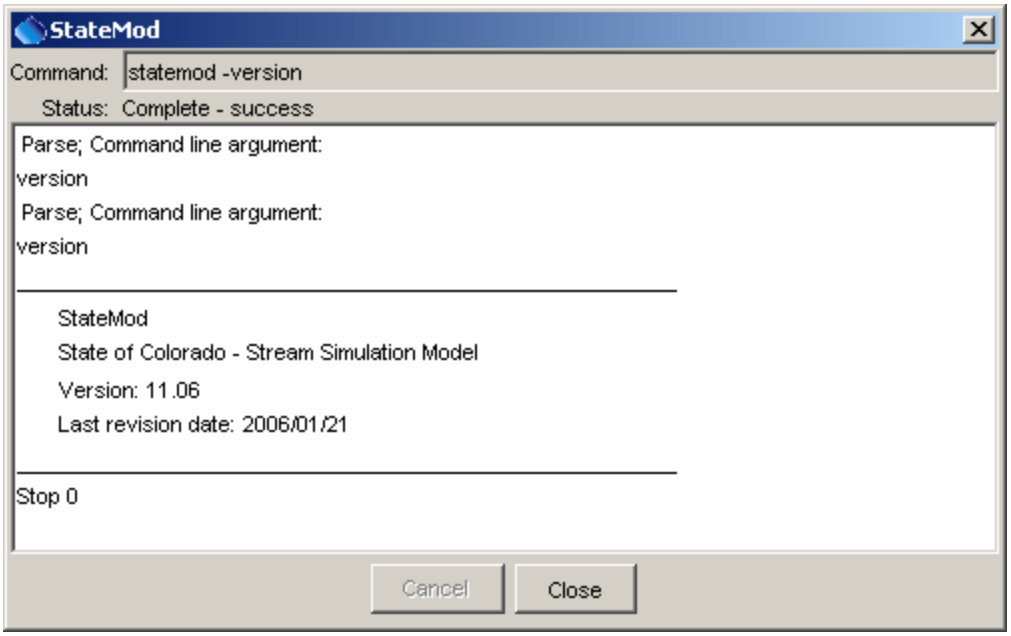
When running simulations that generate output, the StateMod software uses the output request file to limit output. See **Section 5.1.3 – Output Control File** for more information about controlling StateMod output.

## 6.1 Running StateMod in Baseflow, Data Check, Simulation, and Information Modes

Most StateMod run modes only require selecting the proper choice from the ***Run*** menu. Users should be familiar with the StateMod run modes by referring to the StateMod software documentation.

If a run mode is selected that uses the model data files and the data have been modified within the StateMod GUI, a prompt will allow the data to be saved, similar to the ***File…Save*** menu discussed in **Chapter 3**. Changes made in the StateMod GUI must be saved to files in order for the StateMod software to recognize the changes.

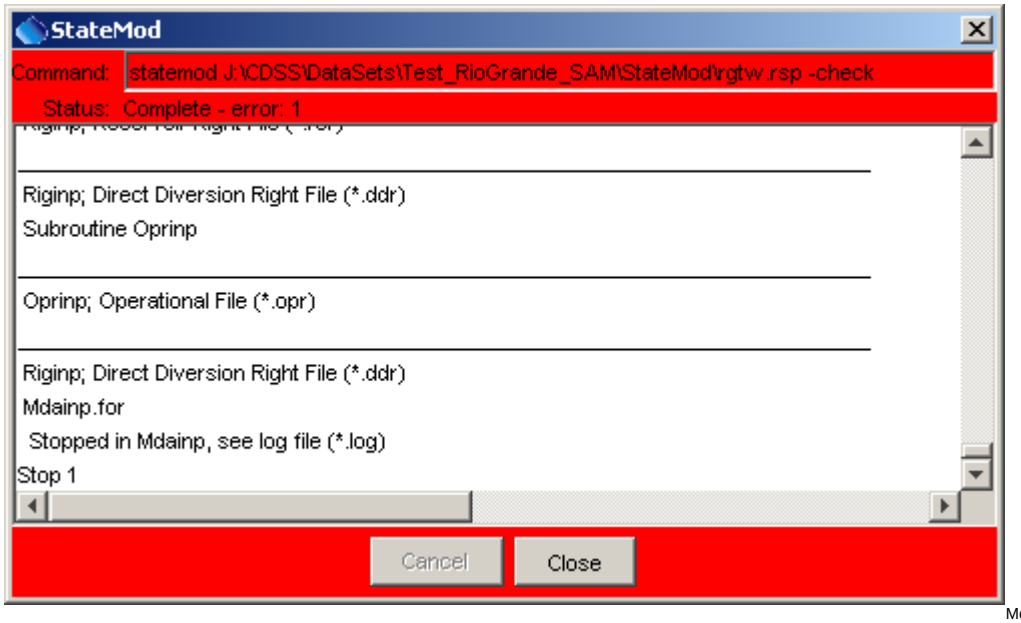
When a run option has been selected (and data set files have been saved, if appropriate), a StateMod model process is created and controlled through the use of a new window, called a process manager, as shown below (in this case for a StateMod version run):




Menu_Run_Version

The StateMod calling command, run and exit status are displayed at the top of the dialog. The StateMod process can be cancelled at any time by pressing the ***Cancel*** button (doing so may require a few seconds to register and an interruption may result in partial output). The dialog does not automatically go away when the model is finished running. Therefore, press the ***Close*** button to dismiss the dialog. **Warning: The run period is set in the *Data…Control…Control* menu. Using a shorter run period during initial model setup and troubleshooting is advised to decrease run time**.

If an error occurs while running StateMod, the ***Process Manager*** dialog will turn red and the status will indicate the StateMod exit status, as shown in the example below:




Menu_Run_Error

**StateMod Run Error Dialog**

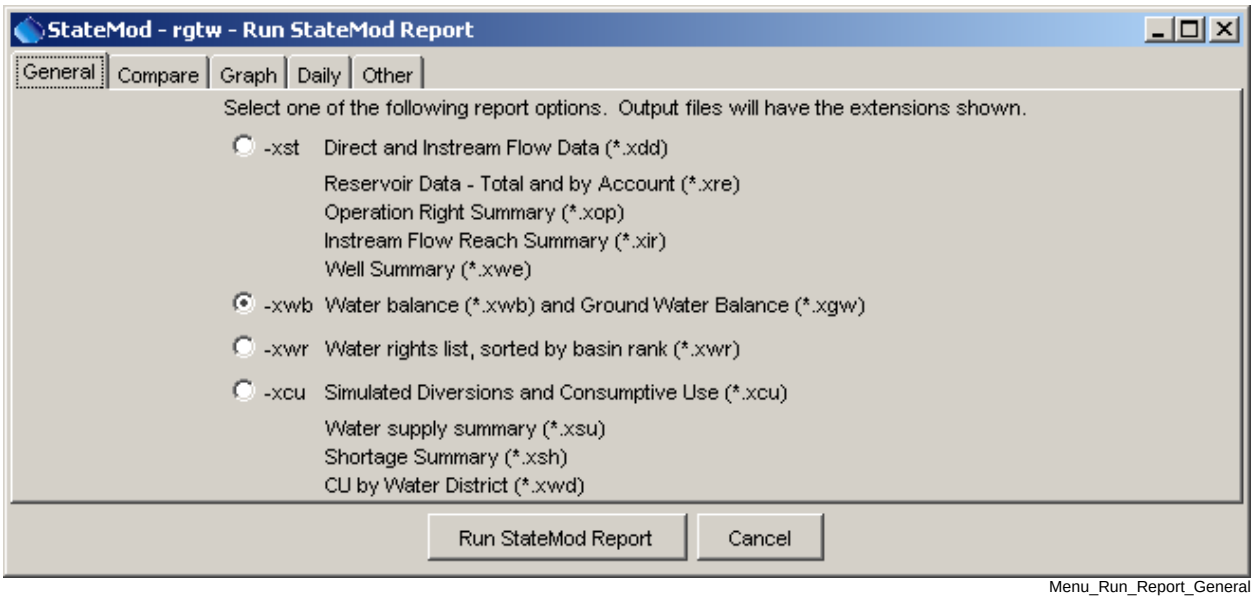
There are variations in how the StateMod GUI interacts with StateMod, depending on the operating system version and Java Runtime Environment that runs the StateMod GUI. Therefore, it is usually best to review the last output from the StateMod run shown in the above window. A messaging indicating a `STOP 0` code will be printed by StateMod if a run is successful. Otherwise, an error occurred and the StateMod log file should be reviewed (see the ***Results…Output Files*** menu and select a log file that matches the response file name).

In most cases a run will fail due to an error in a data file or because the response file references a file that does not exist. After determining the cause of the message, various StateMod GUI windows can be used to correct the data, to allow another run to be made.

In addition to running the main StateMod simulation modes, the StateMod version, update history, and help summary information can be reviewed from the StateMod GUI. The version information may be helpful during troubleshooting.

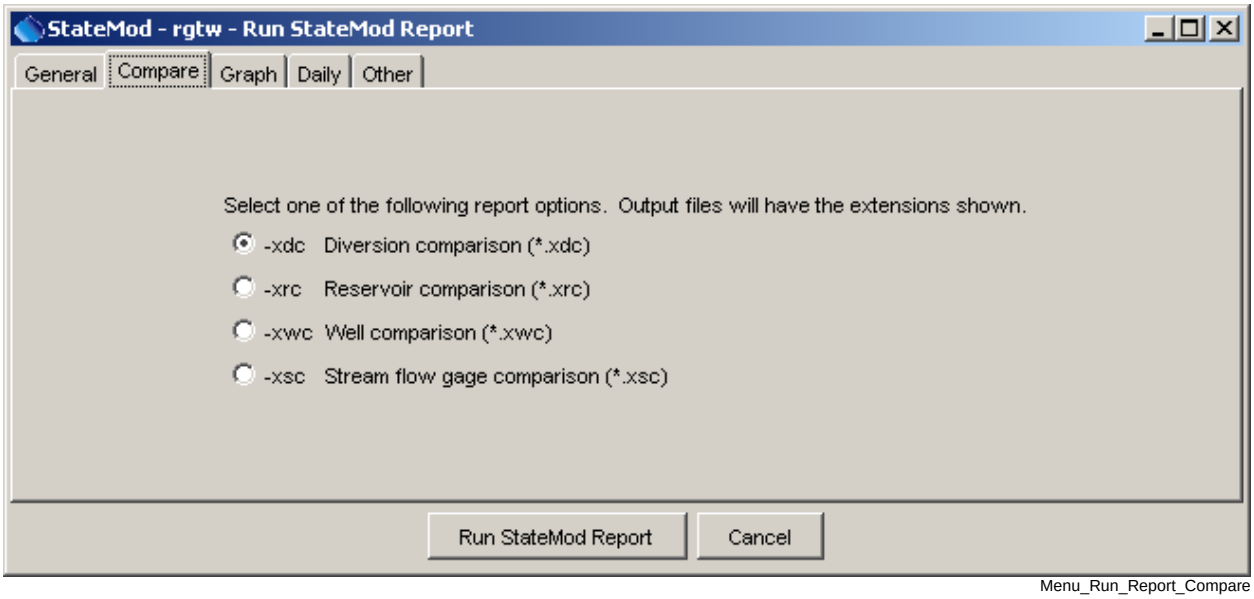
## 6.2 Running StateMod in Report Mode

Selecting the ***Run…Report*** menu displays the following dialog prompting for the report type. The tabbed panels list the various report categories and allow input of user-specified information, as appropriate.



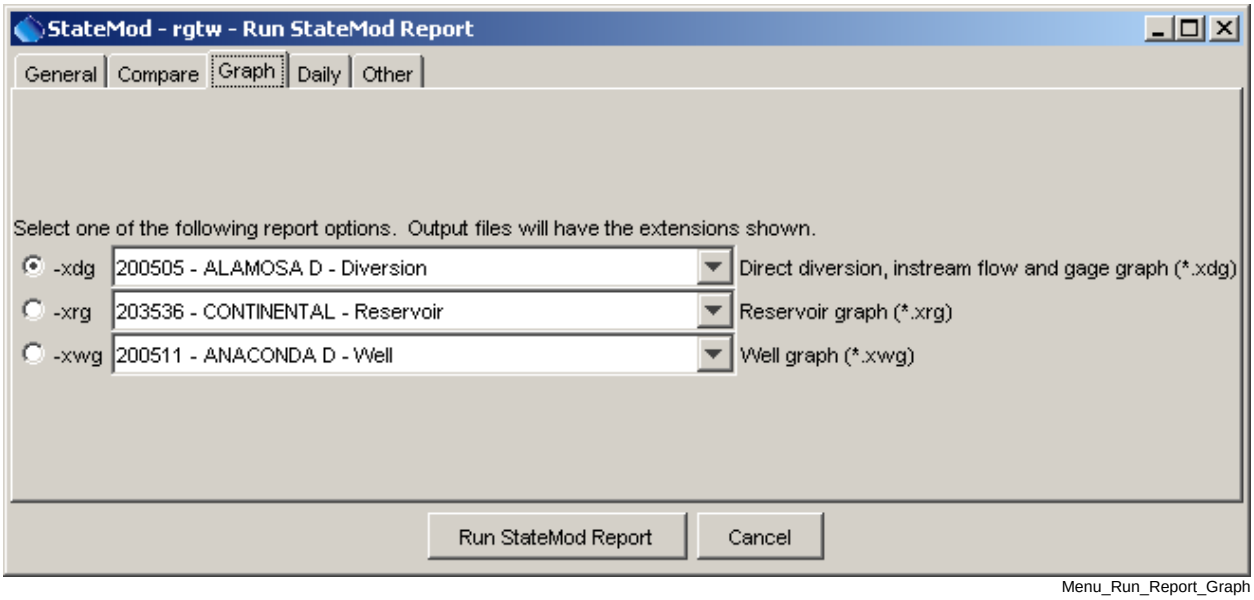

Menu_Run_Report_General

**Report Generation Dialog – General Reports**



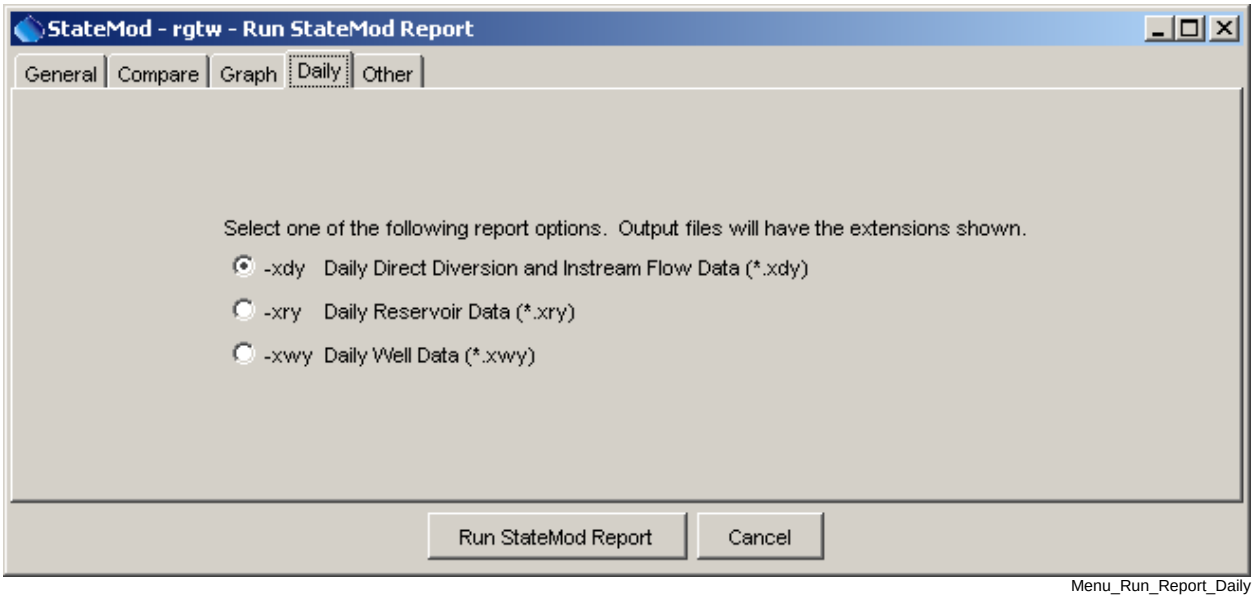

Menu_Run_Report_Compare

**Report Generation Dialog – Comparison Reports**



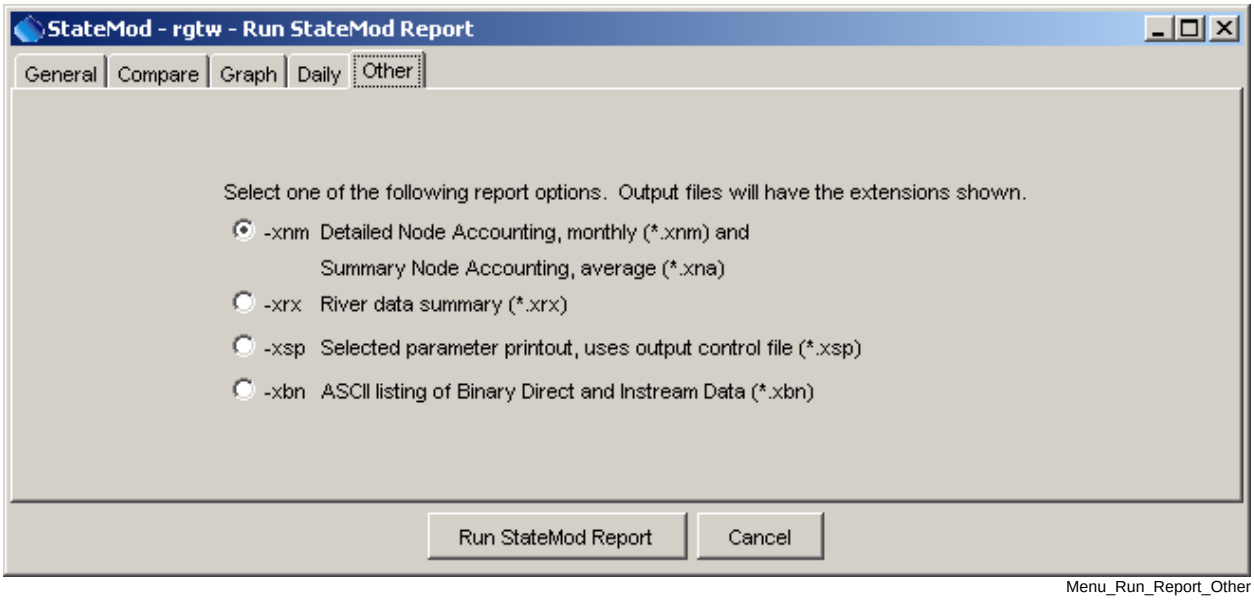

Menu_Run_Report_Graph

**Report Generation Dialog – Graph Reports**




Menu_Run_Report_Daily

**Report Generation Dialog – Daily Reports**




Menu_Run_Report_Other

**Report Generation Dialog – Other Reports**

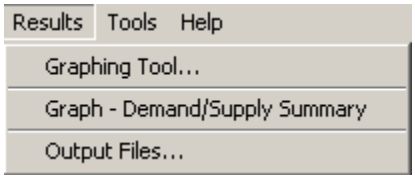
To generate a report, select one of the available radio buttons, and if required, provide additional information. See the StateMod software documentation for more information about each report type. Some reports may be available in the StateMod software but are not yet enabled in the StateMod GUI. Press ***Run StateMod Report*** to generate the output files or press ***Cancel*** to cancel the report.

View the report output using the ***Results…Output Files*** menu. The report files will have the same base name as the response file and an extension that matches the report command line option for StateMod (as indicated in the above tabbed panels).

# 7 Viewing StateMod Results

Version 06.03.02, 2006-03-05, Color, Acrobat Distiller

The ***Results*** menu provides access to StateMod results in graphical form, and allows StateMod output files to be viewed.




Menu_Results

See the ***Data*** menu to view/edit StateMod input data. See the ***Tools*** menu for additional tools for data and output.

## 7.1 Graphing Tool – Create Graphs for StateMod Input and Results

As described in **Chapter 5 – Viewing and Editing Data**, input time series data can be graphed from various data windows if the time series have been read for the data set. However, graphing from the data windows only allows the data for the specific station to be displayed.

The graphing tool overcome these limitations by:

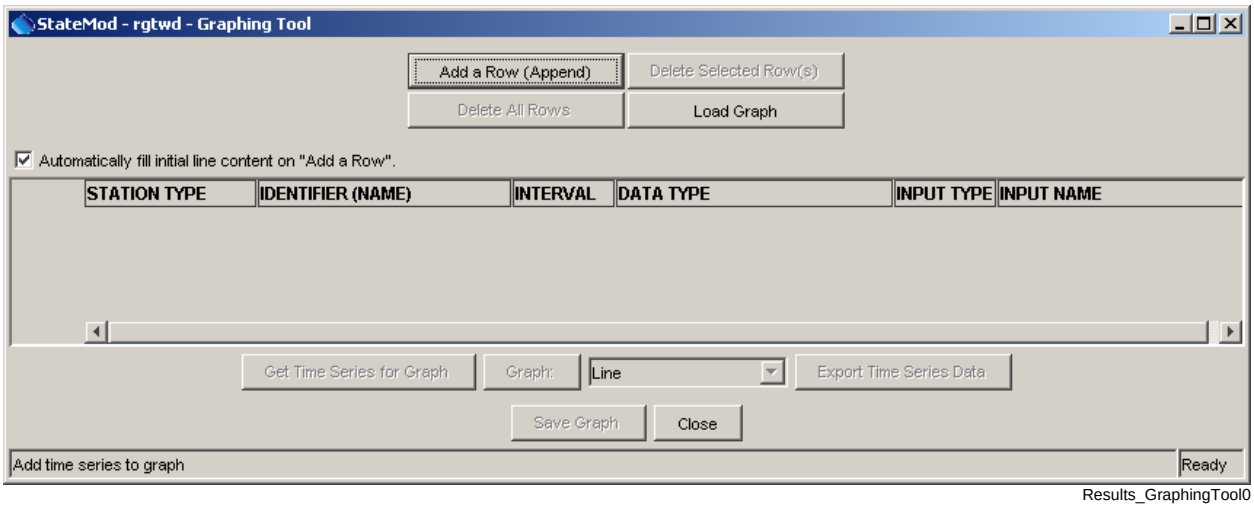
1. Graphing input and output time series on the same graph.
2. Graphing time series from different stations on the same graph.
3. Graphing time series that have been read in, and those that have not on the same graph.

Output time series can only be graphed if the StateMod baseflow and simulate options have been run. In cases where input time series have not been read into the StateMod GUI, processing will be slower because the requested time series will be read from input files.

The features described below are comparable to features available in the TSTool software. The main difference is that the graphing tool described below has knowledge about the StateMod data set and therefore can provide lists of choices to facilitate selecting stations for graphing. The TSTool software is more appropriate if many graphs are to be produced in batch mode, for example during model calibration.

Graph configurations may be saved as "time series products" for use at a later time. The time series product file format is described in the **TSView Time Series Viewing Tools Appendix**.

The ***Results...Graphing Tool*** menu displays a window similar to the following:




Results_GraphingTool0

**Graphing Tool – No Time Series Listed**

The above window facilitates selecting the time series to be graphed. Each row in the list corresponds to a time series and contains the following information:

| | |
|---|---|
| ***STATION TYPE*** | The StateMod station type. |
| ***IDENTIFIER (NAME)*** | The identifier and name for the station associated with the time series to be graphed. A list is provided based on the selected station type. Only the identifier part of provided information is used to identify the time series. |
| ***INTERVAL*** | The interval for data. |
| ***DATA TYPE*** | A unique string indicating the input data type or output parameter. Only the first part of the string is used to indicate the data type for the time series. |
| ***INPUT TYPE*** | `StateMod` for input files and `StateModB` for output parameters read from the binary output files. |
| ***INPUT NAME*** | The name of the file from which to read the time series. If the file has been read as part of the data set, the StateMod GUI will use the time series that was read. Otherwise, the time series will be read from the indicated file. |

To generate a list of time series to graph, and to read or save a list of time series, use the buttons discussed below:

| | |
|---|---|
| ***Add a Row (Append)*** | Add another row. Each time series to be graphed corresponds to a row. Each new row contains default contents matching the previous line to help quickly add a row. The contents of the row should be edited to match a desired time series. |
| ***Delete Selected Row(s)*** | Delete the highlighted rows from the list. Each line to be deleted can be selected by clicking on a cell in the row. Use ***Ctrl***-click to select additional rows. |
| ***Delete All Rows*** | Delete all rows from the list. |

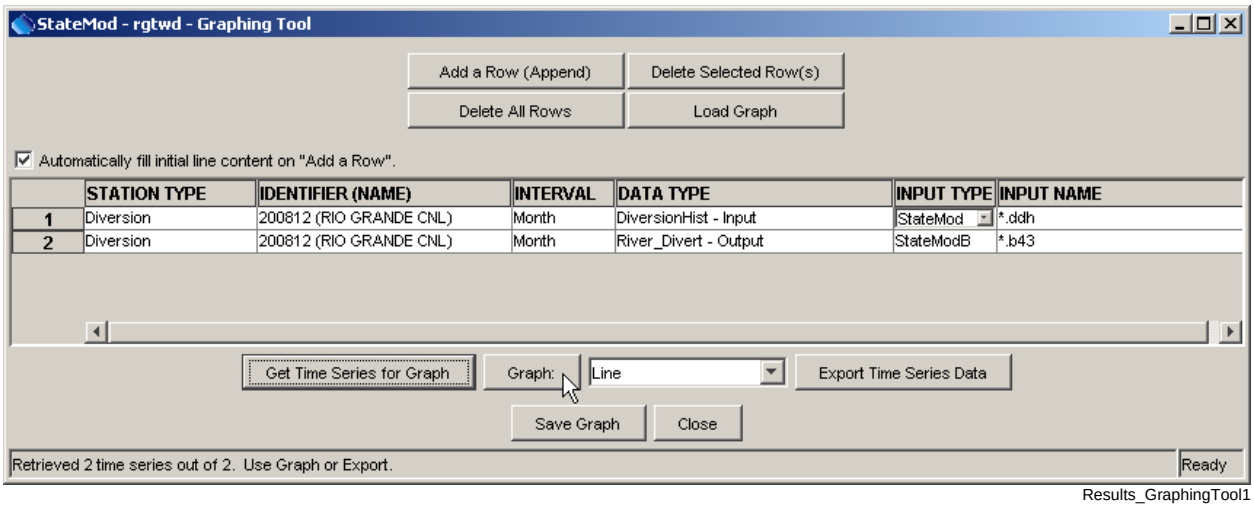
| | |
|---|---|
| ***Load Graph*** | Load a time series product that was previously saved. |
| ***Save Graph*** | Save the list of time series as a time series product (**.tsp*). This feature allows time series products to be loaded at a later time. |

The order of the columns in the time series list facilitates selection of subsequent choices. The remaining buttons at the bottom of the window are used to read time series data and display graphs. The following example illustrates the process to list time series and create a graph; in this case to display historical input data and corresponding simulated output:

1. Start the graphing tool using ***Results…Graphing Tool***.
2. Add an input time series to the list:
    a. Select ***Add a Row (Append)***. A row will be added with default contents.
    b. Select a ***STATION TYPE*** of `Diversion`.
    c. Select an ***IDENTIFIER*** (e.g., `200812`).
    d. Select an ***INTERVAL*** of `Month`.
    e. Select a ***DATA TYPE*** of `DiversionHist - Input`.
    f. Select an ***INPUT TYPE*** of `StateMod` (this is the only choice since an input data type has been selected).
    g. Select an ***INPUT NAME*** of **.ddh* (this is the default and indicates that the time series will be read from the diversion historical time series file corresponding to the data set, matching the response file name). A specific file name can be entered if desired.
3. Add an output time series to the list:
    a. Press ***Add a Row (Append)***. The information from the previous row will be duplicated in the new row.
    b. Select a ***DATA TYPE*** of `RiverDivert - Output`.
    c. Select an ***INPUT TYPE*** of `StateModB` (this is the only choice since an output data type has been specified and must be read from a StateMod output binary file).
    d. Select an ***INPUT NAME*** of `*.b43` (this is the default and indicates that the time series will be read from the diversion historical time series file corresponding to the data set, matching the response file name).
4. Retrieve the time series:
    a. Press the ***Get Time Series for Graph*** button below the list. After retrieving the time series, the ***Graph*** button will be enabled. The process will be slower if time series need to be read from files.
5. Graph the time series:
    a. Select the desired graph type from the choice to the right of the ***Graph*** button.
    b. Press the ***Graph*** button. A graph will appear (see example below). The graphing tool is described in more detail in the **TSView Time Series Viewing Tools Appendix**.
    c. If appropriate, select a different graph type and create additional graphs.
6. Save the time series product:
    a. Press the ***Save Graph*** button. Save the graph as a time series product (**.tsp*) file. The file can then be loaded later using the ***Load Graph*** button, and steps 4 and 5 can be executed.
7. If appropriate, save the time series data to a file using the ***Export Time Series Data*** button.

The above steps can be repeated as appropriate, to change the time series list or select a different graph type for output.

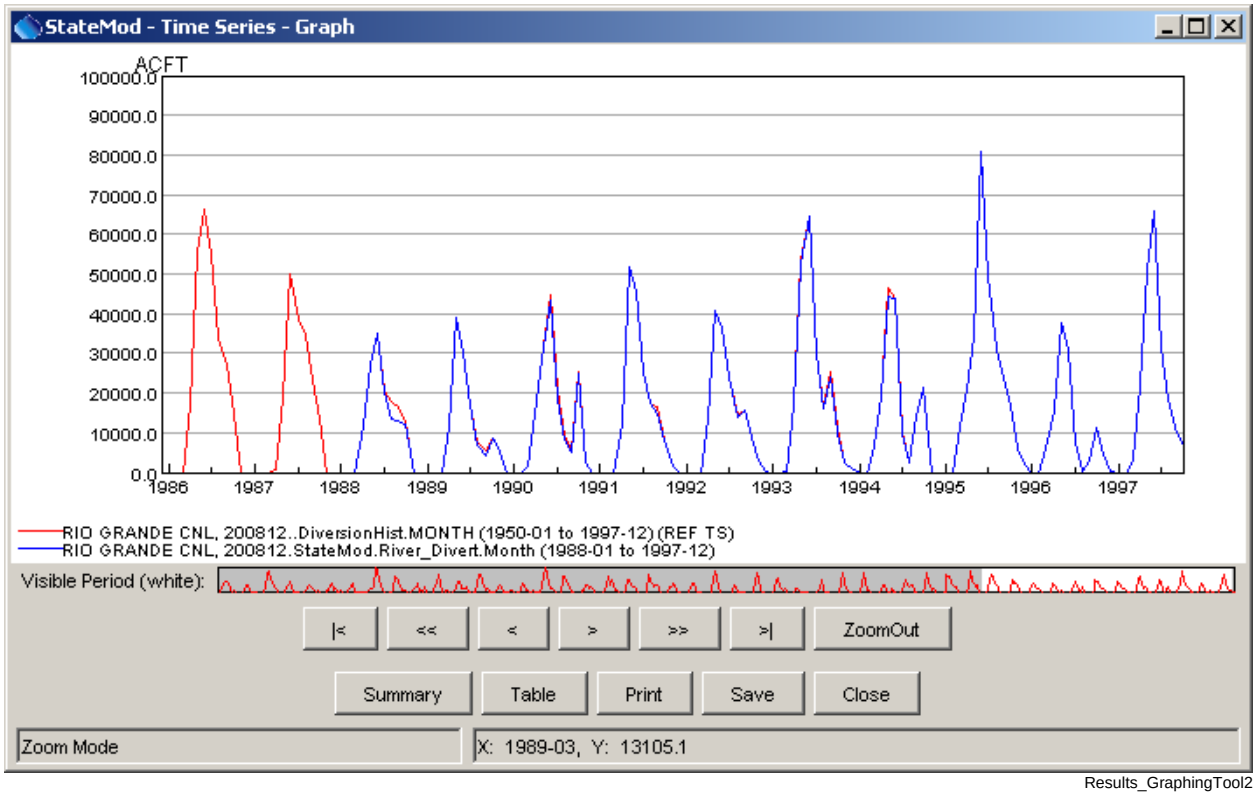
For the above example, the resulting graphing tool interface with a list of time series appears as follows:




Results_GraphingTool1

**Graphing Tool – With Time Series Listed, Before Creating a Graph**

Once time series are retrieved, the ***Graph*** and ***Export Results*** buttons will be enabled. The status message area at the bottom of the window will indicate whether all time series could be retrieved. During the retrieval process, errors may be shown for time series that cannot be retrieved. This is usually due to a time series not being found in output (e.g., because of the output control file limiting the output) or because StateMod has not been run to generate output.



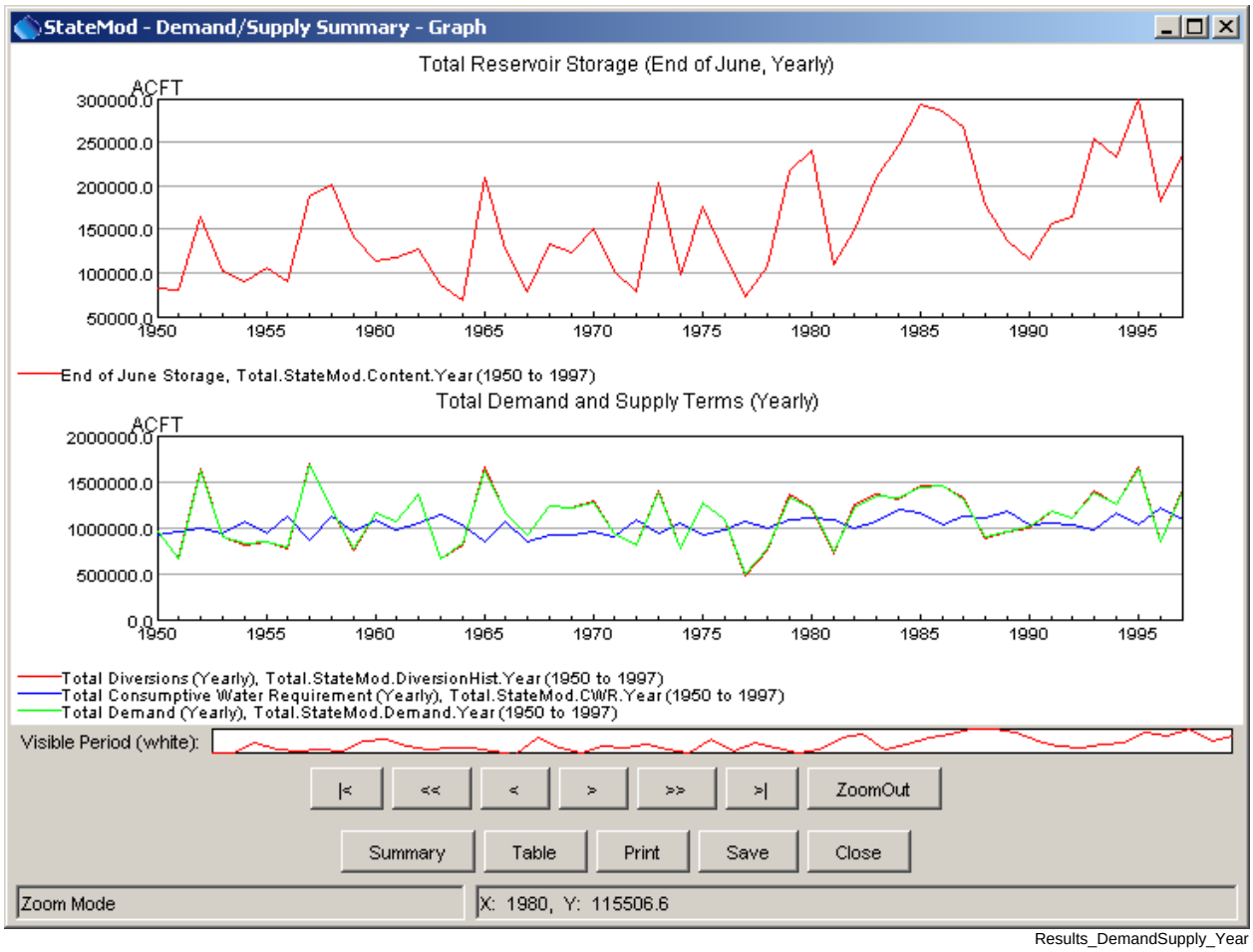

Results_GraphingTool2

**Example Graph Illustrating Input and Output Time Series**

## 7.2 Graph Demand/Supply Summary

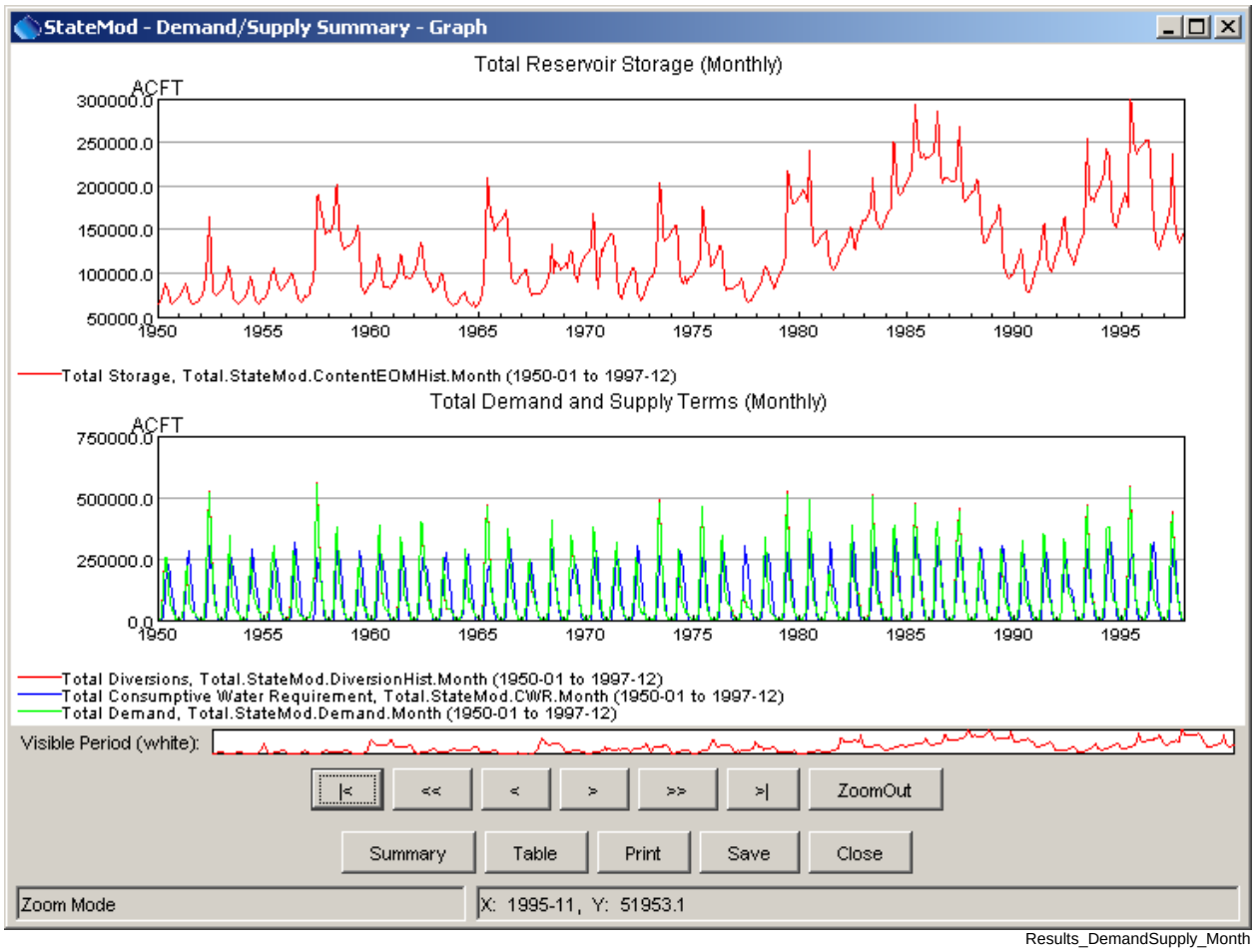
The ***Results…Graph – Demand/Supply Summary*** menu creates a monthly and yearly graph of important water supply and demand time series. Every time series in the data set is added to calculate the total. These graphs are useful for determining overall basin behavior and to illustrate long-term trends.

The following figures are examples of the resulting graphs.




Results_DemandSupply_Year

**Example Demand/Supply Summary Graph (Yearly)**

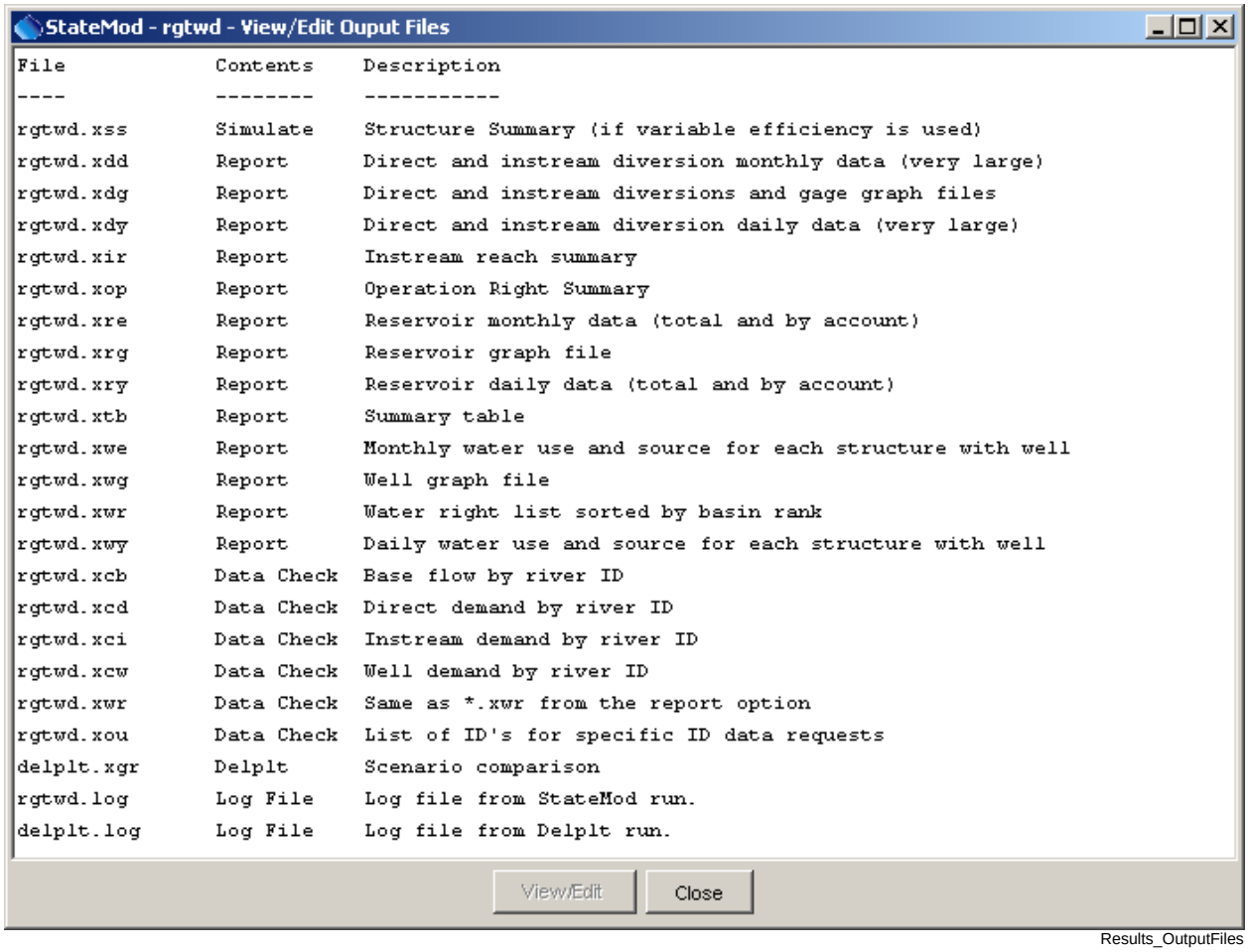



Results_DemandSupply_Month

**Example Demand/Supply Summary Graph (Yearly)**

## 7.3 View StateMod Output Files

The ***Results…Output Files*** menu displays a list of StateMod output files, as shown below:




Results_OutputFiles

**StateMod Output File List**

The output file list is determined by checking recognized file extensions. To view an output file, select an item from the list and then press the ***View/Edit*** button. By default, Notepad is used to display the file. The editor for the session can be changed using the ***Tools ... Options*** menu.

This page is intentionally blank.

# 8 Using the Map

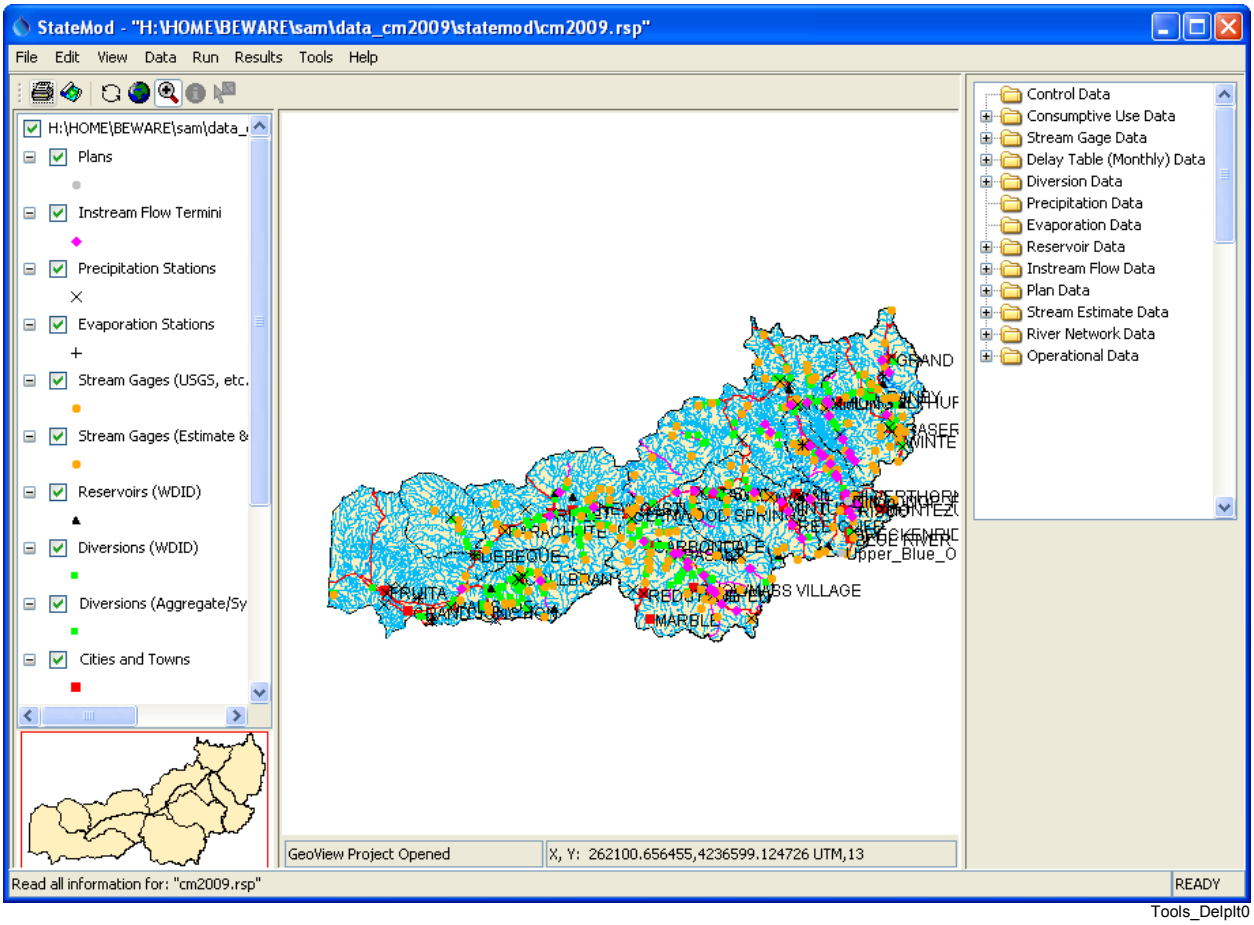
Version 07.04.00, 2013-04-18

This chapter describes how to use the map features in the StateMod GUI. Other information related to maps includes:

- **See Chapter 9 – Tools** for more information about map-based tools, such as displaying summary information on the map.
- The **Configuring Spatial Data for the StateMod GUI** appendix describes how to configure the GeoView Project file (**.gvp*), which controls the initial appearance of the StateMod GUI map display for a data set.

## 8.1 Integration of Spatial Data with a StateMod Data Set

The StateMod GUI will display a map if the selected data set response file specifies a valid GeoView project file (**.gvp*) using the `GeographicInformation` property. The map interface uses software that has been developed for CDSS tools and has features appropriate for visualization, navigation, and data access. It is not a full GIS analysis tool.

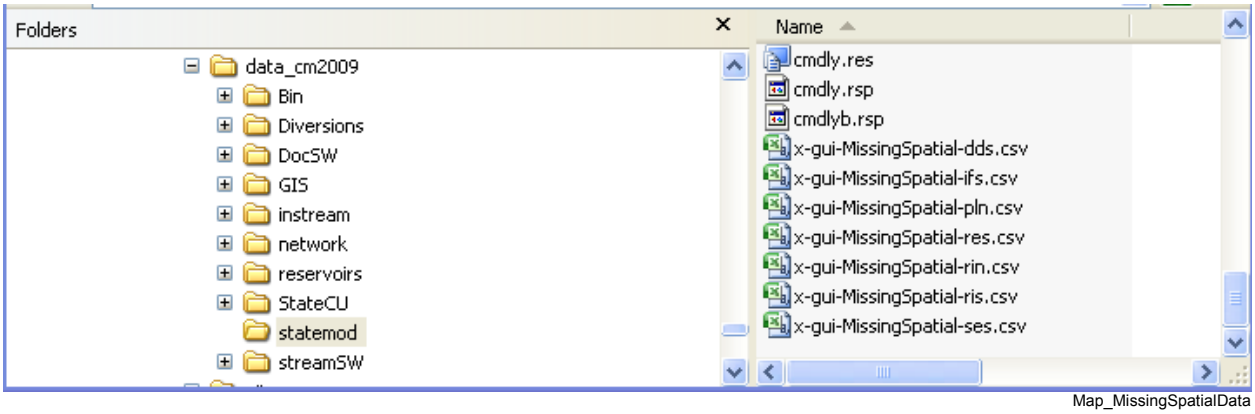



Tools_DelpIt0

**Main Map after Opening Data Set**

StateMod data files do not include spatial information such as latitude and longitude. Therefore, all interactions between the map and the data set occur using identifiers that are present in the data set and as a spatial data layer attribute. For example, the stream gage layer will include USGS stream gage identifiers and the same identifiers are used in the StateMod network and StateMod stream gage data files. Cross-referencing spatial data can be complicated by a number of issues, including:

- Care must be taken to treat zero-padded identifiers as such. USGS stream gages with identifiers that start with zero must be treated as strings to preserve the zeros and allow comparison.
- Spatial data layers may contain more or fewer features than are in a StateMod data set.
    - To facilitate spatial data use, the GUI allows using layers that have more features than are in the StateMod dataset. Only matching features are displayed. See the ***View…Map – Show Features not in Data Set*** menu, which allows displaying all features in a layer.
    - The GUI will identify data set stations with missing coordinates and create *x-gui-MissingSpatialData*.csv* files with the information, as illustrated in the following figure:



Map_MissingSpatialData

    - The resulting CSV files can be configured in the **.gvp* file and coordinates specified (for example use Google Maps or other approach to determine coordinates). To simplify configuration, geographic coordinates (longitude and latitude) are used. An example of a missing data file is shown below:

```
# File generated by...
# program:      StateModGUI 07.04.00 (2013-04-17)
# user:         sam
# date:         Thu Apr 18 14:04:34 MDT 2013
# host:         AMAZON
# directory:    H:\HOME\BEWARE\sam\data_cm2009\statemod
# command line: StateModGUI
#  -home
# #
C:\Develop\StateModGUI_SourceBuild/StateModGUI/test/operational/CDSS
#
# dds has 39 out of 414 locations with missing spatial data.
#
# Specify X and Y in projected coordinates to match other layer data.
#
"ID","Name","Long","Lat","X","Y","Note"
420520,GRAND JCT GUNNISON PL,,,,,"GUI Detected no coordinates"
420541,REDLANDS POWER CANAL,,,,,"GUI Detected no coordinates"
950001,Grand Valley Project,,,,,"GUI Detected no coordinates"
950002,USA Power Plant,,,,,"GUI Detected no coordinates"
950003,Orchard Mesa Check,,,,,"GUI Detected no coordinates"
…
```

- The GUI allows multiple layers to be identified for a single StateMod station type. For example, the **.gvp* file property `AppLayerType=Diversion` indicates that the layer should be associated with diversion stations. In this way, it is possible to use an "official" data file for actual diversion stations (e.g., as produced from a spatial database), and a separate layer for stations that are known only the model, such as aggregate diversions that have contrived identifiers.
- In cases where a location identifier appears in more than one layer, the layer that is matched first is used. Consequently, it may be important to list layers in a specific order in the **.gvp* file.
- The StateMod GUI does not currently edit spatial data files when stations are added through the GUI. This issue may be resolved with additional software enhancements.

With the above data-handling features in the StateMod GUI, it is possible to ensure that every StateMod data set feature has spatial data and can be displayed on the map.

## 8.2 Map Display Features

The map display has been designed to draw maps very quickly, using common input data formats. The following tools are provided to interact with the map:


Print the map (after printing it may be necessary to refresh the map to restore the view)


Save the map as an image file


Redraw the map

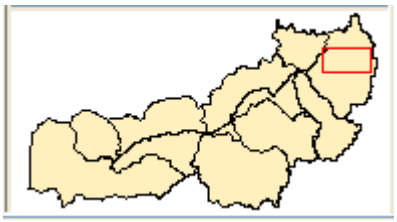

Zoom to full extent of map


Change to zoom mode – the mouse can be used to draw a box on the map to specify the zoom extent


Change to information mode – the mouse can be used to click on or draw a box around map features to display attribute values for the feature(s) (a map layer must first be selected in the legend by clicking on the layer name)


Change to select mode – the mouse can be used to click on or draw a box around map features to select the feature(s) (a map layer must first be selected in the legend by clicking on the layer name) – this feature currently is not used

An overview map is shown in the lower left of the main GUI window.

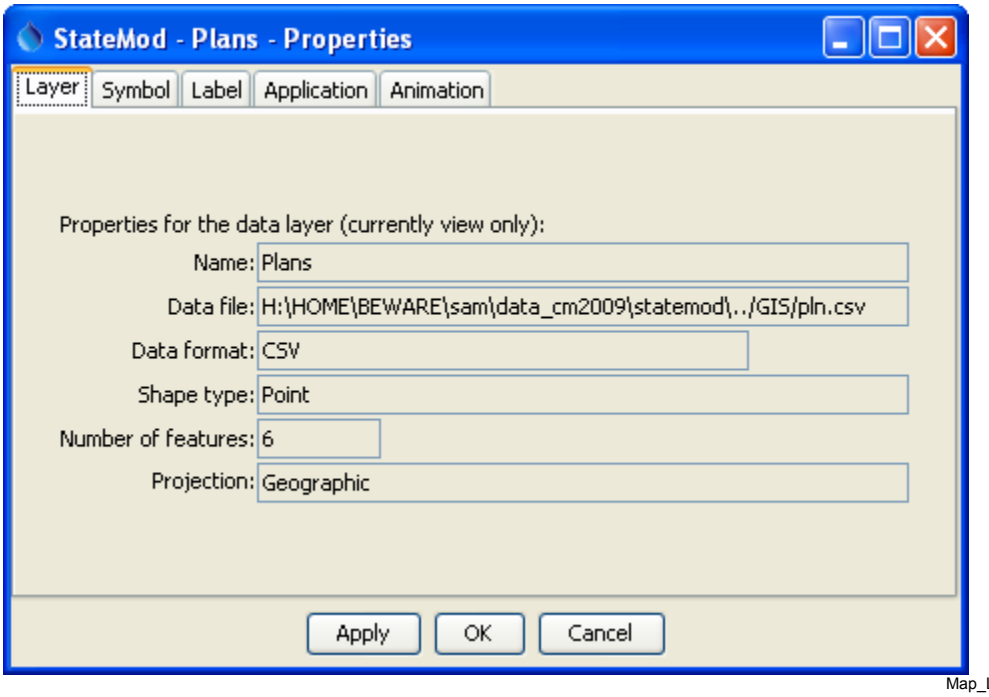


Tools_Overview

**Overview Map Showing Current Zoom Extent**

This map shows the current visible extent of the main map. Using the mouse to draw a box on the overview map will cause the main map to zoom. The layers shown in the overview map are indicated by `ReferenceLayer=True` in the **.gvp* map configuration file.

The map legend (layer list) on the left side of the map displays layers. Use the checkboxes to turn layers on and off. The map will automatically refresh. Right click on a layer name to access its properties:

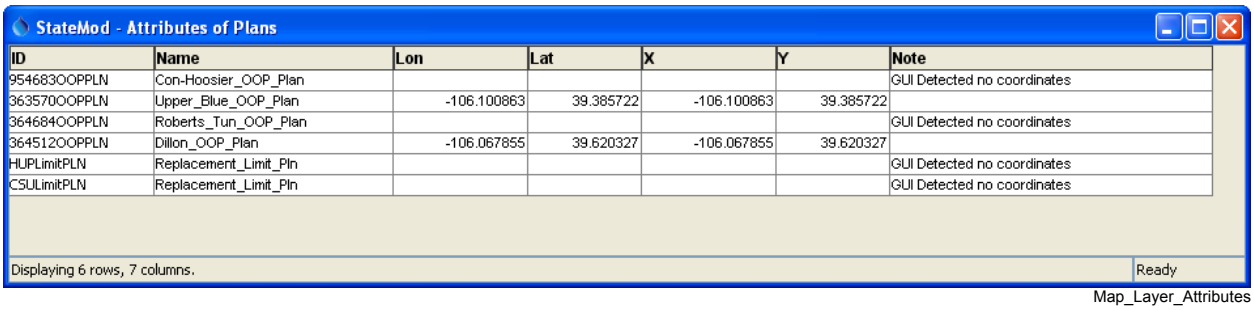



Map_Layer_Properties

**Map Layer Properties**

The properties are useful for understanding data file locations and troubleshooting map issues. Map layer properties can be changed to facilitate understanding the displayed information. For example, layer labels can be enabled to display station names. By default only major features are labeled and the map tool does not provide scale-dependent labeling. Changed map properties cannot be saved (the GeoView Project file, **.gvp*, must be edited with a text editor).

The layer attributes can be viewed by right clicking on the layer name and selecting ***View Attribute Table***, which will display a table as shown in the following figure:



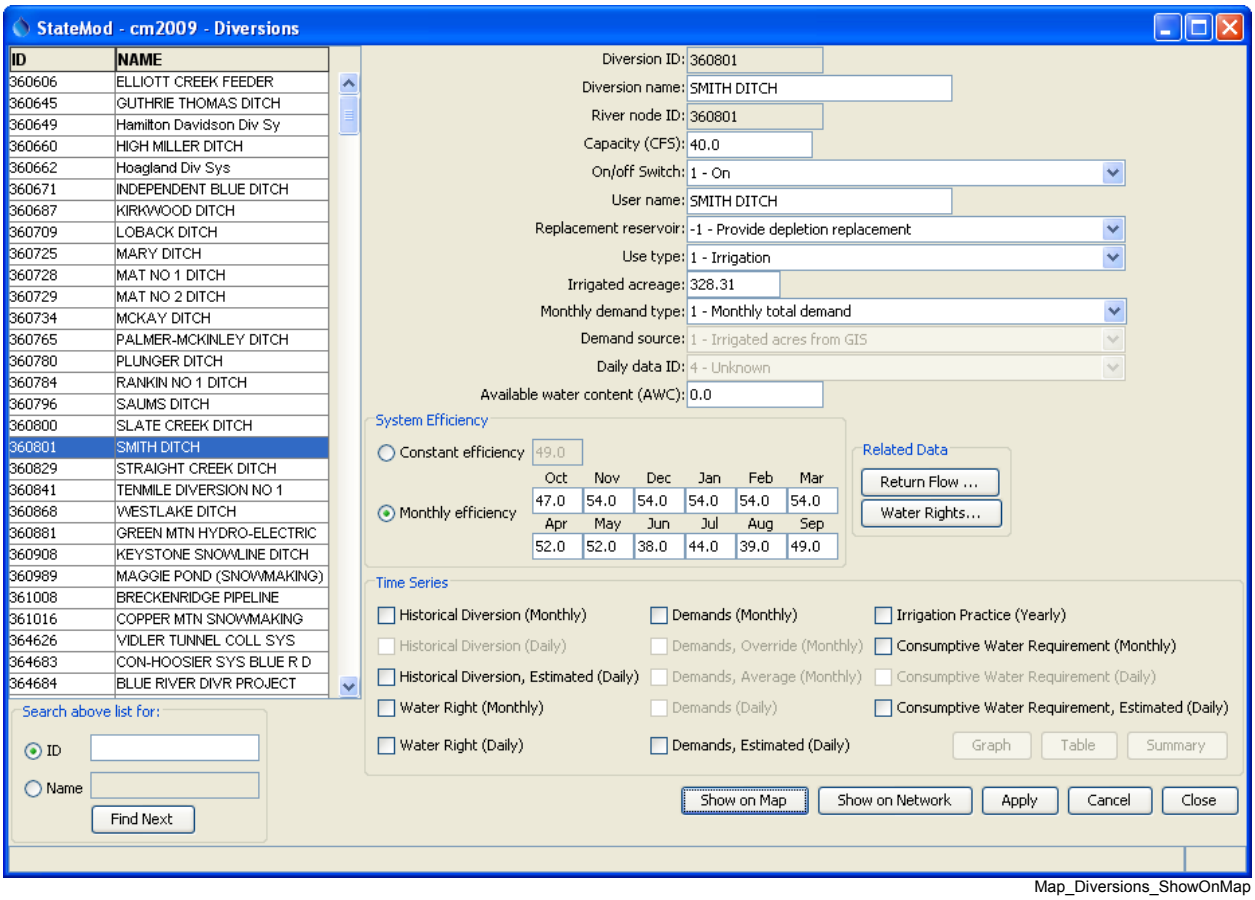

Map_Layer_Attributes

**Map Layer Attributes**

Note that in the above example the list of plans in the spatial data file is provided by a simple comma-separated-value file and some of the coordinates are missing (or perhaps are not appropriate for the plan).

## 8.3 Displaying Specific StateMod Data Set Items on the Map

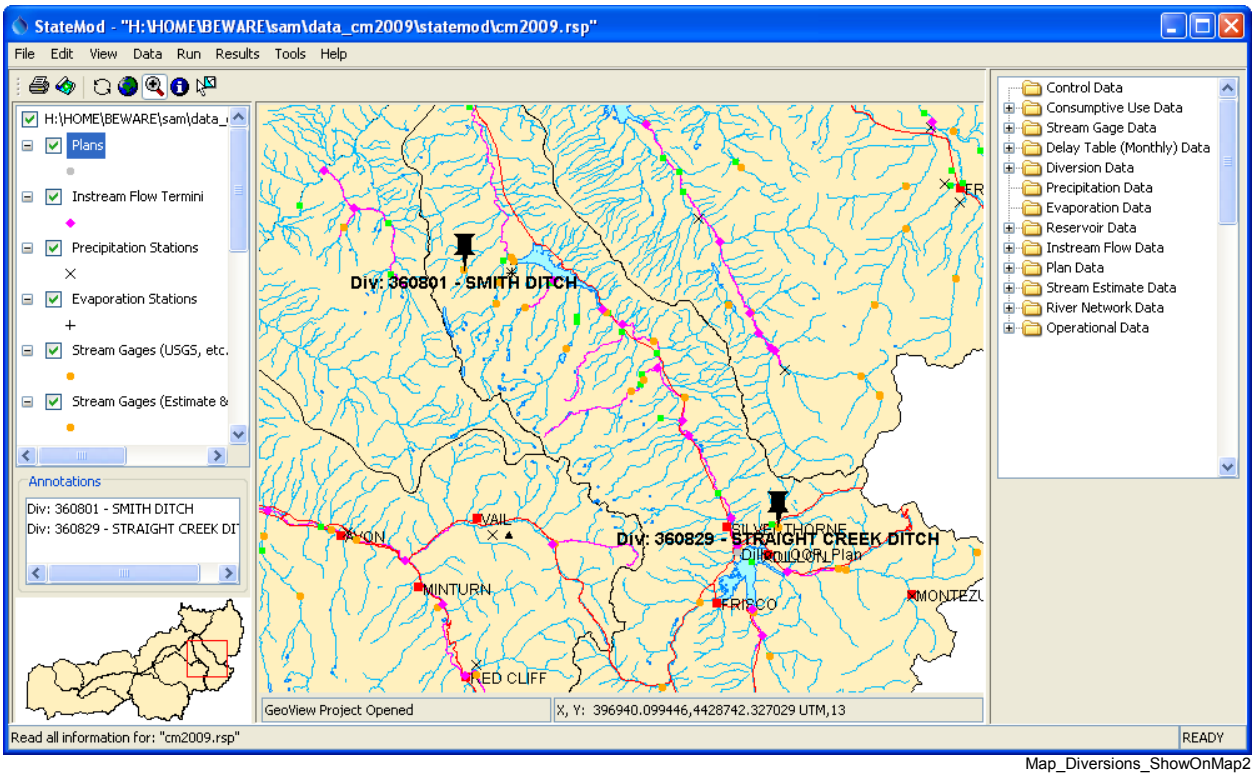
StateMod data windows for primary data types with a physical location (diversion stations, reservoir stations, etc.) include a ***Show on Map*** button at the bottom of the window, which will be enabled if the selected location has spatial information in a map layer (determined by matching the station identifier with identifiers in map layers).




Map_Diversions_ShowOnMap

**Diversion Stations Window Illustrating "Show on Map" Button (bottom)**

Clicking on the ***Show on Map*** button will cause the diversion station to be annoted on the map with a push pin and label. The map will zoom to the region of the diversion station. Repeating the steps will add additional annotations to the map (see figure below). To clear the annotations, right click on the ***Annotations*** list below the map legend.



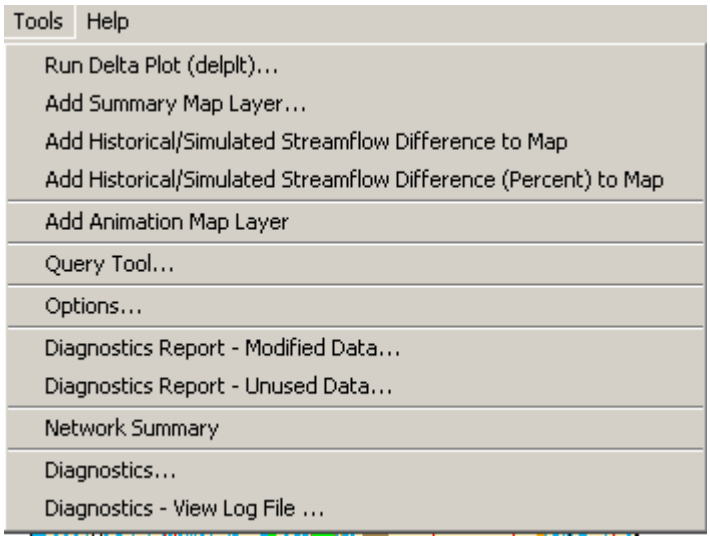

Map_Diversions_ShowOnMap2

**Map Annotated with Diversion Stations After Using "Show on Map" Button**

# 9 Tools

Version 07.04.00, 2013-04-18

The ***Tools*** menu provides access to useful tools.




Menu_Tools

The following sections describe each tool. In some cases, tools have been added to facilitate development and have limited functionality.

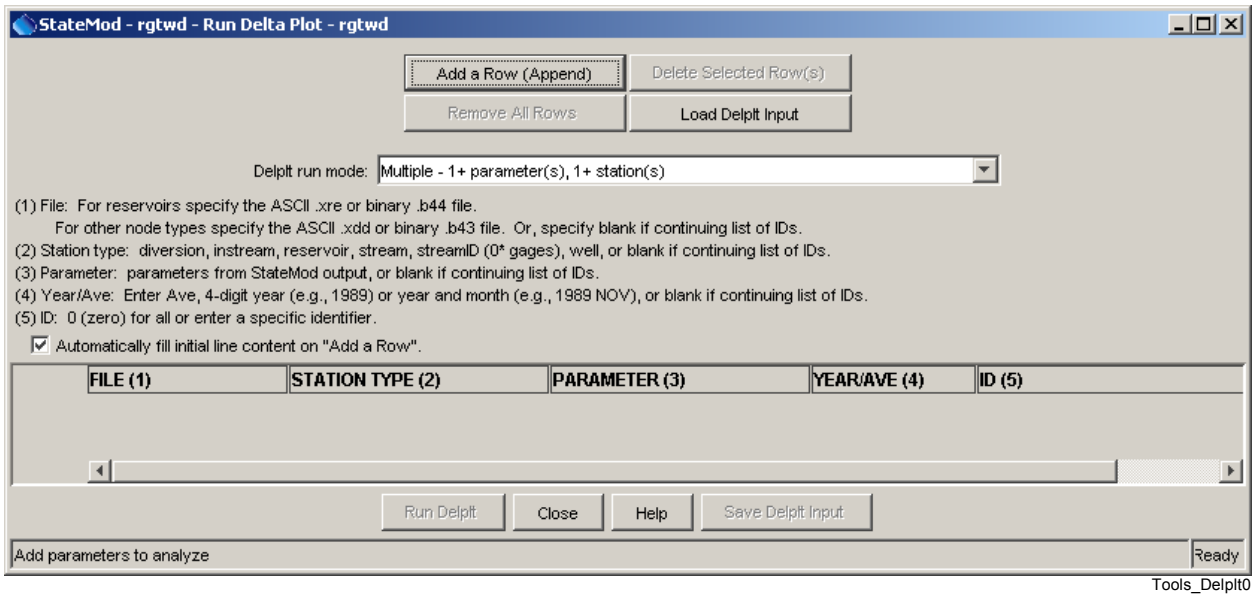
## 9.1 Run Delta Plot – Summarize Simulation Results and Compare Data Sets

**Current StateMod utilities may not be consistent with the GUI. The following feature was implemented with an early version of the delplt tool and may no longer be functional. A work-around is to use TSTool processing features.**

The ***Tools…Run Delta Plot (delplt)*** menu runs the **delplt** program, a utility program supplied with StateMod software that analyzes StateMod results. The **delplt** program can:

1. Summarize parameter values for the current data set.
2. Summarize one or more parameters for a data set, for one or more data sets.
3. Determine the difference for common parameters in multiple data sets.

The **delplt** program functionality is described in the StateMod model documentation. The StateMod GUI edits the **delplt** input file, runs the program, and optionally displays results on the main interface map. When the ***Tools…Run Delta Plot (delplt)*** menu is initially selected, the interface appears as shown in the following figure:




Tools_Delplt0

**Delta Plot Run Window – No Combinations Listed**

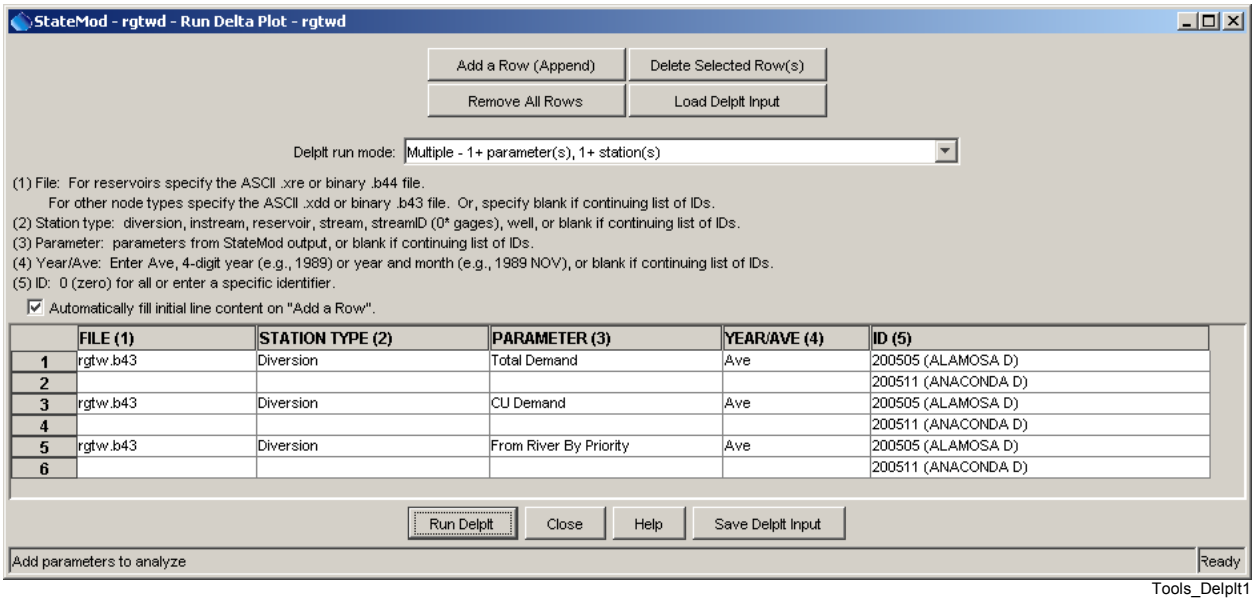
The columns in the list are used to create a **delplt** input file. See the StateMod software documentation for details about the file format. The columns are summarized as follows:

| | |
|---|---|
| ***FILE*** | Specify an *.xre* or *.b44* file for reservoirs. Specify an *.xdd* or *.b43* file for everything else. Using these files requires that the full simulation reports have been generated. Specify the binary files (*.b44* and *.b43*) if files are large, in order to increase performance.<br><br>Specify blank if continuing a list of identifiers in the same file. Specify the file name only to use files in the StateMod data set directory or specify an absolute path to read information from a different StateMod data set. |
| ***STATION TYPE*** | Select a station type from the available choices. This will limit the parameters that are available. Specify blank if continuing a list of identifiers. |
| ***PARAMETER*** | Select a parameter type from the list. This list is dynamically created when the station type is selected. Specify blank if continuing a list of identifiers. |
| ***YEAR/AVE*** | Specify a 4-digit year, or year and month abbreviation (e.g., 1975 NOV), or AVE to compute period averages. Specify blank if continuing a list of identifiers |
| ***ID*** | The station/structure identifier from the StateMod data files. Specify zero "0" to include all identifiers for the combination in the previous columns. |

Use the ***Add a Row (Append)*** button to add new rows to the list. The ***Automatically fill initial line content on "Add a Row"*** checkbox, when selected, will attempt to intelligently fill out a row that is added. For example, if the previous row includes an identifier other than "0", the new row will be blank in the expectation that you will specify only a new identifier. If the previous row has a "0" identifier, then the new row will copy more of the previous row, assuming that you will likely want to change the parameter. If the ***Automatically fill initial line content on "Add a Row"*** checkbox is not selected, then blank rows will be added when rows are added.

Use the ***Remove all Rows*** and ***Delete Selected Row(s)*** buttons to remove existing rows. Use the ***Load Delplt Input*** and ***Save Delplt Input*** buttons to operate on a **delplt** input file, which is useful if the same list of input will be run repeatedly.

After adding information to the table, the window will appear similar to the following:

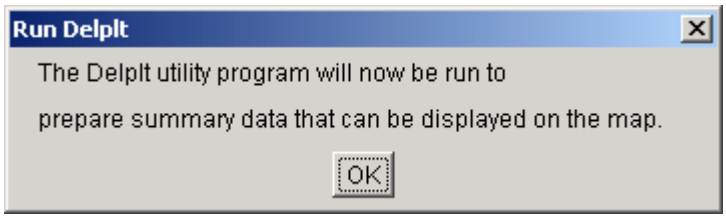



Tools_DelpIt1

**Delta Plot Run Window – With Combinations Listed**

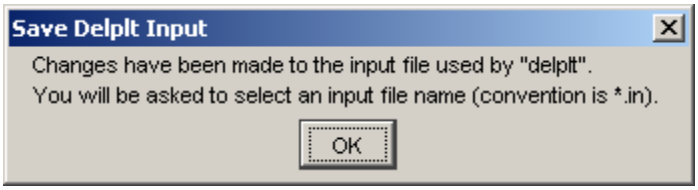
Save the **delplt** input by selecting the ***Save DelpIt Input*** button. Alternatively, the StateMod GUI will prompt to do so when ***Run DelpIt*** is pressed (see step 2 below). To process the StateMod output using **delplt**:

1. Press the ***Run DelpIt*** button. The following dialog will be shown:



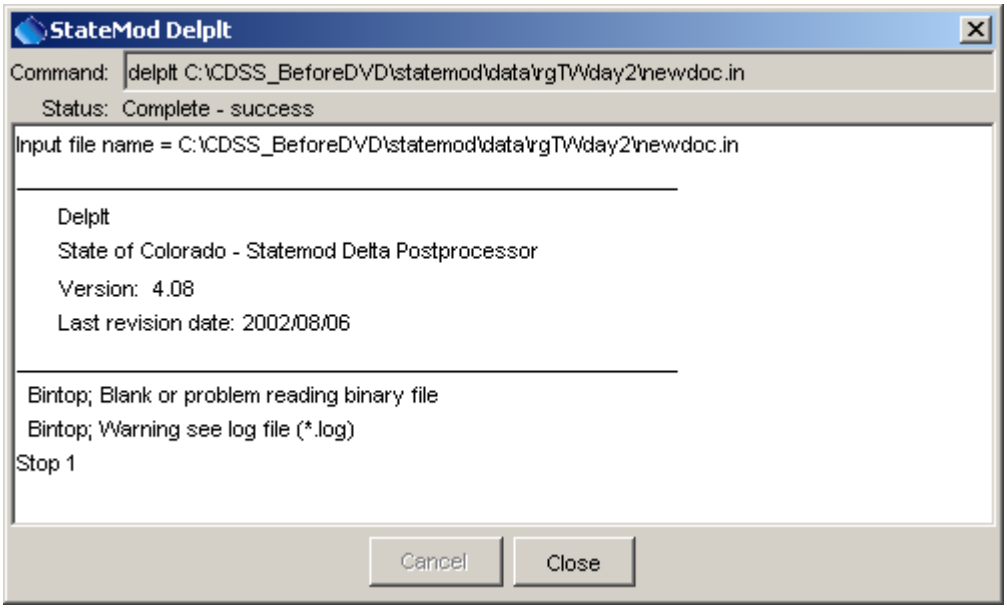

Tools_DepIt2

2. If the input file has not been saved, the following dialog will be shown. Press ***OK*** and then use the file browser to select a file name. **The delplt program can only process filenames that follow the 8.3 DOS filename convention.**



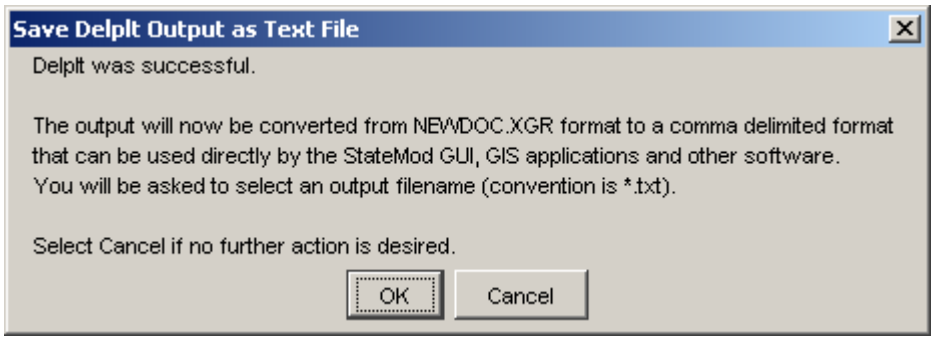

Tools_DelpIt3

3. The **delplt** program is run as a separate process and output is displayed in a dialog as shown below. The output from the run should be reviewed to make sure a `STOP 0` code is printed and the messages do not indicate a problem. If successful, press ***Close*** and continue below. If an error occurs, refer to troubleshooting information and the **delplt** log file, which will have the same base name as the input file and an extension *.log*.



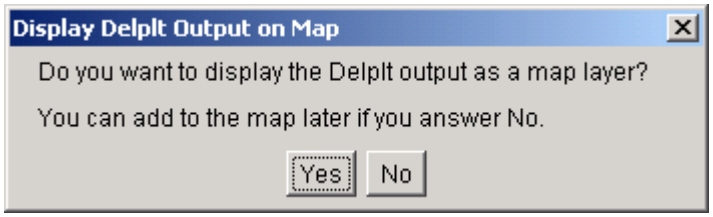

Tools_Delplt4

4. A successful **delplt** run creates an output file with the extension *.xgr*. This file is converted to a more general format by the StateMod GUI, as indicated by the following dialog. Press ***OK*** and then enter a name for the text file using the file browser that is displayed.



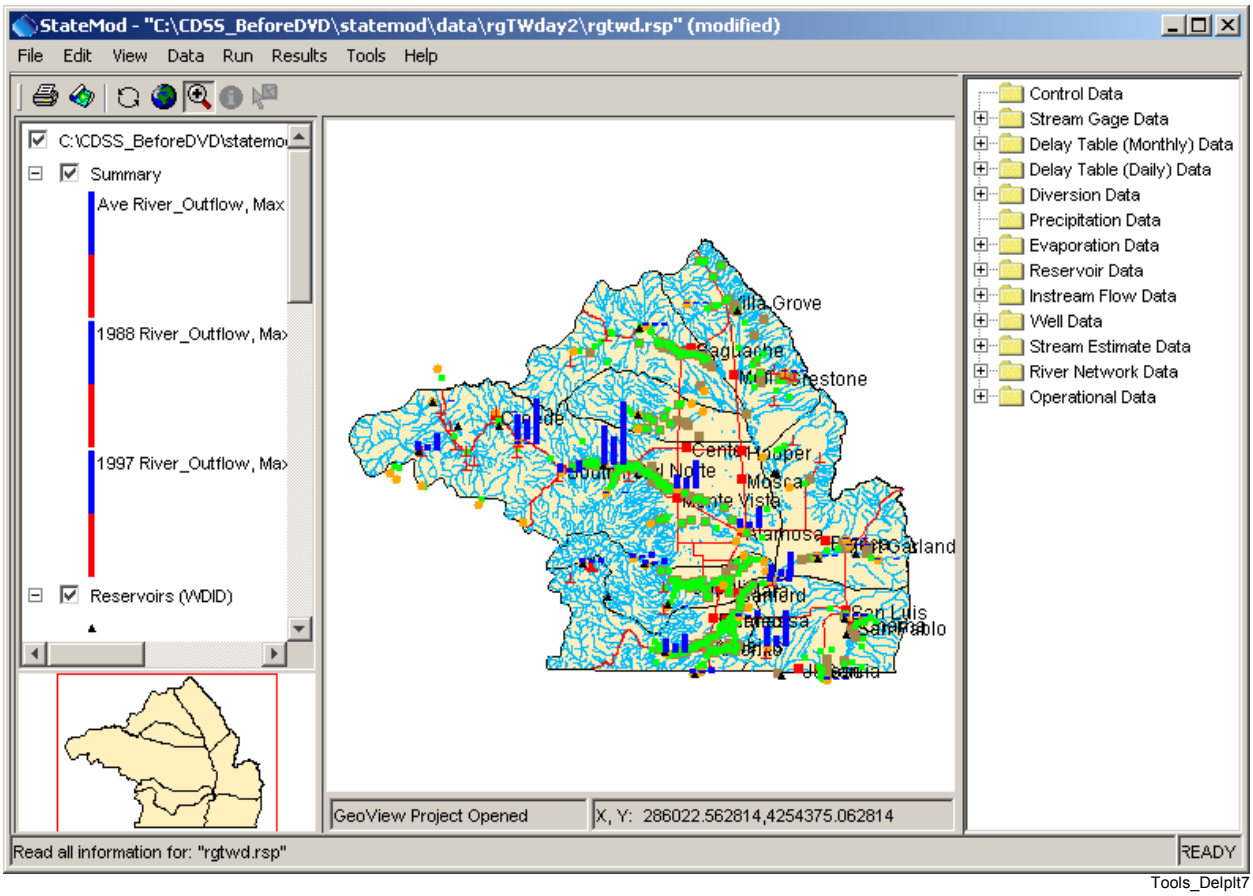

Tools_Delplt5

5. After the **delplt** is converted to a text file, you will be prompted to indicate whether the results should be shown as a summary map layer:




Tools_Delplt6

6. If you specify ***No*** in response to the above dialog, the delplt run is complete and nothing will be shown on the main interface map. Refer to **Section 9.2** for information about adding the information to the map later. The text file can also easily be imported into another application, including a spreadsheet. If you specify ***Yes*** in response to the above dialog, a summary map layer will be added to the main interface, as shown below. The main **delplt** run window can be closed or the combinations of parameters can be further modified and the process repeated.




Tools_Delplt7

**StateMod GUI Main Interface Showing Summary Map Layer**

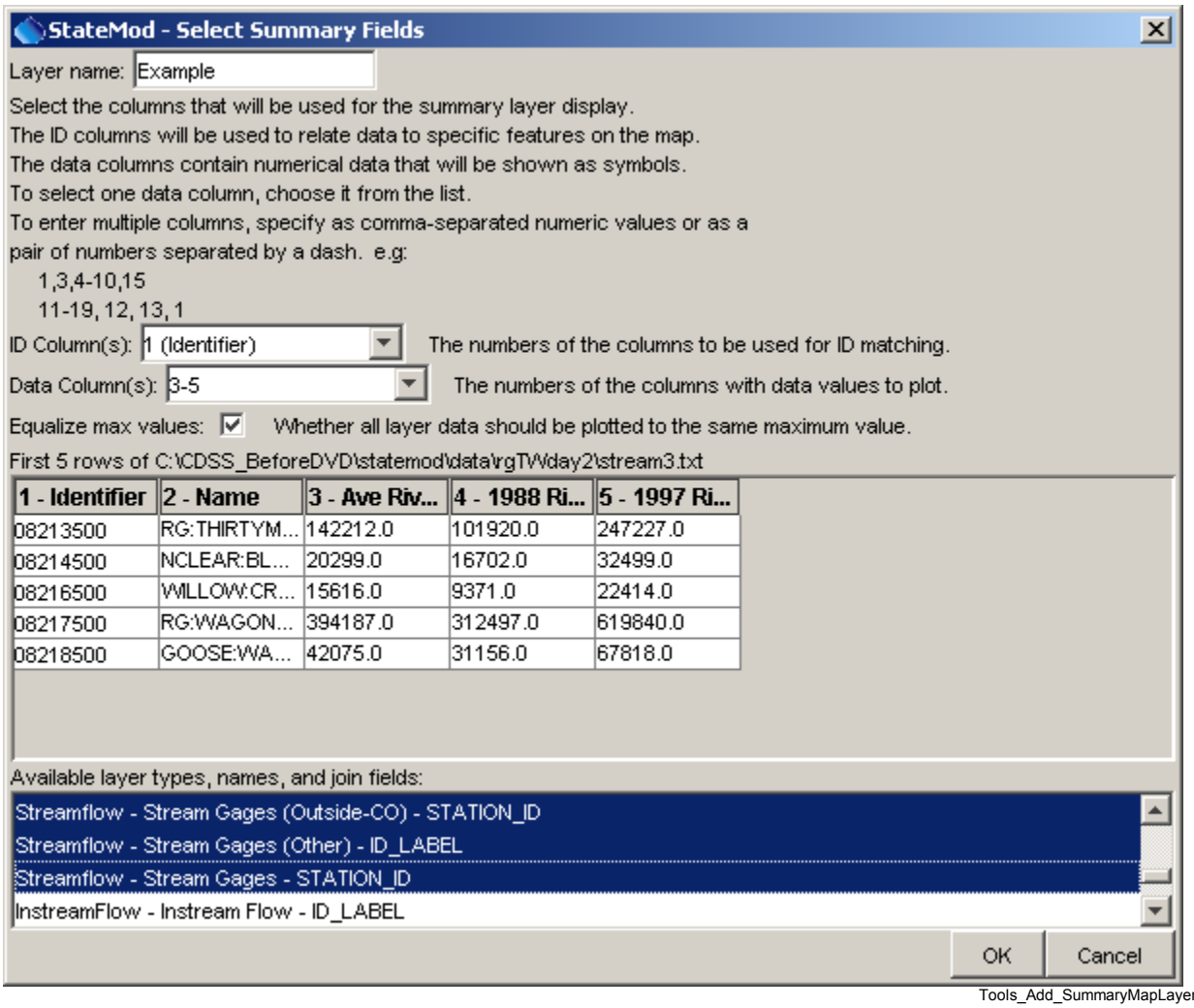
The scale for each bar is consistent and the maximum value is indicated in the legend. Positive values are shown as blue bars and negative values are shown as red bars. The layer can be deleted by selecting it in the layer list, right clicking, and selecting ***Remove Layer***.

## 9.2 Add Summary Map Layer

The ***Tools…Add Summary Map Layer*** menu creates a map layer using existing layers to find coordinates, and draws bars on the map to symbolize data values in a supplied data file. This tool performs steps similar to the final steps discussed in the previous section. An example data file to use for summary data is as follows:

```
Identifier, Name, Ave River_Outflow, 1988 River_Outflow, 1997
River_Outflow
"08213500",RG:THIRTYMILEBRG,142212.0,101920.0,247227.0
"08214500",NCLEAR:BLWCONTRES,20299.0,16702.0,32499.0
…etc…
```

A text file of this format consists of a header line describing the contents of each column, followed by rows of data. Each row contains a station identifier, its name, and data values corresponding to the headers. The identifiers in the file must match the join fields (identifiers) in one or more layers in the data set. Only features with geographic locations can be displayed on the map. After selecting the ***Tools…Add Summary Map Layer*** menu, the following dialog will be shown, which allows configuration of the new layer. Input has been provided as an example. Pressing ***OK*** adds the map layer.



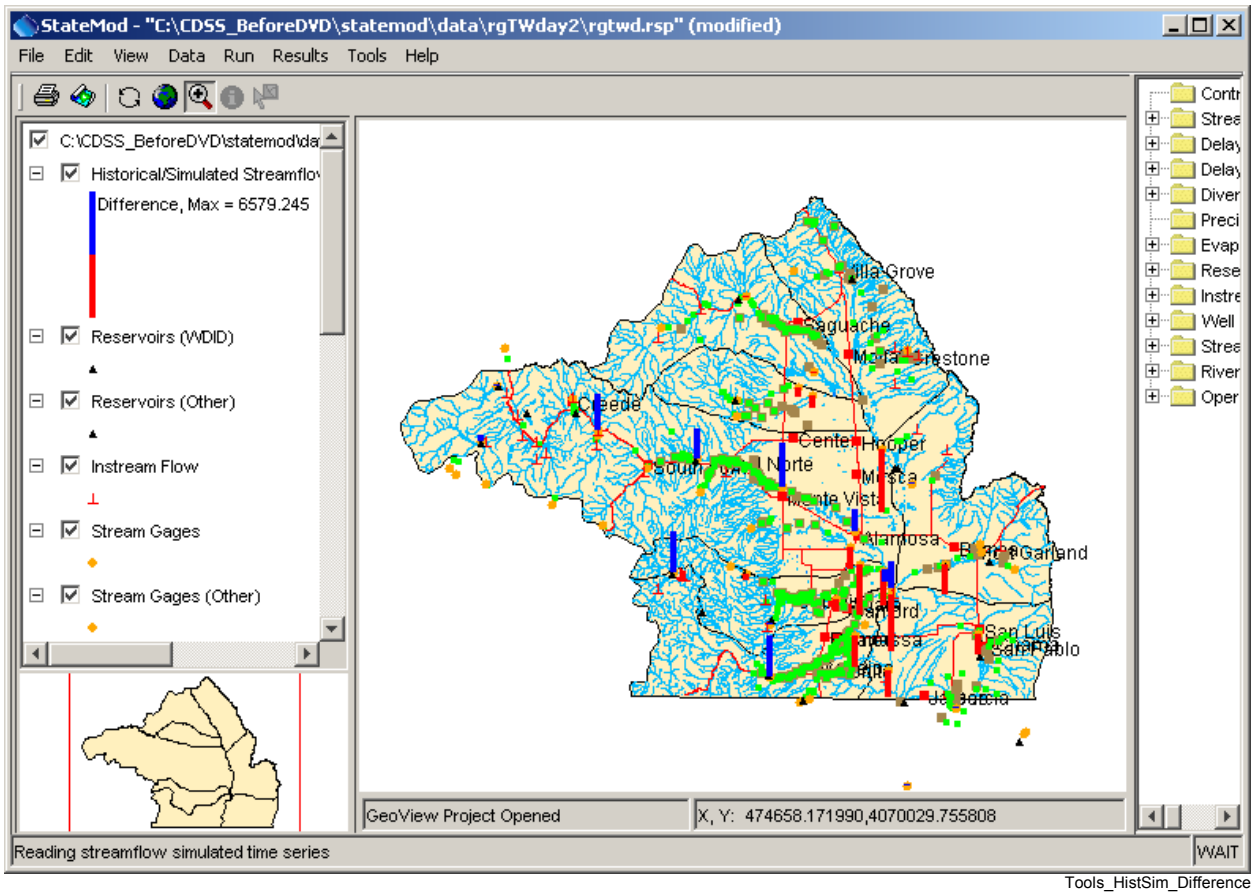
Tools_Add_SummaryMapLayer

**Dialog to Configure the Summary Map Layer**

## 9.3 Add Historical/Simulated Streamflow Difference to Map

The ***Tools…Add Historical/Simulated Streamflow Difference to Map*** menu creates a new map layer that indicates absolute differences between historical and simulated streamflow, for stream gage stations. The difference is computed as the average annual historical streamflow minus the average annual simulated streamflow, using calendar years, for the period of the historical streamflow time series (both time series must have values in a year to be considered in the calculations). This tool is useful for model calibration, in particular to identify locations were large absolute differences between historical data and simulation results are occurring. The tool may require several minutes to process all the data.

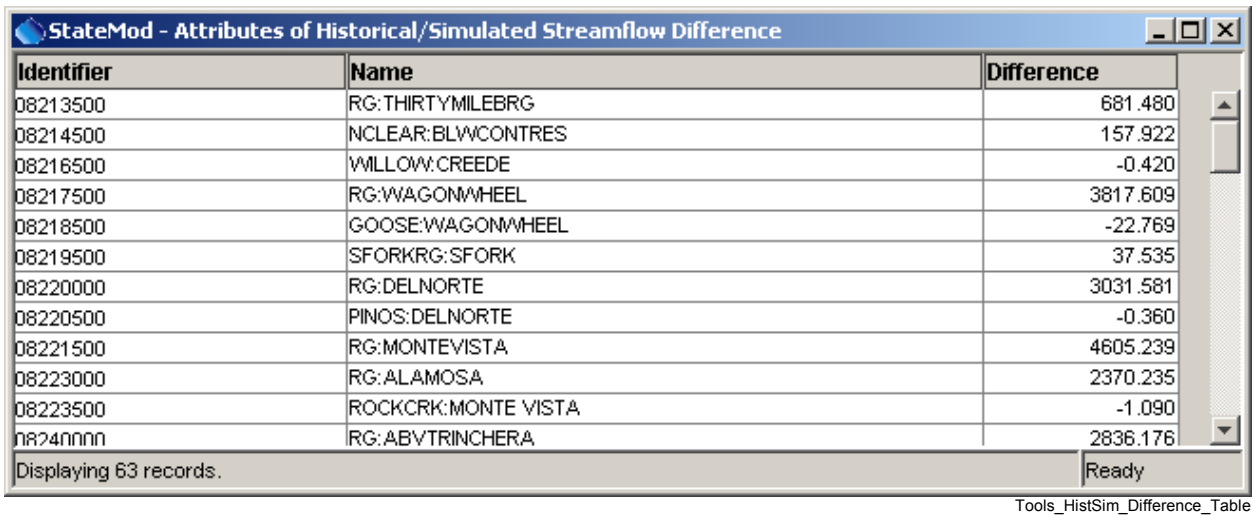
The following figure illustrates the map interface after using the tool. When viewed in color, blue bars indicate cases where the historical time series had larger values and red indicate cases where simulated time series had larger values. Warnings may be shown if streamflow stations are not present in the streamflow station layers on the map.




Tools_HistSim_Difference

**Historical/Simulated Difference Map Layer**

The map layer can be selected and its attribute table viewed (right click on the layer name in layer list and select ***View Attribute Table***):




| Identifier | Name | Difference |
|---|---|---|
| 08213500 | RG:THIRTYMILEBRG | 681.480 |
| 08214500 | NCLEAR:BLWCONTRES | 157.922 |
| 08216500 | WILLOW:CREEDE | -0.420 |
| 08217500 | RG:WAGONWHEEL | 3817.609 |
| 08218500 | GOOSE:WAGONWHEEL | -22.769 |
| 08219500 | SFORKRG:SFORK | 37.535 |
| 08220000 | RG:DELNORTE | 3031.581 |
| 08220500 | PINOS:DELNORTE | -0.360 |
| 08221500 | RG:MONTEVISTA | 4605.239 |
| 08223000 | RG:ALAMOSA | 2370.235 |
| 08223500 | ROCKCRK:MONTE VISTA | -1.090 |
| 08240000 | RG:ABVTRINCHERA | 2836.176 |

Tools_HistSim_Difference_Table

**Historical/Simulated Difference Map Layer Attribute Table**

## 9.4 Add Historical/Simulated Streamflow Difference (Percent) to Map

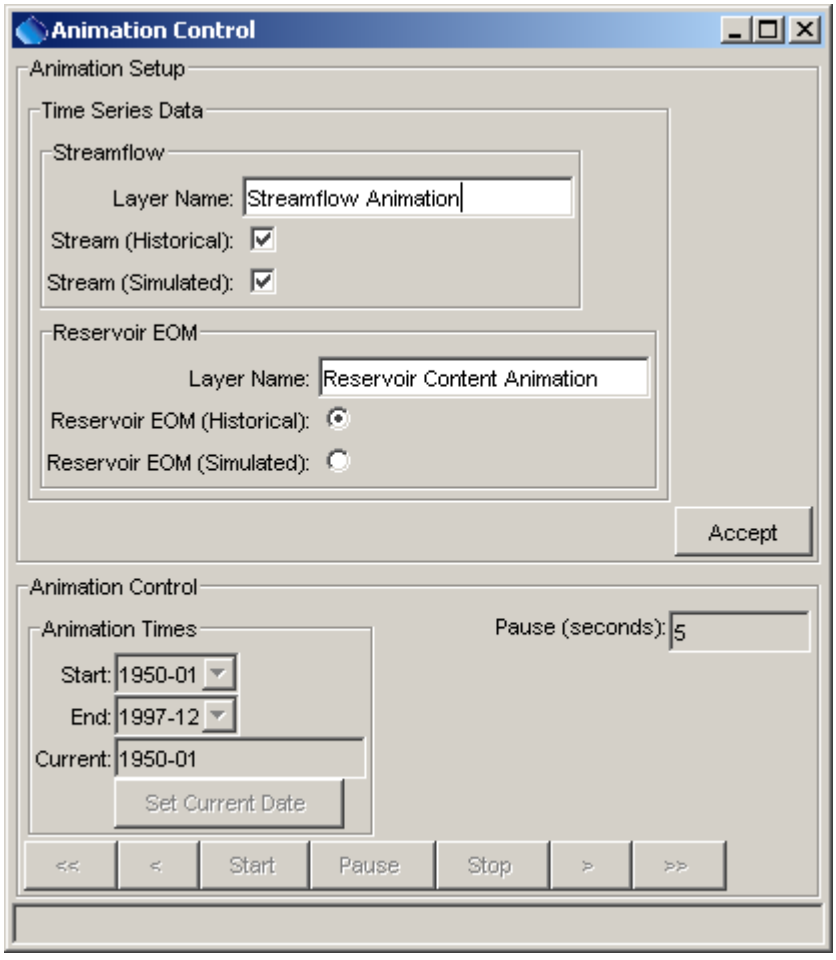
The ***Tools…Add Historical/Simulated Streamflow Difference (Percent) to Map*** menu creates a new map layer that indicates percent (0 to 100) differences between historical and simulated streamflow, for stream gage stations. See the previous section for background information and example output. The percent is calculated by taking the average annual difference divided by the average annual historical streamflow. Displaying percent may provide a better overview of differences.

## 9.5 Add Animation Map Layer

The ***Tools…Add Animation Map Layer*** adds a layer to the map and provides an interface for animating the symbols in the layer. Streamflow levels are shown as bars and reservoir levels are shown as "teacup" symbols, where the size of the symbols is relative to the reservoir capacity.

When the tool is started, the StateMod GUI collects necessary data for the animation. This may require several minutes and progress is reported in the status area at the bottom of the main window.

A map tool controls the animation itself. After initial data collection the following dialog will be shown to initialize the animation:

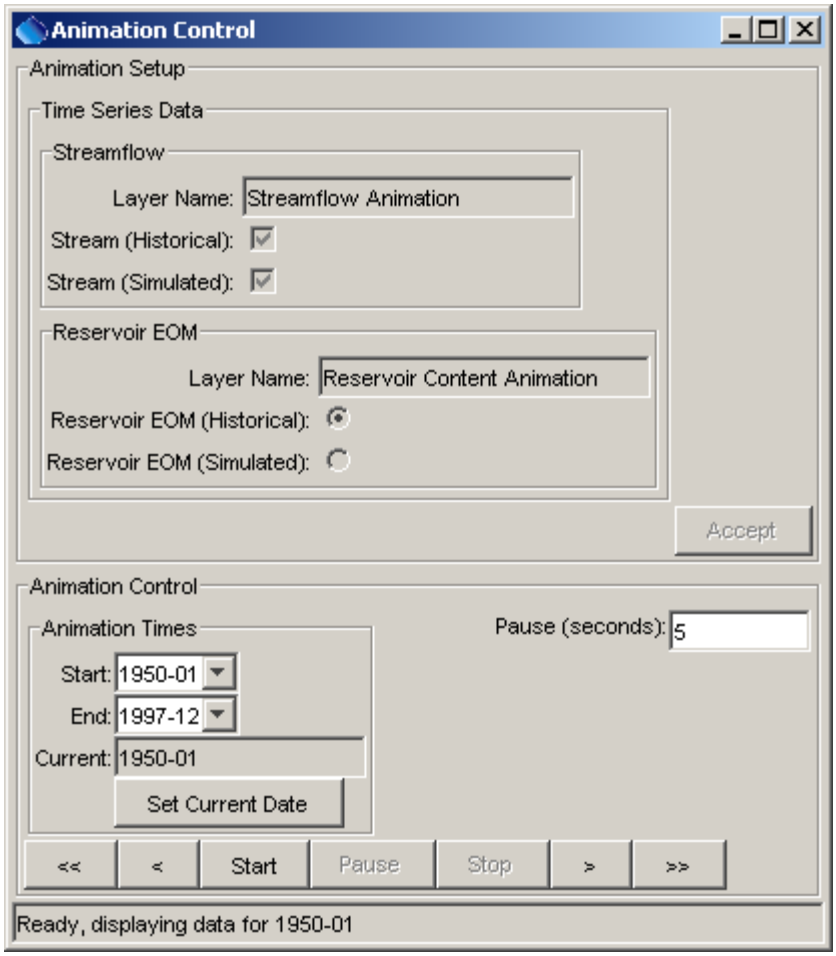



Tools_Animation1

**Animation Setup – Select Reservoir Time Series to Animate**

Historical streamflow (taken from the historical streamflow input file) and simulated streamflow (taken from the `River_Outflow` parameter in the binary file) can be shown as bars on the map. Either reservoir historical data (taken from end of month content time series) or simulated values (taken from the `Sim_EOM` parameter in the StateMod output binary file) can be displayed.

After the time series data input is selected, press ***Accept*** to continue with the animation setup. The StateMod GUI will then attempt to build a map layer for the selected layers, and will associate the time series with the points in the layer. The scaling of map symbols takes into account the full period of time series data. Additional configuration of the animation period can occur, as shown in the following figure.

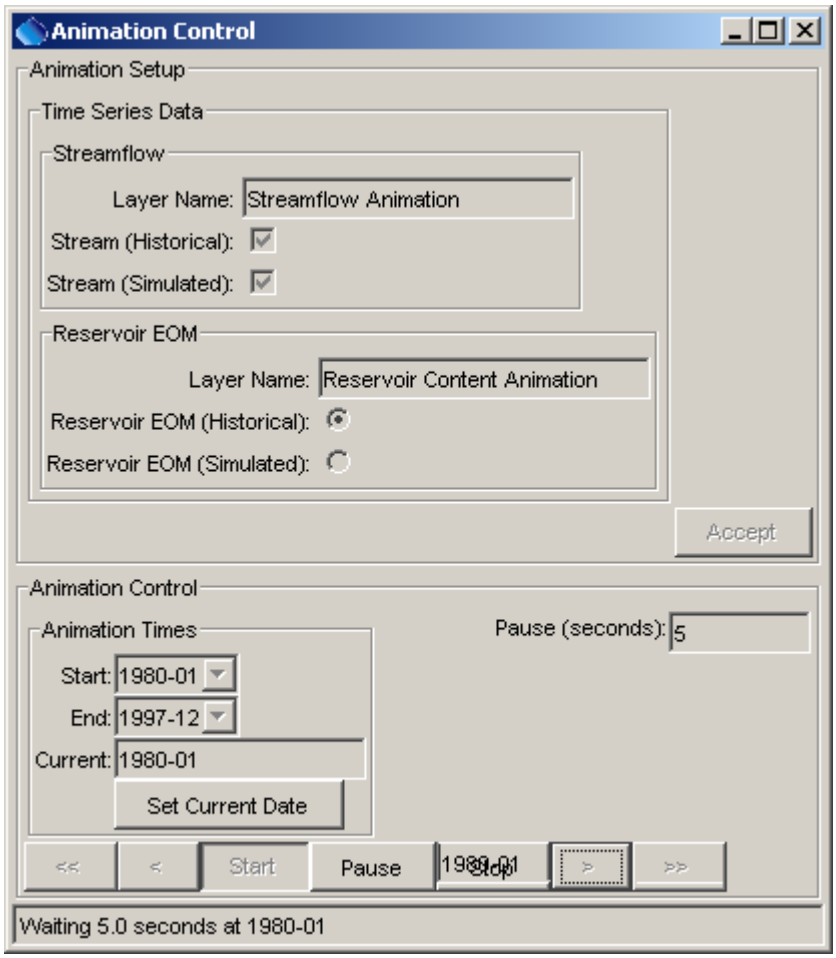



Tools_Animation2

**Animation Setup – Define Period to Run**

Define the period for the animation and set the start date. Press ***Start*** to start the animation. The **<<** button will set the current date to the start date. The **<** button will go to the previous date. The ***Pause*** button will pause the animation. The **>** button will advance to the next date. The **>>** button will set the current date to the end date. The ***Stop*** button will stop the animation.

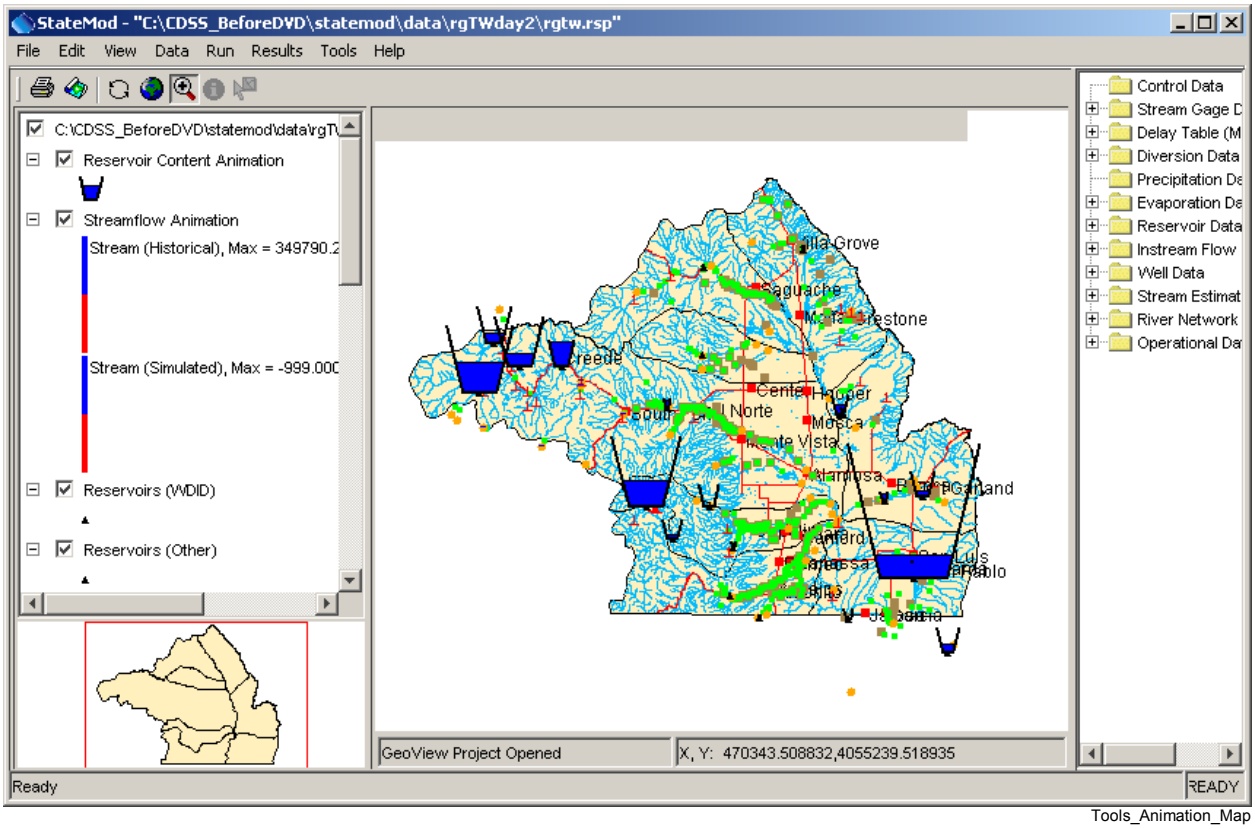
The following figure illustrates how the status bar at the bottom of the window indicates animation progress.




Tools_Animation3

**Animation Control While Running Animation**

A map layer is added to the layer list during animation setup. This layer is treated similar to other layers. However, right clicking on the layer shows the ***Show Animation Control*** dialog (as illustrated above), which is used to control the animation. The following figure illustrates the map during animation. Refer to the animation control for dates associated with the map display.

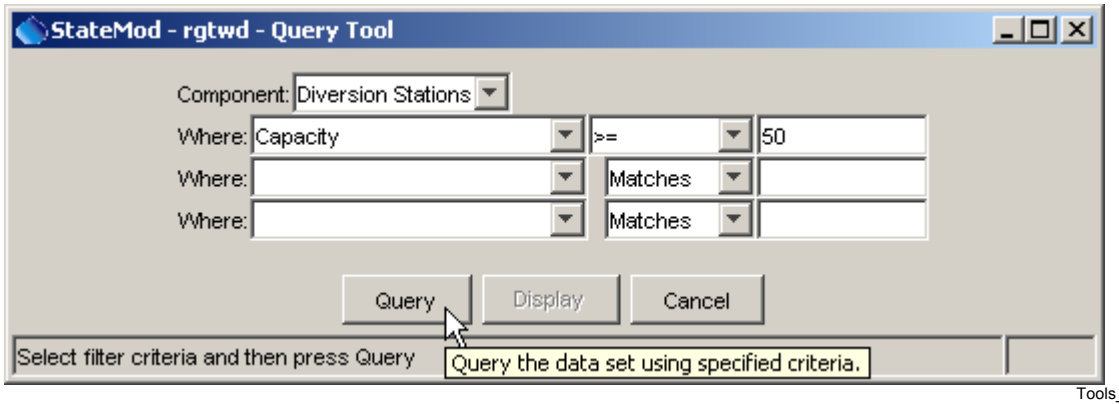



Tools_Animation_Map

**StateMod GUI Map Display During Animation**

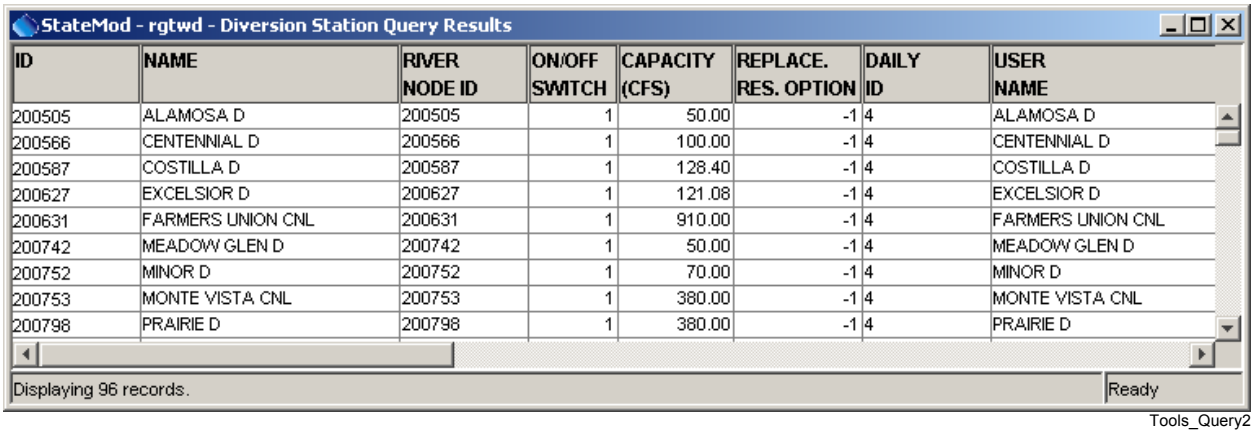
## 9.6 Query Tool – Find Stations that Match Criteria

The ***Tools…Query Tool*** menu displays a query dialog that allows selection of stations based on multiple query criteria. This is similar to a database query and can be used for quality control and analysis. This tool was implemented as a prototype and has limited functionality. The following figures illustrate an example query and results.




Tools_Query1

**Query Tool Before Executing Query**




| ID | NAME | RIVER NODE ID | ON/OFF SWITCH | CAPACITY (CFS) | REPLACE. RES. OPTION | DAILY ID | USER NAME |
|---|---|---|---|---|---|---|---|
| 200505 | ALAMOSA D | 200505 | 1 | 50.00 | -1 | 4 | ALAMOSA D |
| 200566 | CENTENNIAL D | 200566 | 1 | 100.00 | -1 | 4 | CENTENNIAL D |
| 200587 | COSTILLA D | 200587 | 1 | 128.40 | -1 | 4 | COSTILLA D |
| 200627 | EXCELSIOR D | 200627 | 1 | 121.08 | -1 | 4 | EXCELSIOR D |
| 200631 | FARMERS UNION CNL | 200631 | 1 | 910.00 | -1 | 4 | FARMERS UNION CNL |
| 200742 | MEADOW GLEN D | 200742 | 1 | 50.00 | -1 | 4 | MEADOW GLEN D |
| 200752 | MINOR D | 200752 | 1 | 70.00 | -1 | 4 | MINOR D |
| 200753 | MONTE VISTA CNL | 200753 | 1 | 380.00 | -1 | 4 | MONTE VISTA CNL |
| 200798 | PRAIRIE D | 200798 | 1 | 380.00 | -1 | 4 | PRAIRIE D |

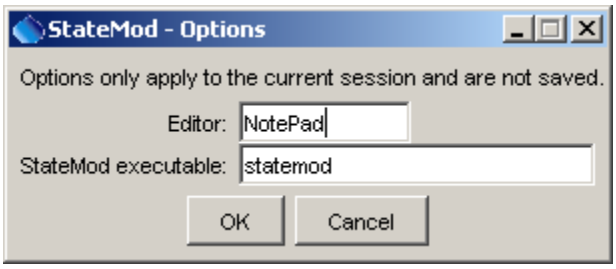
Tools_Query2

**Query Tool Results**

The results can be sorted by right clicking on the column headings and exported by right clicking on the worksheet and selecting from popup menu choices.

## 9.7 StateMod GUI Options

The ***Tools…Options*** menu allows StateMod GUI session properties to be changed:



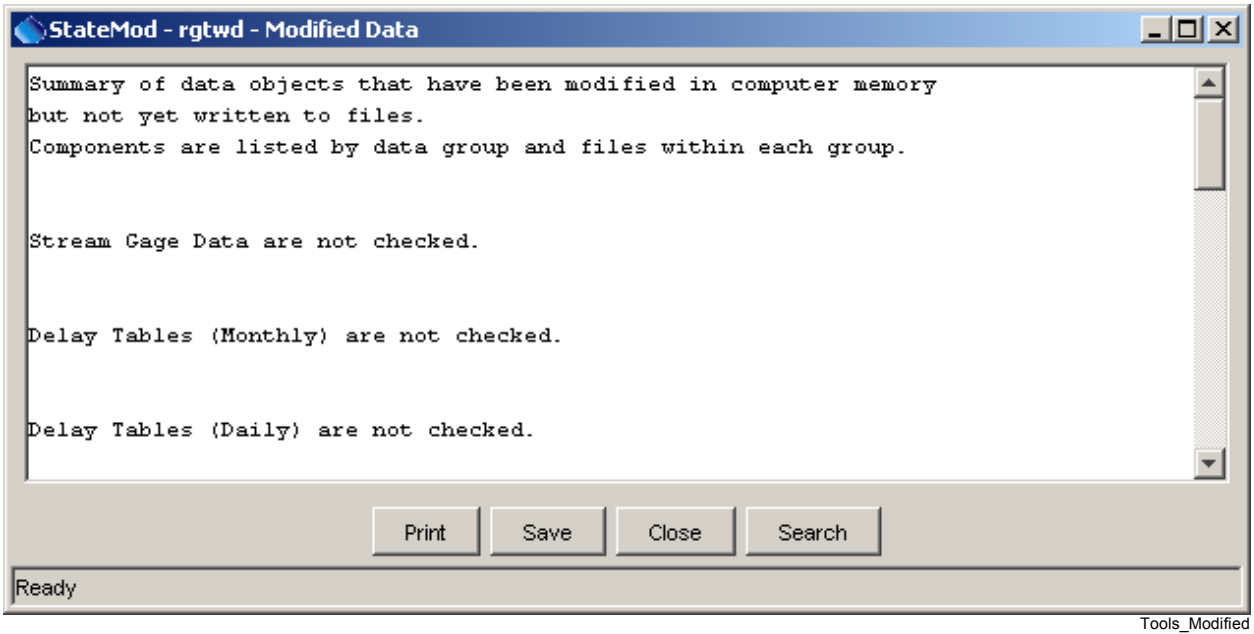

Tools_Options

**StateMod GUI Options**

The settings cannot currently be saved to a file. Changes from the defaults are therefore in effect only during the current session. The editor is used when viewing StateMod output files (***Results…Output Files***). The StateMod executable is normally set to the name that will be found if the `PATH` environment variable is searched. However, it may be changed, for example, if several StateMod versions are being run and a specific executable is to be used. An absolute path can be entered if necessary.

## 9.8 Listing Modified Data

The ***Tools…Diagnostics Report - Modified Data*** menu displays a list of data that have been modified. This tool was developed to aid software developers in properly handling user edits but has limited functionality. Detecting modified data is important because the StateMod GUI must write modified files before the StateMod software will recognize the data changes. The following figure illustrates output from the tool:

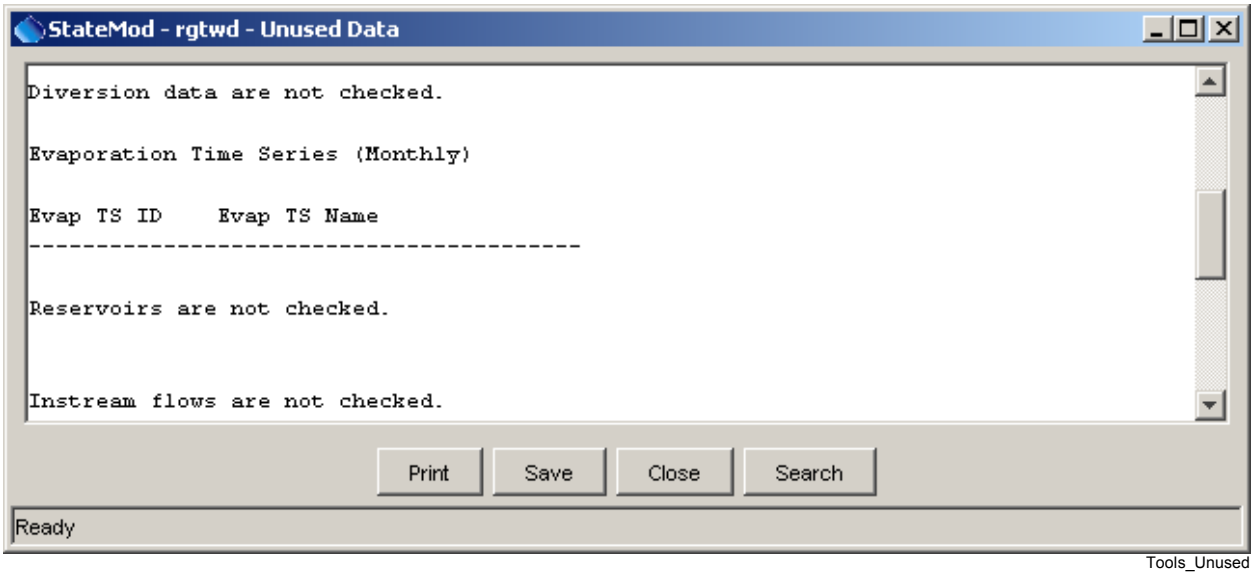



Tools_Modified

**Report Listing Modified Data**

## 9.9 Listing Unused Data

The ***Tools…Diagnostics Report – Unused Data*** menu displays a list of stations that are not used in the data set. This tool was developed to aid software developers and has limited functionality. The following figure illustrates output from the tool:

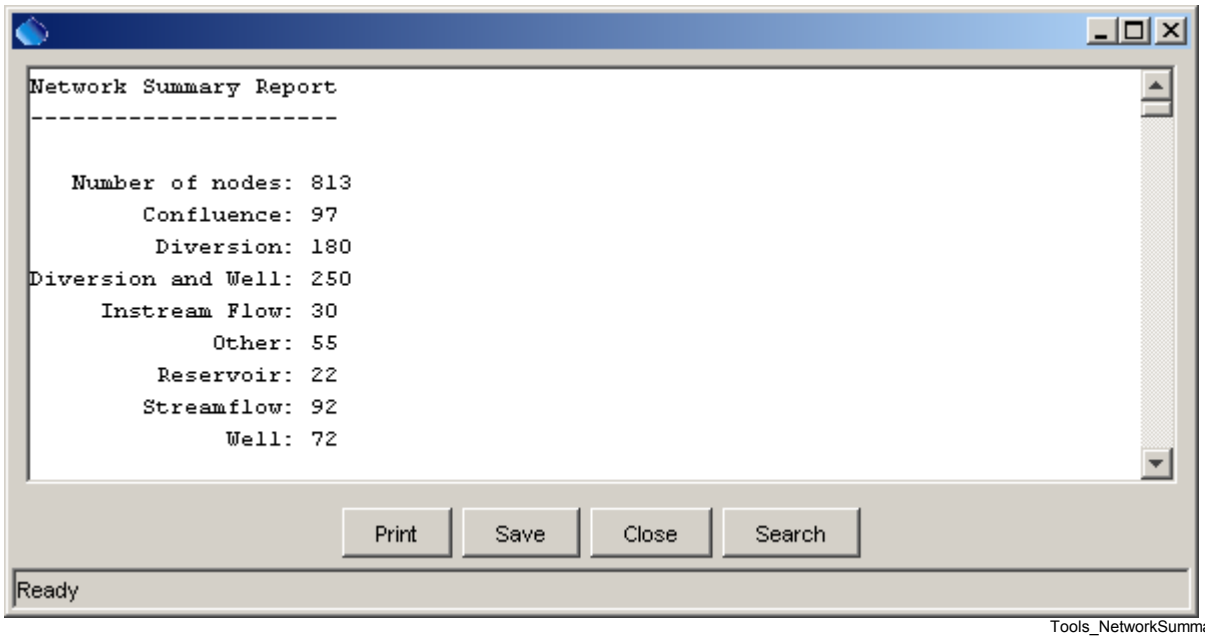



Tools_Unused

**Report Listing Unused Data**

## 9.10 Network Summary

The ***Tools…Network Summary*** tool provides a summary of each node type in the network. This tool was implemented to troubleshoot networks and has limited functionality. The following figure illustrates the tool output:




Tools_NetworkSummary

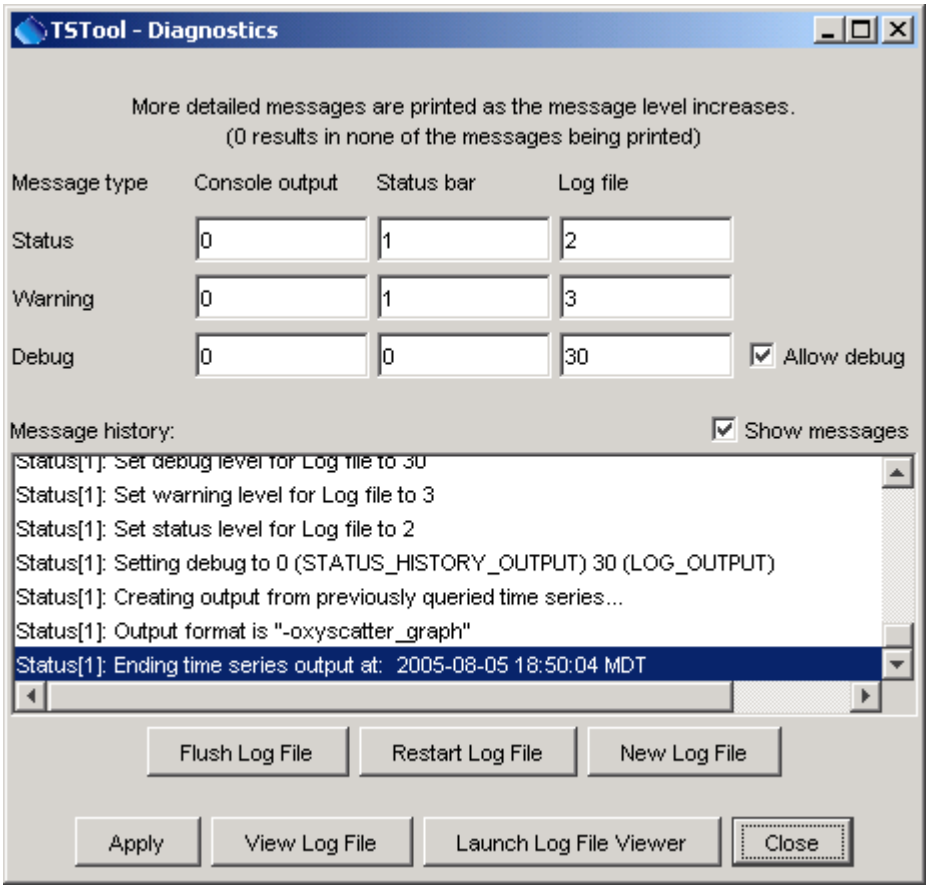
**Network Summary Report**

## 9.11 Diagnostics Tools

Diagnostics features are useful for troubleshooting. When an error occurs, a small warning dialog may be displayed and the message is also recorded in the StateMod GUI log file.

### 9.11.1 Diagnostics Settings

The ***Tools…Diagnostics*** menu item displays the ***Diagnostics*** dialog, which is used to set message levels and view messages as the application runs. The ***Diagnostics*** dialog (see the following figure) can be used to evaluate a problem.




Diagnostics

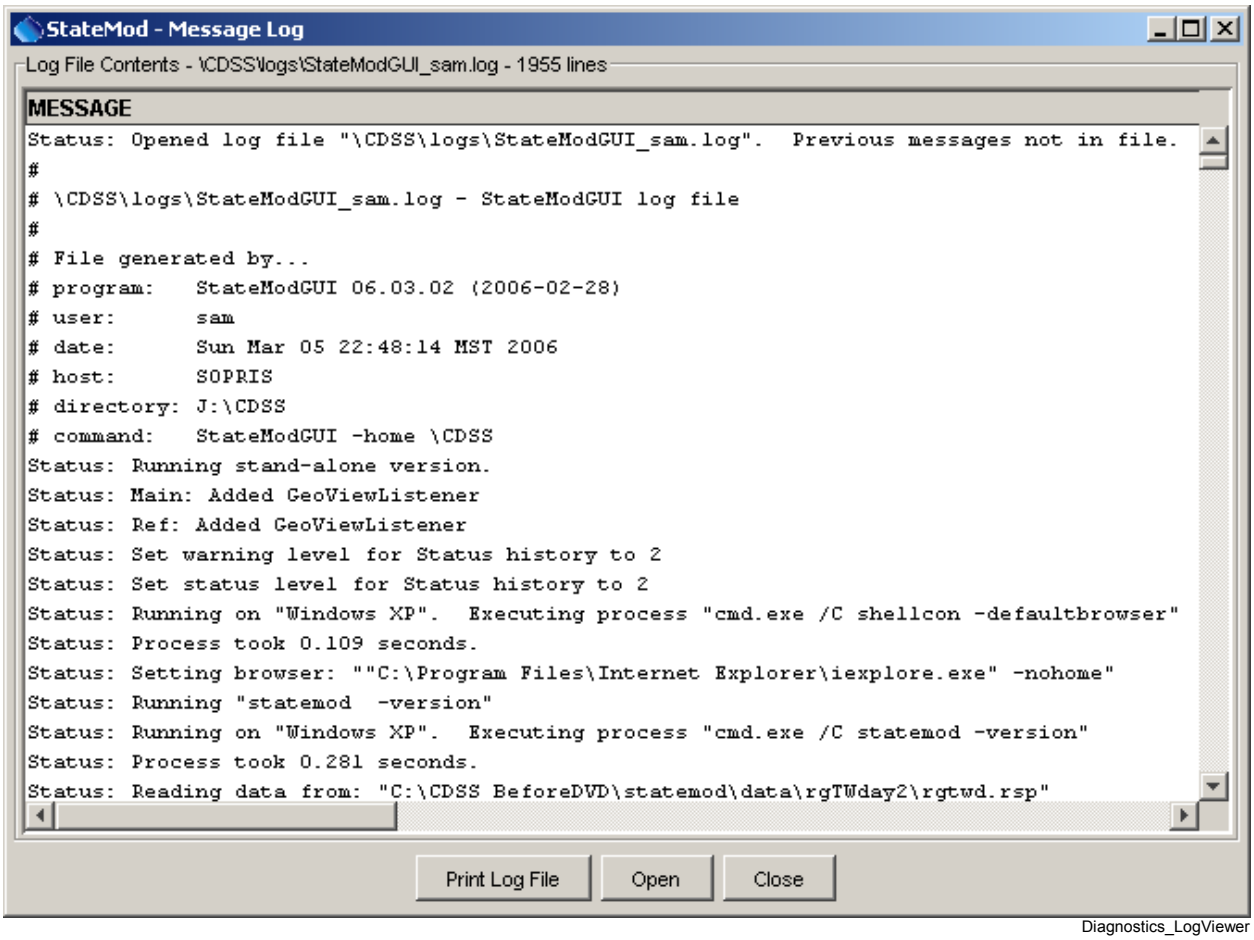
**Diagnostics Interface**

The settings at the top of the dialog are used to specify the level of detail for messages printed to the console window, the status area at the bottom of the main window (and the ***Diagnostics*** dialog), and the log file. The log file contains warning, status, and debug messages, many of which are not normally displayed in the main interface. The log file is created in the logs directory under the installation directory. The ***Diagnostics*** interface features are as follows:

| | |
|---|---|
| ***Status, Warning, Debug*** | Enter integer values, with larger numbers resulting in more output and slower performance. Zero indicates no output. If troubleshooting, a good guideline is to set the debug level to 10 or 30 (and select the ***Allow Debug*** checkbox). The default settings are often enough for normal troubleshooting and result in good software performance. |
| ***Allow Debug*** | Select to enable debug messages. Turning on debug messages will significantly slow down the software. |
| ***Show Messages*** | Select to display messages in the ***Diagnostics*** window. |
| ***Flush Log File*** | Force messages to be written to the log file. Messages can be buffered in memory and may not otherwise immediately be written to the log file. |
| ***Restart Log File*** | Restart the log file. This is useful if a long session has occurred and troubleshooting will occur on new actions. |
| ***New Log File*** | Open a new log file, with a new name. |
| ***Apply*** | Apply the settings in the ***Diagnostics*** dialog. |
| ***View Log File*** | View the log file in an integrated window. The ***View Log File*** button will be enabled if the log file has been opened. |
| ***Launch Log File Viewer*** | View the log file using a viewer from the operating system. On Windows computers, Notepad will be used. |
| ***Close*** | Apply the settings in the ***Diagnostics*** dialog and close the window. |

### 9.11.2 Diagnostics – View Log File

The ***Tools…Diagnostics – View Log File*** menu item displays the integrated log file viewer. Selecting this menu item is equivalent to selecting the ***View Log File*** button in the ***Diagnostics*** dialog. The log file viewer is shown in the following figure:




Diagnostics_LogViewer

**Log File Viewer Window**

The log file messages can be scrolled. To find a string in the log file, right-click and select the ***Find*** menu item. The information in the log file can also be copied and pasted into email when contacting support.

This page is intentionally blank.

# 10 Troubleshooting

Version 07.04.00, 2013-04-18

This chapter discusses how to troubleshoot problems when using the StateMod GUI.

To facilitate troubleshooting, the StateMod GUI creates a log file under the main installation directory (e.g., *\CDSS\StateModGUI-Version\logs\StateModGUI_USER.log*) containing status and warning messages from the run. This log file can be viewed using the ***Tools…Diagnostics – View Log File*** menu. A log file from StateMod is also created with a base name matching the response file that was last run. The log files can be viewed within the GUI with other StateMod output files (***View…Output Files***). Note that StateMod reuses the same log file name each time it is run so a log file may be overwritten as multiple reports are made.

The most common problems are software configuration (see the **Installation and Configuration Appendix**) and data errors (see **Chapter 6 – Running StateMod** for information about running a data check). Other problems should be reported to the CDSS developers at cdss@state.co.us. Checks have been implemented for common errors, but you may need to refer to the log file to determine the nature of a problem. Also see the **Release Notes** for software limitations.

In general, when running the StateMod GUI, warnings are displayed in dialogs and the log file. If the run has been successful, you will only see status messages in the files, with no warnings or errors. Status messages provide useful information, such as indicating the progress of a run.

Due to the complexity of the CDSS data and the needs of modelers, user and data, errors can occur for a number of reasons. The following table summarizes common errors and their fixes. **If modeling in the CDSS environment, you may need to verify HydroBase data using StateView and other CDSS tools**.

**StateMod GUI Errors and Possible Solutions**

| Error | Possible solutions |
|---|---|
| The GUI does not properly edit StateMod data. | The GUI software may not be consistent with the model. These issues are being evaluated in order to ensure consistency between the model and GUI. It is recommended that modelers use a data-centered approach using StateDMI and TSTool software to process model input files. |

# Appendix: StateMod GUI Installation and Configuration for CDSS

07.04.00, 2013-04-18

## 1. Overview

This appendix describes how to install the StateMod GUI in Colorado's Decision Support Systems (CDSS) environment. Installing the StateMod GUI will also install the StateMod model and related utility programs, which can be further updated if separate updates have been provided.

## 2. StateMod GUI File Locations

The State of Colorado's CDSS consists of the HydroBase database, modeling, and data viewing/editing applications. In this environment, StateMod model files are usually prepared in an automated fashion using the StateDMI and TSTool software. The StateMod GUI is used to review data sets, make minor changes to the data sets, and prepare graphical data products from model input and output.

Locations of StateMod GUI software files are as shown below, where Version is similar to 07.04.00. The version allows multiple versions of the StateMod GUI and associated StateMod software to be installed at the same time, facilitating new development and allowing versions to be archived for use with old data sets.

| | |
|---|---|
| *\CDSS\StateModGUI-Version* | Top-level install directory (install home). |
| *bin* | Software directory for *StateModGUI* and StateMod software files. |
| **.jar* | Java software files used by the GUI. |
| *SmDelta.exe* | Program to generate average and difference statistics from StateMod output. |
| *StateMod-Version.exe* | StateMod model executable used by default by the StateMod GUI. |
| *StateModGUI.bat* | Batch file to run the StateMod GUI using the JRE software. This may be phased out since the *StateModGUI.exe* is now used. |
| *StateModGUI.exe* | StateMod GUI executable. |
| *doc\UserManual* | Main documentation directory for the StateMod GUI. |
| *StateModGUI.pdf* | StateMod GUI documentation as PDF. |
| *Training* | Training material PowerPoint and related files. |
| *jre_** | Java runtime environment that executes the software. |
| *logs* | Location of log files. |
| *system* | Location of system files used for configuration. |
| *DATAUNIT* | Data units. |
| *OperatingRules.** | Files used to provide operating rules information in general way. |
| *StateModGUI.cfg* | StateMod GUI configuration file. |

## 3. Installing the StateMod GUI

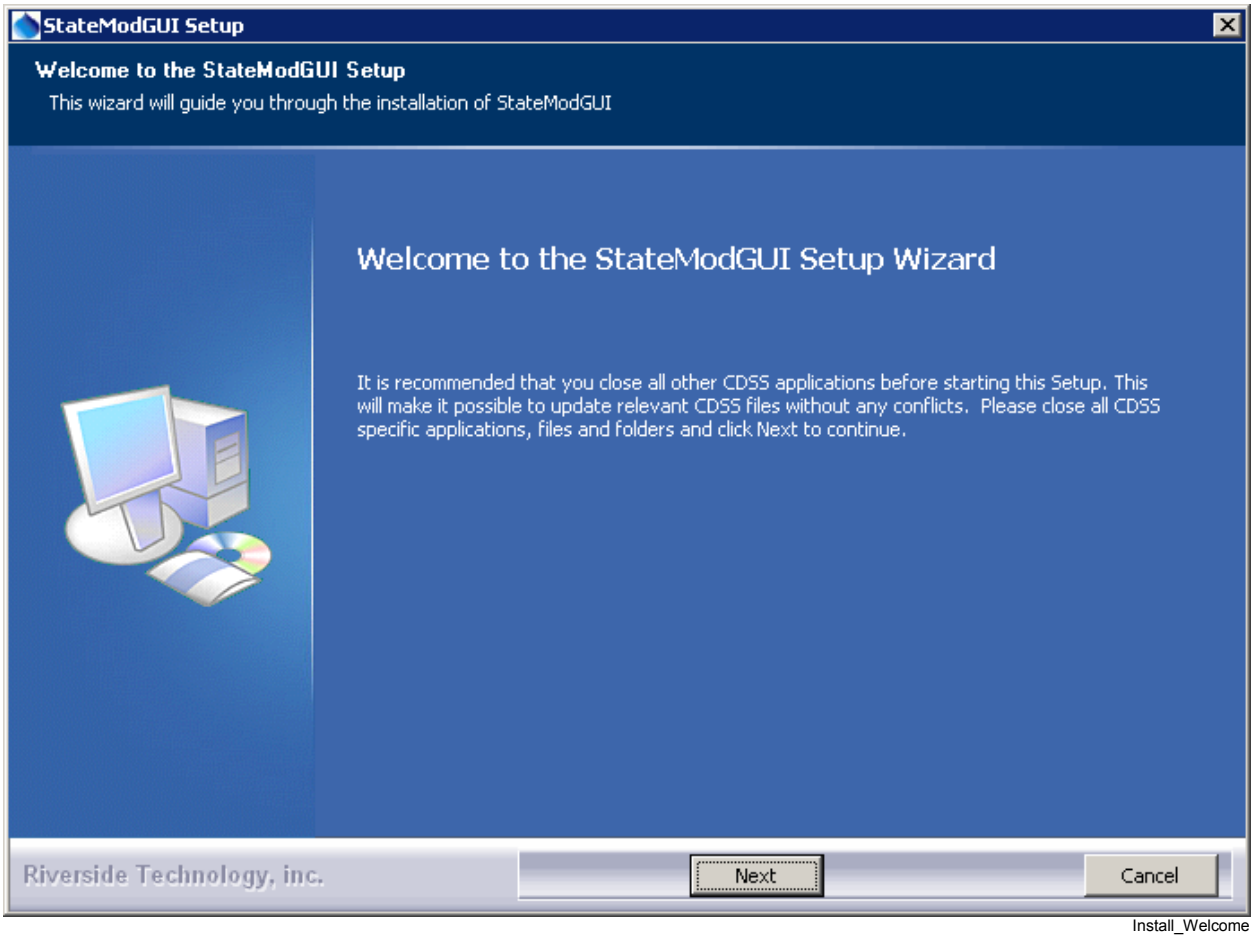
The StateMod GUI can be installed with other CDSS software, or as a separate installation (e.g., to perform StateMod modeling without the support of the State of Colorado's HydroBase and related tools). In both cases, is recommended that the default file organization is used. The StateMod model software is installed with the StateMod GUI, although newer versions of the model may be provided separately.

Use the following instructions to install the StateMod GUI using the *StateModGUI_CDSS_Version_Setup.exe* installer program, for example if StateMod GUI software was downloaded from the CDSS web site (http://cdss.state.co.us):

1. Run the *StateModGUI_CDSS_Version_Setup.exe* file by selecting from Windows Explorer, the ***Start…Run…*** menu, or from a command shell. The setup filename will include a version number (e.g., *StateModGUI_CDSS_07.04.00_Setup.exe*).

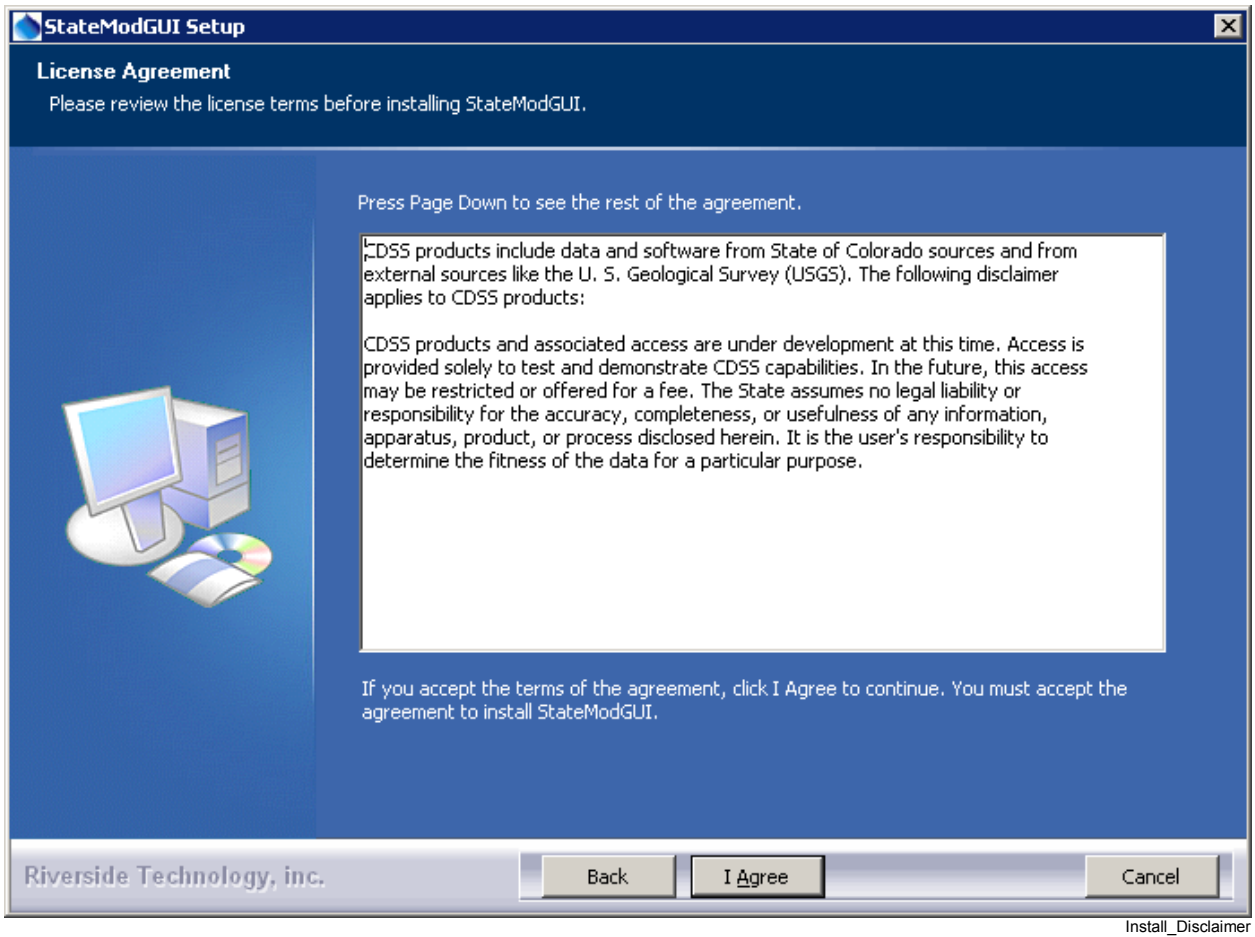
   You must be logged into the computer using an account with administrator privileges. Otherwise, a warning will be displayed.

If you have administrative privileges, the following welcome will be displayed, and the installation can continue:




Install_Welcome

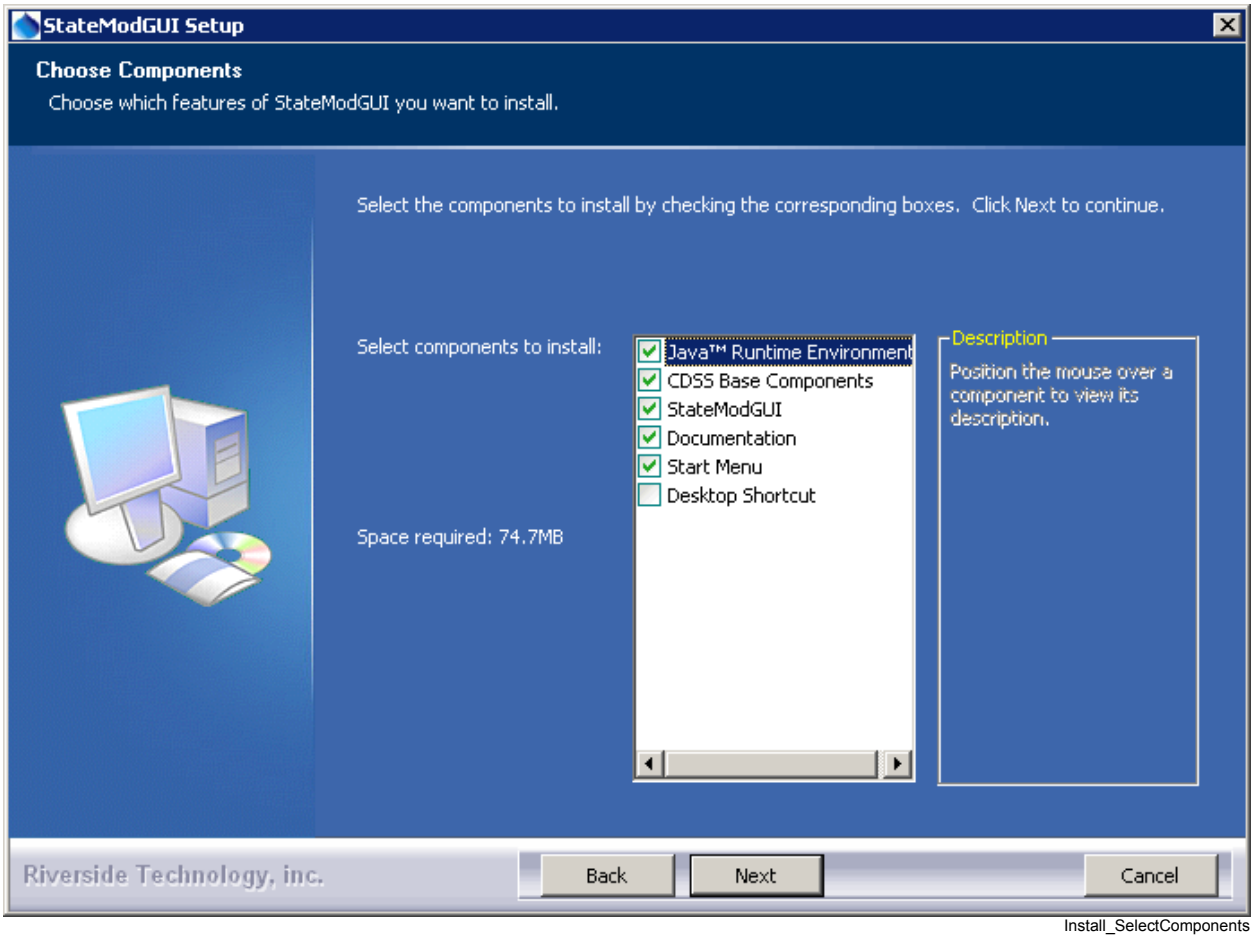
Press ***Next*** to continue with the installation.



Install_Disclaimer

The StateMod GUI is distributed as a component of CDSS with no license restrictions. However the disclaimer must be acknowledged. Press ***I Agree*** to continue with the installation.

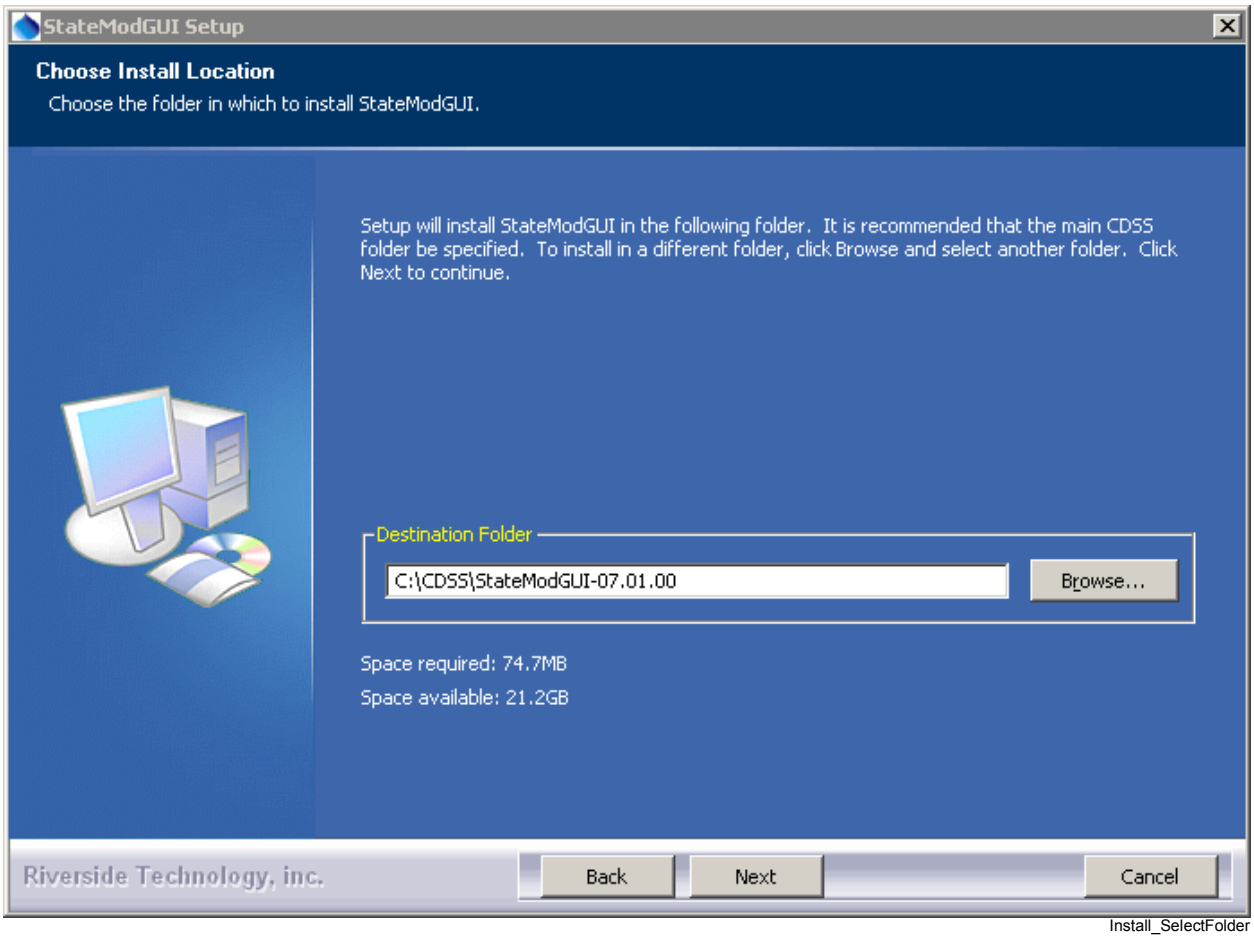
2. Several components can be selected for the install as shown in the following dialog. Position the mouse over a component to see its description.




Install_SelectComponents

Select the components to install and press ***Next***.

3. The following dialog is then shown and is used to select the installation location for the StateMod GUI. To be consistent with other CDSS components, select the versioned folder under the main CDSS folder, as shown by default. The following dialog will display the CDSS install location if the CDSS Base component has been previously installed:

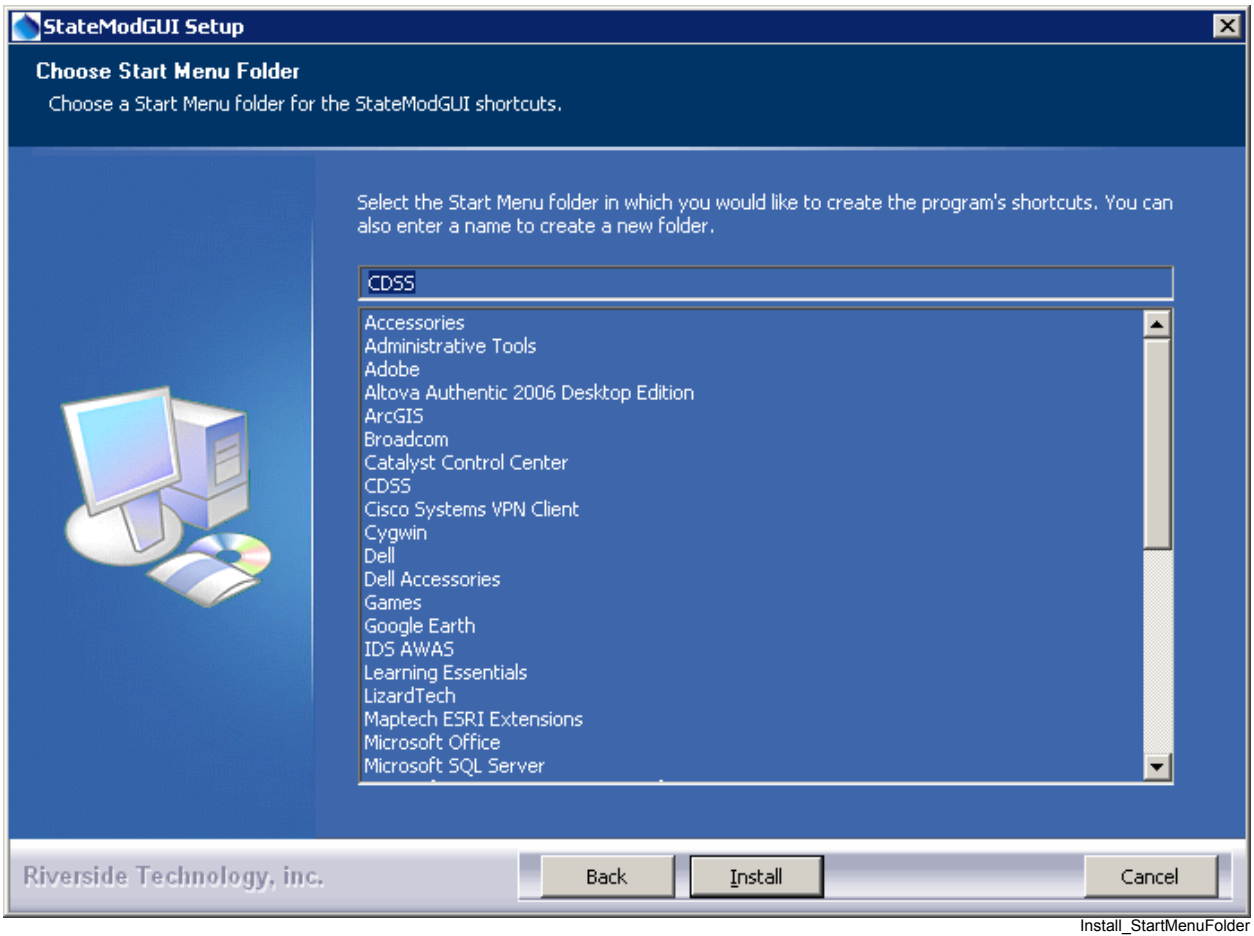



Install_SelectFolder

After selecting the install location, press ***Next***.

Note that this location will be saved as a Windows registry setting (`HKEY_LOCAL_MACHINE\Software\State of Colorado\StateModGUI-Version\Path`) to allow future updates to check for and default to the same install location, and to allow the standard software uninstall procedure to work correctly.

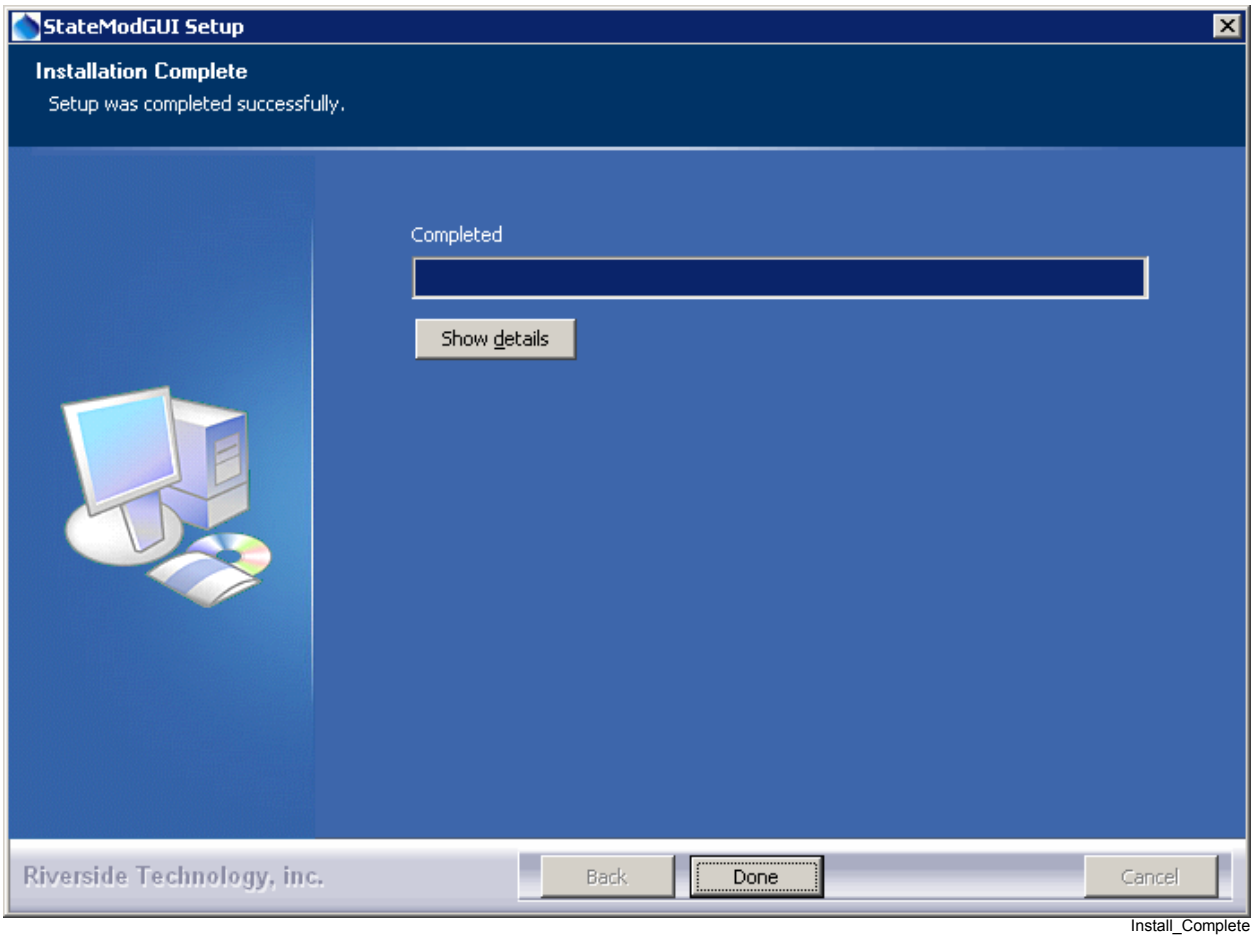
4. The following dialog will be shown to select the menu for the software:




Install_StartMenuFolder

After selecting the folder, press ***Install***.

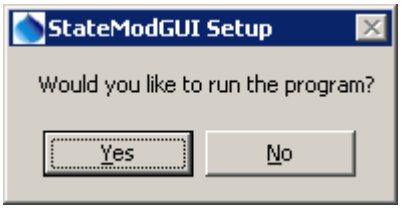
5. The following dialog will show the progress of the installation:




Install_Complete

Press ***Show details*** to see the files that were installed or press ***Done*** to continue.

6. The following dialog will then be shown asking whether the StateMod GUI software should be run:




Install_RunStateModGUIQuestion

Press ***Yes*** to run the software or ***No*** to exit the installation procedure.

7. A reboot is not required to use the software from the ***Start*** menu.

### 3.1 Installing the StateMod GUI on a File Server

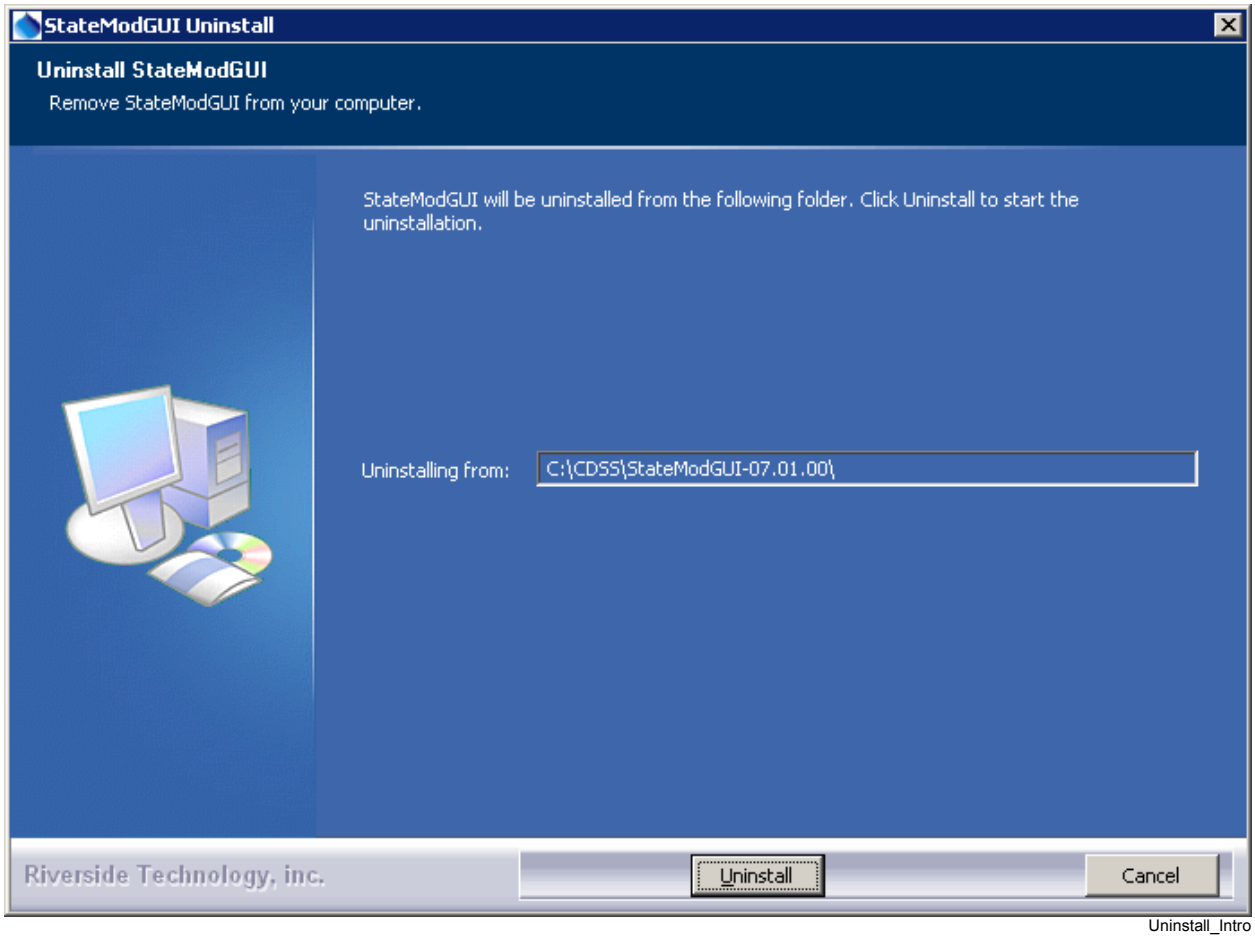
The StateMod GUI can be installed on a file server, which allows software updates to be made in one location, thereby eliminating the need to install software on individual machines. For this type of installation, it is recommended that all computers that access the software must have similar configuration, including network configuration. The standard installer described in this documentation focuses on individual installs on user computers. To install the StateMod GUI software on a server and make available to other computers, perform the following (this is typically performed by system administrators):

1. Run the *StateModGUI_CDSS_Version_Setup.exe* installer on the server, using the steps as described above. During installation specify the CDSS home using a drive letter and path for the server or specify a Universal Naming Convention (UNC) path (e.g., \\CDSSServer\CDSS\StateModGUI-Version). The home directory of the software is determined at runtime to be that corresponding to the parent folder of the *StateModGUI.exe* file. Alternatively, copy the installed files from a local computer installation to a server (in this case menus and shortcuts will not be configured on the server).
2. The menus and shortcuts will only be configured for the computer from which the installation was run. Therefore, menus and shortcuts for other computers will need to be manually configured.

If the StateMod GUI has been installed on a local computer and it is also available on the network, the network version can be run by running the *NetworkInstallHome\StateModGUI-Version\bin\StateModGUI.exe* program.

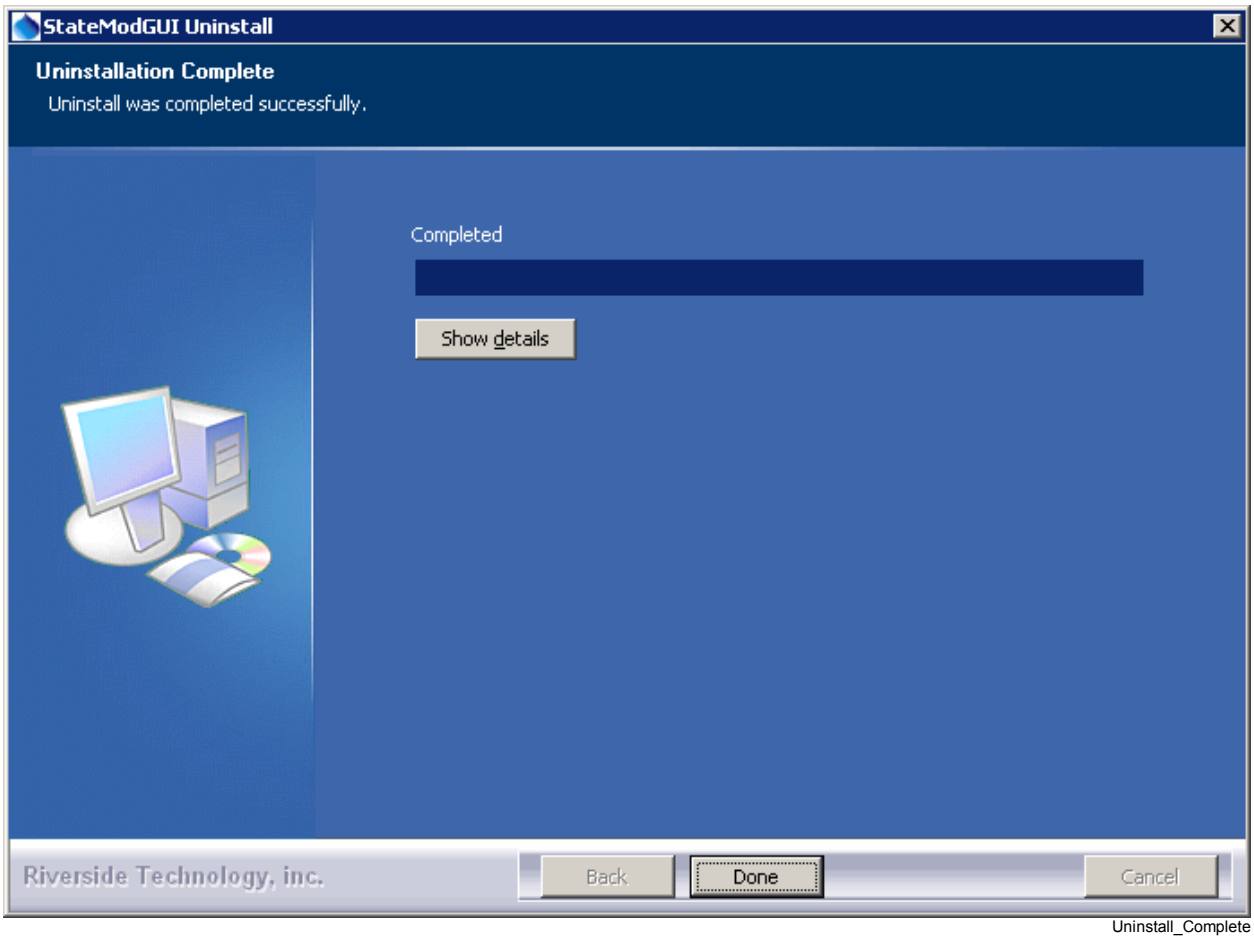
## 4. Uninstalling StateMod GUI Software

To uninstall the StateMod GUI (and StateMod model and utilities) software, select ***CDSS…Uninstall…StateModGUI-Version*** from the Start menu and confirm the uninstall.




Uninstall_Intro

Press ***Uninstall*** to uninstall the software.



Uninstall_Complete

Press ***Show details*** to see the list of files that were removed. Press ***Done*** to exit the uninstall.

## 5. Running the StateMod GUI

The StateMod GUI can be started in several ways as described below.

### 5.1 CDSS Menu

The ***Start…All Programs…CDSS…StateModGUI-Version*** (or ***Start…Programs…StateModGUI-Version***) menu can be used to start the software. This runs the *InstallHome\bin\StateModGUI.exe* software.

### 5.2 Command Line Executable

The installation process DOES NOT add the *InstallHome\bin* folder to the `PATH` environment variable, primarily because different versions of the software can be installed and it can be difficult to automatically determine which one should be listed first in the `PATH`. To add a folder to the PATH, edit the information accessible via ***Start…Control Panel…System…Advanced…Environment Variables***.

Previous StateMod GUI installers (prior to 7.00.00) modified the path and such changes may still be in effect on the computer. Uninstall the earlier version and/or edit the `PATH` environment variable to remove this configuration.

### 5.3 StateModGUI Batch File

The StateMod GUI was previously started with and can still be run with the *InstallHome\bin\StateModGUI.bat* file, for example to support troubleshooting. In this case, the file name *StateModGUI.bat* must be fully specified because running *StateModGUI* will result in the executable program being run (see previous section).

## 6. StateMod GUI Configuration Files

The StateMod GUI requires minimal configuration after installation. This section describes StateMod GUI configuration files that can be customized for a system.

### 6.1 Data Units File

The *system\DATAUNIT* file under the main installation directory contains data unit information that defines conversions and output precision. In most cases the default file can be used but additional units may need to be added for a user's needs (in this case please notify the developers so the units can be added to the default file distributed with installations).

### 6.2 Configuration File

The *system\StateModGUI.cfg* file specifies StateMod GUI configuration information, accessible from the ***Tools…Options*** menu in the StateMod GUI. The defaults in the file currently must be manually edited if they are to remain between StateMod GUI sessions. The following illustrates how the path to the StateMod and SmDelta executables can be set, and are specific to each StateMod GUI version that is installed:

```
#
# StateModGUI configuration properties
#
# ${Home} will be replaced by the software with the -home command line parameter
# from installation.  This is the software installation home, which by default is:
# C:\CDSS\StateModGUI-Version
#
# ${WorkingDir} will be replaced by the response file folder
#

# StateMod executable to run from the GUI
# Although a generic location like "StateMod.exe" could be used, the model version
# may impact model results and therefore the version can be specifically defined

# Use the following if a StateMod executable is not expected to be found with the dataset
# This is preferred to the work-around described below
#StateModExecutablePath = "${Home}\bin\StateMod-13.00.00.exe"
# Use the following if a StateMod executable is expected to be found with the dataset
# The following default is used because the StateModExecutable = ..\Bin\StateMod-13.00.00
# response file property causes an error with StateMod 13.00.00.
# This work-around is not desirable because it lessens the modeler's control of the dataset
# and assumes that all datasets are similarly configured
StateModExecutablePath = "${WorkingDir}\..\Bin\StateMod-13.00.00.exe"

# Delta plot executable to run from the GUI...

SmDeltaExecutablePath = "${Home}\bin\SmDelta.exe"
```

```
# Startup folder for the GUI when doing:  File...Open...Response File

DataHomeDefault = "C:\CDSS\data"
```

### 6.3 Map Configuration

Refer to the **Configuring Spatial Data for the StateMod GUI Appendix** for more information about configuring maps for use with the StateMod GUI.

This page is intentionally blank.

# Appendix: Configuring Spatial Data for the StateMod GUI

07.04.00, 2013-04-18

## 1. Overview

This appendix describes how to configure the spatial data files for the StateMod GUI. Files that are configured according to current standards will be consistent with the **GeoView Mapping Tools Appendix**. The following section describes the current conventions for spatial data. A later section discusses how to update old files to new conventions.

## 2. Current StateMod Spatial Data Conventions

The current spatial data conventions for the StateMod GUI allow for flexible use of spatial data, integration with StateMod data files, and continued enhancement of the software.

The spatial data are controlled by a GeoView Project File (**.gvp*), which is specified in the free-format StateMod response file using the `GeographicInformation` property. The **GeoView Mapping Tools Appendix** provides a detailed explanation of the *gvp* file, which can be edited with a text editor. Currently the StateMod GUI does not provide capabilities to configure the file and a text editor is the only approach. The *gvp* file is the same format used in other CDSS tools, including StateView and TSTool. In fact, any of these tools can open *gvp* files created for the different applications. However, in practice, the files cannot often be shared because they have different purposes (e.g., StateMod GIS files contain features that are not present in the general data used with StateView). Ideally the spatial data that are displayed in the various CDSS software programs should use common symbols, but this may not be practical.

The following *.gvp* file from the Rio Grande Decision Support System (RGDSS) StateMod data set illustrates the format of the file. No firm standard has been put in place for the naming of gvp files, especially because there may be several StateMod response files as part of a StateMod data set. A rough guideline is to use the data set base name followed by *_StateMod.gvp* as the GeoView Project file name (e.g., *rgTW_StateMod.gvp*). Another guideline is to place all spatial data in a *gis* directory under the StateMod data set directory and place the *gvp* file in the *gis* directory. Sharing spatial data for modeling between StateMod, StateCU, and other software minimizes redundant.

```
# GeoView project file for Rio Grande basin.

# Main GeoView properties.

[GeoView]

# Main home for data
# If a directory is not specified, the directory will be determined when the
# GeoView project file is selected.
#GeoDataHome = "C:\cdss\statemod\data\rgtwday\gis"
# ArcView/ArcExplorer Default...
#SelectColor = Yellow
# Arc 8...
#SelectColor = Cyan
# All-purpose (magenta/pink)
SelectColor = "255,120,255"
MaximumExtent = "266400,4090475 503060,4260700"
```

```
# Now list the layer views.  A layer view consists of specifying a data layer
# (e.g., shapefile) and view information (e.g., symbol).  This is equivalent to
# the ESRI "theme" concept.  The layers specified first are drawn on the bottom.
# Start with number 1 and increase the layer number sequentially as layers are
# added on top.
#
# Important properties are:
#
# GeoLayer - file name for layer data
# Name - Name to show in legend
# SkipLayerView - if true, don't display (but keep in file to maintain layer
#                 numbering)
# AppLayerType - this is "hardcoded" in software to allow recognition in
#                 specific displays.  The value is "softcoded" here to
#                 allow flexibility.
# AppJoinField - attribute field(s) in the GIS data that are used to link to
#                 the StateMod data.  The number of fields must match what
#                 is needed for the AppLayerType (e.g., diversions
#                 require a wd,id).

[GeoLayerView 1]
GeoLayer = div3_districts.shp
Name = "Water Districts"
# tan
Color = "255,240,190"
OutlineColor = black
ReferenceLayer = true
AppLayerType = "BaseLayer"

# The following layer is currently skipped.  It can be turned on.  Additional
# classification/symbolization is being implemented.
[GeoLayerView 2]
SkipLayerView = true
GeoLayer = div3_irrig.shp
#Name = "Irrigated Parcels"
Name = "Crops"
AppLayerType = "BaseLayer"
# Draw as single color...
#SymbolClassification = "Single"
# Green
#Color = "0,255,0"
# black
OutlineColor = "0,0,0"
# ---- OR draw as unique values -----------
#SymbolClassification = "UniqueValues"
#SymbolClassField = "wd"
#Color = "red;blue;green;yellow;pink;cyan;orange;magenta"
# ---- OR draw as class breaks -----------
SymbolClassification = "ClassBreaks"
#ColorTable  = "Custom;3;red;green;blue"
ColorTable = "YellowToRed,10"
#ColorTable = "BlueToCyan,10"
#SymbolClassField = "wd"
#SymbolClassBreaks = "20,22,24,26,30"
# Try acres for testing
SymbolClassField = "ACREAGE"
SymbolClassBreaks = "1,10,20,40,100,150,200,300,400,500"

[GeoLayerView 3]
GeoLayer = div3_lakes.shp
#GeoLayer = div3_lakes.shp
Name = "Lakes"
```

```
# - blue
Color = "165,250,254"
OutlineColor = "0,130,254"
AppLayerType = "BaseLayer"

[GeoLayerView 4]
Name = "Rivers"
GeoLayer = div3_rivers.shp
#GeoLayer = div3_rivers.shp
# RGB - blue
Color = "0,188,253"
AppLayerType = "BaseLayer"

[GeoLayerView 5]
GeoLayer = div3_highways.shp
Name = "Roads and Highways"
Color = "255,0,0"
AppLayerType = "BaseLayer"

[GeoLayerView 6]
GeoLayer = div3_cities.shp
Name = "Cities and Towns"
SymbolStyle = "Square-Filled"
SymbolSize = 6
Color = "red"
LabelField = "Name"
LabelPosition = RightCenter
AppLayerType = "BaseLayer"

[GeoLayerView 7]
#SkipLayerView = true
GeoLayer = rgtw_user_well.shp
Name = "Wells (WDID)"
# brown
Color = "164,134,77"
SymbolStyle = "Square-Filled"
SymbolSize = 6
AppLayerType = "Well"
AppJoinField = "ID_LABEL"

[GeoLayerView 8]
#SkipLayerView = true
GeoLayer = rgtw_user_dw.shp
Name = "Div & Well (Other)"
# brown
Color = "164,134,77"
SymbolStyle = "Square-Filled"
SymbolSize = 6
AppLayerType = "DiversionWell"
AppJoinField = "ID_LABEL"

[GeoLayerView 9]
#SkipLayerView = true
GeoLayer = rgtw_dw.shp
Name = "Div & Well (WDID)"
# brown
Color = "164,134,77"
SymbolStyle = "Square-Filled"
SymbolSize = 6
AppLayerType = "DiversionWell"
AppJoinField = "ID_LABEL_6"

[GeoLayerView 10]
```

```
#SkipLayerView = true
GeoLayer = rgtw_user_other.shp
Name = "Diversions (Other)"
# Green
Color = "0,255,0"
SymbolStyle = "Square-Filled"
SymbolSize = 4
AppLayerType = "Diversion"
AppJoinField = "ID_LABEL"

[GeoLayerView 11]
#SkipLayerView = true
GeoLayer = rgtw_user_div.shp
Name = "Aggregate Diversions"
# Green
Color = "0,255,0"
SymbolStyle = "Square-Filled"
SymbolSize = 4
AppLayerType = "Diversion"
AppJoinField = "ID_LABEL"

[GeoLayerView 12]
#SkipLayerView = true
GeoLayer = div3_diversions_2001-10-24.shp
Name = "Diversions (WDID)"
# green
Color = "0,255,0"
SymbolStyle = "Square-Filled"
SymbolSize = 4
AppLayerType = "Diversion"
AppJoinField = "ID_LABEL_6"

[GeoLayerView 13]
#SkipLayerView = true
GeoLayer = outco_flowstations_2000-06-02.shp
Name = "Stream Gages (Outside-CO)"
# orange
Color = "254,167,0"
SymbolStyle = "Circle-Filled"
SymbolSize = 6
AppLayerType = "Streamflow"
AppJoinField = "STATION_ID"
#LabelField = "STATION_NA, STATION_NA"
#LabelFormat = "%s, %s"

[GeoLayerView 14]
#SkipLayerView = true
GeoLayer = rgtw_user_gage.shp
Name = "Stream Gages (Other)"
# orange
Color = "254,167,0"
SymbolStyle = "Circle-Filled"
SymbolSize = 6
AppLayerType = "Streamflow"
AppJoinField = "ID_LABEL"
#LabelField = "STATION_NA, STATION_NA"
#LabelFormat = "%s, %s"

[GeoLayerView 15]
#SkipLayerView = true
GeoLayer = div3_flowstations_2001-10-24.shp
Name = "Stream Gages"
# orange
```

```
Color = "254,167,0"
SymbolStyle = "Circle-Filled"
SymbolSize = 6
AppLayerType = "Streamflow"
AppJoinField = "STATION_ID"
#LabelField = "STATION_NA, STATION_NA"
#LabelFormat = "%s, %s"

[GeoLayerView 16]
#SkipLayerView = true
GeoLayer = rgtw_user_mf.shp
Name = "Instream Flow"
# RGB 255 0 0 - red
Color = "0xFF0000"
SymbolStyle = "InstreamFlow"
SymbolSize = 6
AppLayerType = "InstreamFlow"
AppJoinField = "ID_LABEL"
#LabelField = "ID_LABEL"
#LabelFormat = "%s"

[GeoLayerView 17]
#SkipLayerView = true
GeoLayer = rgtw_user_res.shp
Name = "Reservoirs (Other)"
# black
Color = "black"
SymbolStyle = "Triangle-Up-Filled"
SymbolSize = 6
AppLayerType = "Reservoir"
AppJoinField = "ID_LABEL"

[GeoLayerView 18]
#SkipLayerView = true
GeoLayer = div3_reservoirs_2001-10-24.shp
Name = "Reservoirs (WDID)"
# black
Color = "black"
SymbolStyle = "Triangle-Up-Filled"
SymbolSize = 6
AppLayerType = "Reservoir"
AppJoinField = "ID_LABEL_6"
```

The example *gvp* file shown above illustrates several important points (see the **GeoView Mapping Tools Appendix** for detailed information on the properties in the file):

- Comments are allowed anywhere in the file using lines that start with `#`.
- The main sections of the file are indicated by using the [Section] notation. In particular, the `[GeoView]` and `[GeoLayerView N]` section headings are important. The first specifies global properties for the map and the second indicates properties for a layer on the map.
- The numbers in the `[GeoLayerView N]` sections should be consecutive, starting with `1`. If a break is detected, layers numbered after the break will not be displayed on the map.
- Currently, one layer (e.g., a shapefile) can be displayed in one "layer view". If a different view of the layer is needed with different symbols, the file is reread.
- Each layer view can have different properties, as per the **GeoView Mapping Tools Appendix**.
- Rather than renumbering layers each time a change is made, the `SkipLayerView=true` property can be used to turn a layer view on/off.

- The `AppLayerType` property is used to tell an application (like the StateMod GUI) the basic data type for the layer and the `AppJoinField` is used to tell the application how to join its data with the spatial data. This information is used by the StateMod GUI to relate a StateMod's data set features with spatial data features. The above example illustrates values that are consistent with shapefiles generated from HydroBase. Once a map is displayed in the StateMod GUI, a layer can be highlighted in the layer list, and the right click ***Show Attribute Table*** choice can be used to display the attributes. This information may need to be reviewed to determine appropriate values for the `AppJoinField`.

The StateMod GUI recognizes the following values for `AppLayerType`:

| AppLayerType Value | How Used in the StateMod GUI |
|---|---|
| BaseLayer | Indicate base layers that can be ignored by specific displays. |
| Baseflow | Used when displaying stream estimate stations. |
| Diversion | Used when displaying diversion station information. |
| DiversionWell | Used when displaying diversion & well (D&W) station information. |
| InstreamFlow | Used when displaying instream flow station information. |
| Precipitation | Used when displaying precipitation station information. |
| Reservoir | Used when displaying reservoir station information. |
| Streamflow | Used when displaying stream gage station information. |
| Well | Used when displaying well station information. |

Spatial data layers can be specified as a comma-separated-value file. The format of the file is illustrated below. If X and Y are specified, the projection is indicated in the **.gvp* file. This simple format can be generated by database queries or exported from Excel.

```
# File generated by...
# program:      StateModGUI 07.03.01 (2009-06-19)
# user:         sam
# date:         Tue Dec 21 17:51:02 MST 2010
# host:         AMAZON
# directory:    C:\CDSS\data\cm2009\statemod
# command line: StateModGUI -home
#            C:\Develop\StateModGUI_SourceBuild/StateModGUI/test/operational/CDSS
#
# pln has 4 out of 6 locations with missing spatial data.
#
# A couple of locations are filled in using lat/long for the coordinates
#
"ID","Name","Lon","Lat","X","Y","Note"
954683OOPPLN,Con-Hoosier_OOP_Plan,,,,,"GUI Detected no coordinates"
363570OOPPLN,Upper_Blue_OOP_Plan,-106.100863,39.385722,-106.100863,39.385722,
364684OOPPLN,Roberts_Tun_OOP_Plan,,,,,"GUI Detected no coordinates"
364512OOPPLN,Dillon_OOP_Plan,-106.067855,39.620327,-106.067855,39.620327,
HUPLimitPLN,Replacement_Limit_Pln,,,,,"GUI Detected no coordinates"
CSULimitPLN,Replacement_Limit_Pln,,,,,"GUI Detected no coordinates"
```

## 3. Configuring Spatial Data for StateMod

Spatial data files in CDSS are typically distributed on a Water Division extent, and as statewide layers. Non-point features like rivers are typically clipped to a water division's boundaries. Data for trans-basin and out of state structures are also available. Spatial data layers are distributed with HydroBase releases on CD/DVD and are available on the CDSS web site.

For StateMod modeling, it is best to use the available data whenever possible. However, these spatial layers often do not satisfy all the needs of a modeler for a number of reasons:

- Locations in HydroBase may not be correct or may be missing.
- StateMod data sets may include additional features that are not in HydroBase (e.g., aggregate diversions).
- Features in HydroBase may not be available in spatial data (e.g., instream flow reaches).

For these reasons, spatial data used with a StateMod data set often consist of some "official" data corresponding to HydroBase and some model-specific data that may be generated in various ways (e.g., by digitizing locations within ArcView and then saving as a shapefile). If an older data set is available that has spatial data, the associated shapefiles can serve as a starting point for a data set, but the locations may not be consistent with the current HydroBase data. Otherwise, the official data from the State can serve as a starting point. In either case, it is likely that some locations will be missing. The following procedure recommends one possible solution to creating a current set of spatial data for a StateMod data set:

1. Copy a GeoView Project file (*.gvp*) from an existing StateMod data set. Such files should be available from State of Colorado staff and are being phased in on baseline data sets. If one is not available, copy from this documentation or create using a text editor. Point data configuration properties in particular often have similar properties. Place the resulting file in a *gis* directory under the StateMod data directory or a directory to be shared with other tools. It is recommended that the file be named *XXXXX_StateMod.gvp*, where *XXXXX* is the data set name.
2. Copy shapefiles from available sources, such as old StateMod data sets, State of Colorado spatial data sources, or digitized layers, and place in the *gis* directory mentioned in the previous step. The GeoView tools have limited ability to project on the fly so using data with a consistent coordinate system is recommended.
3. Edit the *gvp* and order the use of the shapefiles as appropriate. In general the map layers should be ordered back to front as: background color/image, then polygons, then lines, then points. Use examples from GeoView Project documentation or guidelines from the State to define symbols. Use placeholder layers and the `SkipLayerView` property where needed until all data can be found.
4. Make sure that point data have appropriate `AppLayerType` and `AppJoinField` properties. The `AppJoinField` should be a field in the shapefile that corresponds exactly to the identifiers used for the StateMod data type. For example for diversions that use WDID in StateMod, the identifiers may be padded to six or seven characters. Spatial data files generated from HydroBase spatial data typically include character fields similar to `id_label_6` and `id_label_7` to allow matching with multiple uses. This information must be specified correctly in order for the StateMod GUI to properly join spatial data with the StateMod data set.
5. Reference the *gvp* file in the StateMod data set response file for the `GeographicInformation` property. Use a relative path like *gis\XXXXX_StateMod.gvp* to increase options to move the data set from one computer to another.

6. Run the StateMod GUI and select the response file that references the *gvp* file. The spatial data will be loaded and a map will display. Note that when the StateMod GUI loads spatial data, it processes the data so that identifiers that are not matched in spatial data will not display. Consequently, although extra data in shapefiles may require resources, the shapes are not actually displayed. Note also that the first match that is found for an identifier is used. If a structure is in a diversions layer and a transbasin layer, the first layer to be processed will be used. It is therefore important that if a location is re-defined in a user-generated layer that the layer is specified first in the gvp file. The GUI will also create files in the same folder as the response file with names like *x-gui-MissingSpatial-dds.csv*. These files can be edited and configured as a layer in the **.gvp* file to specify additional location information.
7. Use the StateMod GUI ***View…Data Set Summary*** menu to list features that do not have location data.
8. In many cases, missing locations can be added by adding another layer (e.g., add another Division's layer or the trans-basin layer for the data type). It may be useful to review spatial data using GIS software. Repeat above steps as needed to completely define spatial data.
9. If locations are still missing (e.g., for aggregate diversions), then it may be necessary to digitize locations. For this case, it is recommended that the official shapefile from HydroBase not be edited but instead an additional layer be added. This more easily allows for updates of the official data. No naming convention is currently suggested for "user defined" layers and official CDSS tools do not exist to define missing location.

## 4. Old Spatial Data Conventions

In StateMod GUI versions prior to version 05.10.xx, the map interface was a holdover from previous versions of the software. Although this interface worked relatively well within the StateMod GUI, it had some limitations. In more recent versions, the map interface is consistent with other software in Colorado's Decision Support Systems, and is more consistent with general GIS tools.
A GIS Control File referenced in the StateMod response file controlled the old map interface. An example of the file is as follows:

```
#
# rg.gis - graphics files for the Rio Grande data set
# ------------------------------------------------------------------
basin:         gis\counties.shp  displayReferenceMap=true
rivers:        C:\cdss\statemod\rg\gis\riversd3.shp
streamflow:    gis\div3_flowstations_1999-02-02.shp id=station_id
diversions:    gis\div3_diversions_1999-02-02.shp id=ID_LABEL_6
reservoirs:    gis\div3_reservoirs_1999-02-02.shp id=ID_LABEL_6
precipitation: gis\div3_climatestations_1999-02-02.shp id=station_id
```

Comments in the file are lines that start with a # character. Data lines consist of a keyword and a shapefile path. File names can contain explicit or relative paths. Multiple keywords of the same type can be specified to plot more than one file. The keywords are used to set the data type, which the GUI then uses to internally set colors, symbols, etc., for plotting. Recognized keywords are:

| | |
|---|---|
| **baseflow** | Baseflow nodes (if not in other site files) |
| **basin** | Overall basin boundary |
| **diversions** | Location of diversion headgates |
| **instream** | Location of instream (minimum) flow nodes |
| **precipitation** | Location of precipitation gages |
| **reservoirs** | Location of reservoirs |
| **rivers** | River reaches |
| **streamflow** | Location of stream gages |

| | |
|---|---|
| **usersites** | Location of user-specified sites |
| **wells** | Location of wells |
| **dw** | Location of diversion/well nodes |

The layers are displayed in the order listed, with the last layer displayed on top. Any layers to be displayed in the reference map (typically only the basin layers) must have the keywords `displayReferenceMap=true` after the filename. Also, for the GIS to know which column from the shapefile to use as the identifier, specify the column title using the id keyword (e.g., `id=ID_LABEL_6`). This information allows the shapefile data to be joined to StateMod data.

Although the above convention can be easily implemented, it is not very flexible, in particular with respect to assigning symbol colors and other properties. Therefore, old GIS control files should be migrated to the new conventions.

This page is intentionally blank.

# Appendix: StateMod GUI Release Notes

Version, 07.04.00, 2013-04-17

This appendix provides information about changes that have occurred in StateMod GUI versions.

## StateMod GUI Version History

The following table summarizes the StateMod GUI release history. See the following sections for more detailed information about each version. Only recent versions are documented in detail. Comments for minor versions may be listed under a version that is publicly released. Recent release note items are categorized as follows:

Bug Fix – A bug has been fixed. Users should evaluate whether their work is impacted.

Known Limitation – A known limitation has been documented and may impact the user. The limitation will be addressed in a future release.

Change – An existing feature has been changed.

Remove – A feature has been removed.

New Feature – A new feature has been added, with functionality that was not previously available.

**StateMod GUI Version History Summary (most current at top)**

| StateDMI Version | Version Information | Release Date |
|---|---|---|
| 07.03.00 – 07.04.00 | Enhance training documentation. Rebuild software consistent with components from other CDSS tools. | 2013-04-17 |
| 07.02.00 | Improve operating rule editing and allow display on network. Update installer to support Windows 7. | 2010-12-13 |
| 07.01.00 | Updates to be compatible with data sets on CDSS web site. Limitations are noted below. | 2008-03-19 |
| 07.00.00 | New software installer. | 2007-03-02 |
| 06.05.00 | Continue implementing plan stations and call time series. | 2006-08-22 |
| 06.04.00 | Initial implementation of plan stations. | 2006-08-16 |
| 6.03.02 | Full documentation update; add reservoir curve graphs, other maintenance. | 2006-02-28 |
| 06.03.01 | Fix bug in diversions window that would corrupt data if the diversion list was sorted by the user and return flows or rights are viewed, fix graph button being hidden on the delay table window, minimize console output. | 2006-01-20 |
| 06.03.00 | Major release including enhanced tabular displays consistent with StateDMI. | 2004-10-14 |
| 06.02.00 | Internal release. | 2004-08-25 |
| 06.01.08 | Graphing tool uses the StateMod binary output file. | 2003-12-01 |
| 06.01.07 | Include demand/supply summary tool. | 2003-11-07 |
| 06.00.01 Beta – 06.01.06 Beta | Releases to the State for testing and review. | 2003-10-02 to 2003-11-06 |
| 06.00.00 | Initial Java 1.4.2 release using new graphical components, add –v and –h command line parameters. | 2003-06-03 |

| StateDMI Version | Version Information | Release Date |
|---|---|---|
| 05.11.00 | Final Java 1.1.8 release. | 2003-01-31 |
| 05.10.04 | Maintenance release. | 2002-10-29 |
| 05.10.03 | Increase performance of calls to StateMod. | 2002-10-20 |
| 05.10.02 | Change to new, more general menu structure, use GeoView if a *.gvp* file is detected, add baseflow to graphing tool, | 2002-08-22 |
| 05.10.01 | Fix bug detecting delplt exit status. | 2002-07-15 |
| 05.10.00 | Add user name to log file. Fix problems running on a server. | 2002-07-03 |
| 05.09.00 | Add additional support for wells. Update to let delplt work with new parameter names. | 2002-06-02 |
| 05.08.00 | Minor update. | 2001-11-21 |
| 05.07.00 | Update to use map display code that is compatible with StateView/CWRAT 02.06.00. Fix bug where some absolute paths in response file were not being handled correctly. | 2001-10-19 |
| 05.06.01 | Minor update. Tested with StateMod 9.96. | 2001-09-23 |
| 05.06.00 | Change to new graphing tool. Update GUI to be support current StateMod 9.92 model and be backward compatible with older data sets. Running on NT now shows incremental model screen output. | 2001-08-23 |
| 05.05.00 | General maintenance. Included operational rights 15 and 16 and reworked delay file display to accommodate 240 return values. | 2000-04-24 |
| 05.04.00 | Improved performance of running StateMod through the GUI. Included 2 new data types in BigPicture plot | 2000-02-09 |
| 05.03.00 | Included the well data type and daily time series. | 1999-12-22 |
| 05.02.00 | Added GIS interface, including legend, mouse tracking and big picture plot. | 1999-09-15 |
| 05.01.00 | Internal release. | |
| 05.00.00 | Assuming you are running StateMod 8.14, SMGUI can now be run from the Start menu. Added functionality to allow creating BigPicture plots using ArcView. JRE is now a separate installation. Export from graph window available. | 1998-05-19 |
| 04.03.00 | Work-around implemented for running StateMod on NT platform. Added ability to view historical diversion time series. Allow second scenario to be read. Addressed minor bugs/labeling. | 1998-11-11 |
| 04.02.00 | Second major PC release. Many enhancements and minor bugs have been addressed. | 1998-10-03 |
| 04.00.01b | First major PC release. | 1998-06-08 |
| 04.xx.xx | Start development in Java on PCs. | 1997-06-01 |
| 1.x – 3.x | UNIX versions of SMGUI. | 1995 – 1997 |
| - | Start of development. | 1995 |

### Known Limitations

- Known Limitation Although significant effort has been invested in the operating rules editor, additional discussion needs to occur to determine how best to support the many operating rules, and ensure that formatting is tested. Additional documentation explaining the details of operating rules is needed.
- Known Limitation Resources for StateMod GUI development have lagged behind development of the StateMod model. Therefore, the GUI may not be fully consistent with model features. Additional resources need to be applied to ensure this consistency. For example, preliminary support has been added for plan stations; however, StateMod development occurred after initial GUI features were completed.
- Known Limitation The StateMod model operates on text input files, which typically are created with the StateDMI and TSTool software. To provide editing capabilities, the GUI reads the input files, allows interactive edits, and rewrites the text files. However, the updated files do not indicate markers for changes, and the original StateDMI and TSTool command files are not updated. Additional resources need to be applied to improve integration.
- Known Limitation The StateMod training data set focuses on response file cm2009.rsp, including configuration for the map interface; however, the other response files were not updated. Additional resources need to be applied to StateMod data sets to better integrate the data with StateMod GUI features.
- Known Limitation Exiting the StateMod GUI may show dialog indicating that files have changed, when the user has not made any changes. This is due to the GUI cleaning up data, dealing with migration of data from one format or another, or a limitation in editing features for a certain file. If the data set is only being viewed, cancel the request to save data.

### Changes in Versions 7.03.00 – 7.04.00

- Change [07.04.00] Rebuild software using updated components consistent with other CDSS tools, essentially a maintenance update.
- New Feature [07.04.00] **Chapter 8 – Using the Map** in the documentation has been completed.
- New Feature [07.03.00] Operating rules can now be displayed on the map display, to help troubleshoot.
- New Feature [07.03.00] Map layers can now be configured using simple text comma-separated-value files. The GUI creates template files indicating stations that do not have coordinates, to allow the files to be completed and supply coordinates.
- New Feature [07.03.00] Training materials are now available with the software installation, including a Colorado River dataset used for examples.
- New Feature [07.03.00] An attempt has been made to represent all StateMod model files in the GUI, even if full editing/viewing features are limited. This will help ensure that future development can be targeted and all data are represented to users in some form. Basic data viewing features for consumptive use locations have been added.
- New Feature [07.03.00] Basic data viewing features for consumptive use locations have been added.
- New Feature [07.03.00] All operating rules can be viewed in the editor window. Additional testing is needed to ensure that interactive editing is correct.

### Changes in Versions 7.02.00

- Change [07.02.00] Old fixed-format response files are no longer supported. If an old data set needs to be opened, update the response file using the *SmNewRsp* utility program.
- New Feature [07.02.00] Operating rules can now be displayed on the network, to help troubleshoot.

### Changes in Version 7.01.00

- The software installer has been updated to install in a versioned location (folder and menus) using a convention like *\CDSS\StateModGUI-07.01.00*. This allows multiple versions of the software to be installed, which is useful during development and in order to archive software versions used with specific modeling efforts. The conventions of the installer are consistent with other common software.
- Documentation is now distributed as a navigable PDF.
- This version of the GUI is distributed with StateMod version 12.200 (2008/03/13).
- The *system\StateModGUI.cfg* file has been added to store StateMod GUI configuration information, including the StateMod and SmDelta programs to be used at run-time.
- Fix bug in the network editor that occurred when the number of annotation labels was greater than the number of nodes. This had caused annotation positions to not be editable.
- Fix bug where diversion properties could not be displayed from the data tree on the main screen.
- Allow unknown operating rule types to be read as simple text by assuming that lines 2+ with a space in character 1 are a continuation of the operating rule. This allows the entire operating rule file to be read. See known limitations below.
- A number of changes have been made in the dependencies of software components. The StateMod GUI is no longer dependent on HydroBase software components. The Batik package is now being distributed to allow saving graphs as Scalable Vector Graphics (SVG) files. It is envisioned that the network and map displays will also be updated to allow saving to SVG.
- The prompt as to whether to read all the data has been removed. Computers are now powerful enough to read all the data.
- The performance of running the StateMod program from the GUI has been improved. Less output from the model is displayed in the GUI feedback dialog and there may be a delay in displaying output as StateMod runs. Full output can still be viewed in StateMod log file. The performance of running the StateMod model from the GUI and command line should be similar.
- **Known limitation**: Operating rule 23 is the last to be fully understood by the software. Enhancements are pending that will correct this. Newer operating rules cannot be fully viewed or edited by the StateMod GUI.
- **Known limitation**: The SmDelta functionality is not fully supported. There appear to have been changes in the program operation that need to be made consistent in the GUI. The GUI has been updated to use "SmDelta" rather than "delplt".
- **Known limitation**: Annual precipitation and evaporation time series are not recognized by the GUI (monthly data are).
- **Known limitation:** The Stream Estimate Stations list is the same as the Stream Gage list. Additional enhancement needs to occur to separate the **.ris* file into two lists for display purposes. Stream gages are associated with historical flow time series and stream estimate stations are associated with proration data in the **.rib* file.
- **Known limitation:** Plan stations are not fully supported. Although the plan station file can be read and displayed, enhancements are pending to finalize features consistent with current StateMod specifications.

- **Known limitation:** Some data sets are distributed that have **ipy* files with older header formats. StateCU does not recognized water year (all analysis is calendar year). An appropriate header is as follows:
  1950 2005 CYR
  However, the data set may have something like:
  10/1908 9/2004 ACRE WYR
  The StateMod GUI will read the file, but will complain that the file may not be properly formatted. StateMod will read the file.
- **Known limitation:** The StateMod.exe program was previously typically installed in *\cdss\bin*. However, versioned installs of the StateMod GUI now distribute the StateMod program in the GUI *StateModGUI-Version\bin* folder in order to ensure software compatibility. Modelers are also often including the StateMod executable in the *StateMod* folder in data sets, to ensure compatibility with the data set. It is envisioned that the StateMod GUI options will be enhanced to allow users to configure which StateMod executable should be used. In the meantime, use the *system\StateModGUI.cfg* file to configure the location. Running StateMod from the command line will be controlled by the current directory, the `PATH` environment variable, and whether the user has typed the path to the model or has a batch file that does so.

#### Changes in Version 7.00.00

- This is the first version to use the new software installer, which will facilitate distribution and installation of the software.
- Fix bug where binary files were not getting read with newer versions of StateMod.

#### Changes in Version 6.05.00

- Additional work on plan stations and call time series – these features are preliminary pending finalization of StateMod software features.

#### Changes in Version 6.04.00

- Begin adding plan stations – these features are preliminary pending finalization of StateMod software features.

#### Changes in Version 6.03.02

- The documentation has been updated to be consistent with the current version of the software.
- Add graphs for the reservoir curves.
- Fix bug where instream flow rights window for an instream flow station was showing all rights in the data set.
- Fix bug where changing a file name in the response file window was causing the file names to be replaced by the data component name.
- Fix bug where well stations that do not have an associated diversion station were always being shown with the first diversion station.
- Fix bug where for diversion stations in the tree, the “show how used” tool was also showing the diversion station data.
- Fixed bug where when editing delplt input, the delete rows button was not working.
- Fixed button where the animation layer properties could not be displayed from the layer list.
- Make a number of changes to enforce network and other data edits being consistent.

This page is intentionally blank.

# Appendix: DateValue Input Type

2004-08-05, Acrobat Distiller

## Overview

The DateValue time series file format can be used to store one or more time series of consistent time interval. The format has been developed by Riverside Technology, inc. (RTi). The example below shows the format of the file. Important comments about the file format are:

- The file is divided into a header section (top) and data section (bottom). Comments can occur anywhere in the file and are lines that start with #.
- If not specified, many of the header properties will be set to reasonable defaults as data are read by software such as TSTool. However, as much information as possible should be specified to allow complete time series handling. Header information is displayed by applications like TSTool to allow selection of time series before the data section is read.
- Properties are checked in a case-independent fashion.
- The `TSID`, `Start`, `End`, and `Units` properties are important for basic time series handling.
- The interval part of the `TSID` is used to determine how memory should be allocated for data.
- The `Start` and `End` values are used to allocate memory for regular interval time series. Dates associated with data values are used to allocate memory for irregular interval time series.
- For regular interval time series, if data lines between the start and end dates are omitted, the unspecified values are set to the missing data value for the time series (default is `-999`).

```
# DateValueTS 1.1 file
#
# This is a sample of a typical DateValue minute time series. This format
# was developed by Riverside Technology, inc. to store time series data. An
# example file is as follows and conforms to the following guidelines:
#
# * Comments are lines that start with #.
# * Applications often add a comments section at the top indicating how the
#   file was created
# * Any line that starts with a number is assumed to be a data line.
# * Date hours should be in the range 0 to 23 (an hour of 24 will be
#   converted to hour 0 of the next day).
# * If a time is necessary, the date/time may be separated by a space, T, :, or
#   @. If a space is used, use a date and time column headings,
#   if headings are used.
# * The same general format is used for year, month, day, hour, and minute
#   data, except the format of the date is adjusted accordingly.
# * If multiple time series are written, header variables are delimited with
#   space or tab characters. Data are delimited by tab or space (or use the
#   Delimiter property to set the delimiters used for data lines)
# * Internally, the time series identifier may initially be set using the file
#   name. For example, a file name of XXX.USGS.Streamflow.MONTH will result
#   in the location being set to "XXX", the data source to "USGS", the data
#   type to "Streamflow", and the interval to 1 month. The identifier
#   information is reset if individual properties are specified in the file.
# * This format is free-format and additional information may be added in
#   future (e.g., data quality strings).
# * For portability, data in a DateValue file should have compatible intervals.
# * Header variables and column headers can be enclosed in double quotes if
#   the data contain spaces.
# * Missing data can either be coded as the missing data value or no value
```

```
# * Missing records will result in missing data being used when read.
#
# The following header variables are recognized. This information can be
# used by software.
Version = 1.1                   # Optional. File format version
                                # (to handle format changes)
Delimiter = " "                 # Optional. Delimiter for data lines
                                # (default is space and tab)
NumTS = 2                       # Optional. Number of time series in file
                                # (default is 1)
TSID = “XXX.USGS.Streamflow.15MINUTE” “YYY.USGS.Streamflow.15Minute”
                                # Required.
                                # List of time series identifiers in file
                                # Location.Source.DataType.Interval.Scenario
                                # Do not include input type and name in identifier
SequenceNum = 1950 1951  # Optional – used with time series traces.
                                # Indicates the year for the trace
Description = "Flow at XXX" "Flow at Y"
                                # Optional. Description for each time series.
DataFlags = true,1 false        # Optional. Indicates whether data flags (e.g.,
                                # character data quality) are provided. If true,
                                # specify the maximum number of characters that
                                # will be used in any flag (the default is 2 if
                                # not specified). The data value column for each
                                # time series with data flags is followed by a
                                # column for the data flag. Surround the flags by
                                # “” if a flag is not specified or is a space.
DataType = Streamflow Streamflow
                                # Optional. Data types for each time series
                                # (consistent with TSID if specified).
                                # The default is to use the data type in the TSID
                                # Supplied to simplify use by other programs.
Units = CFS CFS                 # Optional. Units for each time series
                                # (default is no units).
MissingVal = -999 -999          # Optional. Missing data value for each
                                # time series (default is -999).
IncludeCount = true             # Optional. If true, column after date/time
                                # is record count (1...) (default is false).
IncludeTotalTime = true         # Optional. If true, column after date
                                # is cumulative time (0...) (default is false).
# Both of above can be true, and both columns will be added after the date
Start = 1996-10-18:00:00        # Required. Start date for time series
End = 1997-06-14:00:00          # Required. End date for time series
                                # Period dates should be of a precision consistent
                                # with the dates used in the data section below.
# Optional. The following line can be read into a spreadsheet or database for
# headers. The lines above this line can be ignored in a spreadsheet import.
# The number of headings should agree with the number of columns.
Date "Time" "Count" "TotalTime" "Description 1" “DataFlag1” "Description 2"
1996-10-18 00:00 1 0 110.74 “m” 14.2
1996-10-18 00:15 2 15 113.24 “” 13.7
...
```

## DateValue Files and Standard Time Series Properties

The standard time series identifier for DateValue files is of the form:

```
Location.DataSource.DataType.Interval.Scenario~DateValue~PathToFile
```

Because DateValue time series files are a persistent storage format for in-memory time series objects that have been developed by RTi, the properties stored in the file closely match the standard time series properties of the objects. In particular, the time series data type, units, and missing data value are consistent with time series header information. The TSID property in a DateValue file is directly applied to time series objects read from the file, allowing explicit identification of the time series in the file, regardless of the name of the file. This allows multiple time series to be saved in a single file. The data source typically agrees with that determined from a data-supplying agency or model that generates the data.

## Limitations

DateValue files have the following limitations:

- The header information in DateValue files may be too technical for some general tools. However, simple delimited files cannot be handled as rigorously by some applications, like TSTool. Spreadsheets can import DateValue files easily by ignoring the header lines.
- Because date/time values are included on every data line, processing DateValue time series files requires more disk space and processing time. However, using the dates on each line also allows gaps in data to be omitted from the file. Inclusion of the date/times for each data point is considered a reasonable trade-off to ensure data quality and readability. Many other time series file formats also include the date/time on each line.

This page intentionally blank.

# Appendix: StateMod Input Type

2004-07-27, Acrobat Distiller

## Overview

The StateMod time series input type corresponds to the file format used by the State of Colorado's StateMod model, including standard daily, monthly, average monthly (referred to as annual in the StateMod documentation) file formats. See also the StateModB input type, which corresponds to StateMod binary output files and the StateCU input type, which corresponds to the State of Colorado's StateCU consumptive use model.

The following example illustrates the format of the three main file formats. See the **StateMod Documentation** for a complete description of StateMod input files. Important comments about the file format are:

- The file is divided into a header section (top) and data section (bottom). Comments can occur only at the top and are lines that begin with #.
- One or more time series can be stored in a file.
- Consistency in the order and number of the stations is required for each year of data, within the file.
- Other than comments, the file is fixed-format, compatible with FORTRAN applications. See the **StateMod Documentation** for field specifications.
- The format is optimized to allow a full year of data to be read for the entire data set. Reading a time series for a single location for the full period requires reading through the entire file.
- In addition to the required values, a total/average value is accepted as the far-right value on each data line. This value may be ignored by applications (it can be computed from the data values on the line if necessary).
- The precision of data values may be controlled by software, resulting in more or fewer fractional digits. This may lead to round-off differences when comparing raw data values with the total/average in the optional end column.

```
# StateMod time series files can have 3 main forms (monthly, average monthly, daily) as
# described below. The order of time series is important for
# some files (e.g., order of diversion time series should match order of
# diversion stations in .dds file); however, StateMod is being updated over
# time to remove this requirement). Different StateMod input files have
# slight variations on the general format (e.g., the reservoir target file
# has two time series for each reservoir for minimum and maximum targets).
# Missing data are typically indicated by -999.
# The generic extension for StateMod time series files is .stm, although specific
# extensions are used in a StateMod data set.
#
# 1) This is an example of a StateMod monthly time series for water year data:
#
# Comments are lines at the top of the file starting with the # character.
# The header may contain software-generated comments about the time series.
# The remainder of the file is fixed format, with the first non-comment
# line being a header with the following elements (i5,1x,i4,5x,i5,1x,i4,a5,a5):
#
# Beginning month (1=Jan)
# Beginning year (4-digit)
# Ending month
# Ending year
# Data units (AF/M, ACFT, CFS or ""), where rates are for diversions and
#   flow, and volume is for reservoir contents. Units are not used for
#   dimensionless data (like weight or percent).
# Year type (CYR=calendar, WYR=water, IYR=irrigation)
```

```
#
# Data lines then follow with:
# Year Station 12-monthly-values year-total/average (i4, 1x, a12, 12f8, f10)
# The year value is optional and is generally not read as input but is
# computed for output. The year in data lines corresponds to the calendar type.
# An example follows:
10/1926 -      9/1998 ACFT  WYR
1927 08236000        1229.8     892.6     922.3  737.9  555.4  922.3  7049.4  32263.6
31000.1 14541.0  5662.9   8326.7   104104.0
1927 08235250        -999.0   -999.0   -999.0 -999.0 -999.0 -999.0 -999.0  -999.0 -
999.0  -999.0  -999.0  -999.0          0.0
1927 08235700        -999.0   -999.0   -999.0 -999.0 -999.0 -999.0 -999.0  -999.0 -
999.0 -999.0   -999.0  -999.0          0.0
1927 08236500        1047.3     595.1     614.9  614.9  555.4  1900.2 6769.7  31226.2
20338.8          14777.1  9465.3   4476.8   92381.5
...
#
# 2) This is an example of a StateMod average monthly time series for water year data:
#
# The average monthly time series is a pattern of twelve monthly values
# that are applied for each year in the period.
# The format is exactly the same as a monthly time series; however, the
# years in the header should be set to zero and year and month are ignored in data rows
# and can therefore be blank.
#
# An example follows:
10/ 0 -       9/ 0   ACFT WYR
       08236000      1229.8     892.6   922.3     737.9  555.4  922.3 7049.4   32263.6
31000.1 14541.0 5662.9 8326.7 104104.0
       08235250      -999.0    -999.0  -999.0    -999.0 -999.0 -999.0 -999.0    -999.0 -
999.0  -999.0   -999.0   -999.0       0.0
       08235700      -999.0    -999.0  -999.0    -999.0 -999.0 -999.0 -999.0    -999.0 -
999.0 -999.0 -999.0 -999.0 0.0
       08236500      1047.3     595.1   614.9     614.9  555.4 1900.2 6769.7   31226.2
20338.8 14777.1 9465.3 4476.8  92381.5
...
#
# 3) This is an example of a StateMod daily time series for water year data:
#
# The daily time series is similar to the monthly time series except that
# a year and month are included on the data lines and 28, 30, or 31 daily
# data values can occur on each line (end values ignored, depending on month).
# The data format is (i4, i4, 1x, a12, 31f8, f8). The month total/average
# is optional and is generally read as input but is computed for output.
# Regardless of the calendar type in the header, the year and month in data records use
# calendar year (month 1 = January).
#
# An example follows:
10/1926 -     9/1998 ACFT   WYR
1926   10 08236000     -999.00 -999.00 -999.00 -999.00 -999.00 -999.00 -999.00 -999.00 -
999.00 -999.00 -999.00 -999.00 -999.00 -999.00 -999.00 -999.00 -999.00 -999.00 -
999.00 -999.00 -999.00 -999.00 -999.00 -999.00 -999.00 -999.00 -999.00 -999.00
0.00      0.00
...
1927   4 08236000        38.00   42.00   42.00   67.00   90.00   90.00  100.00  118.00
93.00   80.00   93.00    80.00   80.00   80.00   80.00   80.00   68.00   80.00   68.00
68.00   80.00   80.00   106.00  136.00  170.00  229.00  250.00  296.00  322.00  348.00
0.00   114.65
1927   4 08235250      -999.00 -999.00 -999.00 -999.00 -999.00 -999.00 -999.00 -999.00 -
999.00 -999.00 -999.00 -999.00 -999.00 -999.00 -999.00 -999.00 -999.00 -999.00 -
999.00 -999.00 -999.00 -999.00 -999.00 -999.00 -999.00 -999.00 -999.00 -999.00
0.00      0.00
...
```

## StateMod Files and Standard Time Series Properties

The standard time series identifier for StateMod files is of the format:

```
Location...Interval~StateMod~PathToFile
```

StateMod files contain limited header information. Time series properties are set using the following guidelines:

- The location part of the time series identifier is taken from the identifier field in the data records (from the first year of data). A change in the year indicates that all time series have been identified.
- The data source part of the time series identifier is set to `StateMod` or blank. In the past this information was used to indicate the input type (file format) in the time series identifier; however, the new input type notation has a specific field for the input type and therefore data source can be used more appropriately. In the future, it may be possible to pass along the original input source but this information cannot currently be saved in the StateMod file format.
- The data type is often not assigned because it is not defined in the file. Currently no interpretation of the file name extension occurs. Some specific applications (e.g., the StateMod GUI) may set the data type, based on reading a StateMod data set response file (and therefore knowing the specific contents of the file).
- The data interval is assigned as `Day` or `Month` based on the file format (determined automatically).
- The scenario is typically not assigned. Older software may use the scenario to store the file name; however, the new time series identifier notation stores the file name as the input name field (see below).
- The input type part of the time series identifier is set to `StateMod`, indicating the file format. Software will use the interval and/or examine the file contents to verify whether the data are in daily or monthly format.
- The input name part of the time series identifier is set to the file name, either as the full path or a relative path to the working directory.
- The units are assigned to those indicated in the file header.
- The missing data value is assigned to `-999.0`.
- The description is set to the same value as the location. A verbose description can typically be determined by cross-referencing the identifier with another StateMod data file (e.g., diversion stations).
- The period is set based on the header information.

## Limitations

StateMod files have the following limitations:

The format of the does not facilitate extracting one time series from the file. Software has been optimized to perform this within current constraints.
Some time series properties are not explicitly included in StateMod files (e.g., data type). Therefore, general software like TSTool may not be able to provide default information. For example, a graph may show multiple time series with nearly the same legend text because more detailed information cannot be defaulted.
If two time series for the same station are stored in the same file (e.g., reservoir maximum and minimum targets), there is no way to uniquely identify the two time series. The application or user must understand the file type and data organization. Some specific software (e.g., StateMod GUI) may be able to recognize the specific format.

This page is intentionally blank.

# Appendix: StateModB Input Type (StateMod Binary Output Files)

2004-07-27, Acrobat Distiller

## Overview

The StateModB time series input type corresponds to the file format used by the State of Colorado's StateMod model, in particular the binary FORTRAN direct access output files. These files contain important water balance information for every node in the model network. The following table summarizes the contents of the binary files and corresponding text report files (all files can be large for large data sets):

| Node Type | Monthly Binary File | Monthly Report File | Daily Binary File | Daily Report File |
|---|---|---|---|---|
| Diversion | **.b43* | **.xdd* | **.b49* | **.xdy* |
| Instream flow | **.b43* | **.xdd* | **.b49* | **.xdy* |
| Reservoir | **.b44* | **.xre* | **.b50* | **.xry* |
| Stream gage and Stream estimate | **.b43* | **.xdd* | **.b49* | **.xdy* |
| Well | **.b42* | **.xwe* | **.b65* | **.xwy* |

The following documentation describes the format of the B43 binary file. Other files are similar. See the **StateMod Documentation** for a complete description of StateMod output files. Important comments about the file format are:

- The file is generated by StateMod as a direct access binary file with fixed-length records. The record length is 140 bytes.
- The file is divided into a header section (top) and data section (bottom).
- The format is optimized to allow a full year of data to be read for the entire data set. Efficiently reading a time series for a single location for the full period requires reading appropriate lines of the file using direct access. Because the file is binary and consistent for a given data set, file reads can be optimized.
- The data period and the calendar year type are consistent with the StateMod control file.
- All character strings are left justified and are padded with spaces. Therefore, software that reads the file should trim trailing spaces after reading the strings.
- River node identifiers in record 5 are included for all nodes in the network and data records (record 11) follow this order. Subsequent lists for various node types are a subset of the list in record 5 and have data items to reference the position in the river node list. Time series are queried using the identifiers in records 6+. However, the river node position is actually used to retrieve data in the file.

The B43 binary file contains the following records:

| Record | Field | StateMod Variable | Type | Description |
|---|---|---|---|---|
| 1 | 1 | iystr0 | integer | Beginning year of simulation, for year type in StateMod control file. |
| | 2 | iyend0 | integer | Ending year of simulation, for year type in StateMod control file. |
| 2 | 1 | numsta | integer | Number of river nodes. |
| | 2 | numdiv | integer | Number of direct diversions. |
| | 3 | numifr | integer | Number of instream flows. |
| | 4 | numres | integer | Number of reservoirs. |
| | 5 | numown | integer | Number of reservoir owners. |
| | 6 | nrsact | integer | Number of active reservoirs. |
| | 7 | numrun | integer | Number of base flow nodes. |
| | 8 | numdivw | integer | Number of diversion structures with wells. |
| | 9 | numdxw | integer | Number of well only structures. |
| 3 | 1 | xmonam(14) | Each is char(4). | Month names corresponding to the calendar type for the simulation. This information is provided as a convenience for data processing. For example, if the year type is WYR (water year), xmonam(1) is 'OCT'. The 13th value is 'TOT' and the 14th value is 'AVE'. |
| 4 | 1 | mthday(12) | Each is integer. | Number of days per month, corresponding to the calendar type for the simulation. This information is provided as a convenience for data processing and to convert daily data values to monthly. For example, if the year type is WYR (water year), mthday(1) is 31 for October. The number of days in February is typically 28 and is used for all data processing, regardless of whether a year is a leap year. |
| 5 **Repeat record for `numsta`** | 1 | j | integer | Counter for record type 5. |
| | 2 | cstaid(j) | char(12) | River node identifiers. |
| | 3 | stanam(j) | real(6) | River node names. |
| 6 **Repeat record for `numdiv`** | 1 | j | integer | Counter for record type 6. |
| | 2 | cdivid(j) | char(12) | Diversion identifier. |
| | 3 | divnam(j) | real(6) | Diversion name. |
| | 4 | idvsta(j) | integer | River node position (1+) to allow cross-reference with river nodes. |
| 7 **Repeat record for `numifr`** | 1 | j | integer | Counter for record type 7. |
| | 2 | cifrid(j) | char(12) | Instream flow identifier. |
| | 3 | xfrnam(j) | real(6) | Instream flow name. |
| | 4 | ifrsta(j) | integer | River node position (1+) to allow cross-reference with river nodes. |
| 8 **Repeat record for `numres`** | 1 | j | integer | Counter for record type 8. |
| | 2 | cresid(j) | char(12) | Reservoir identifier. |
| | 3 | resnam(j) | real(6) | Reservoir name. |
| | 4 | irssta | integer | River node position (1+) to allow cross-reference with river nodes. |

| Record | Field | StateMod Variable | Type | Description | |
|---|---|---|---|---|---|
| 9<br>**Repeat record for `numrun`** | 1 | j | integer | Counter for record type 9. | |
| | 2 | crunid(j) | char(12) | Base flow node identifier. | |
| | 3 | runnam(j) | real(6) | Base flow node name. | |
| | 4 | irusta(j) | integer | River node position (1+) to allow cross-reference with river nodes. | |
| 10<br>**Repeat record for `numdivw`** | 1 | j | integer | Counter for record type 10. | |
| | 2 | cdividw(j) | char(12) | Well identifier. | |
| | 3 | divnamw(j) | real(6) | Well name. | |
| | 4 | idvstw(j) | integer | River node position (1+) to allow cross-reference with river nodes. | |
| 11<br>**Repeat record for every river node `numsta`, for every month of simulation.**<br><br>See the StateMod documentation for a full description of parameters.<br><br>Parameters are grouped as shown in the **.xdd* file. | 1 | dat(1) | real | Demand | Total_Demand |
| | 2 | dat(2) | real | Demand | CU_Demand |
| | 3 | dat(3) | real | Water Supply | From_River_By_Priority |
| | 4 | dat(4) | real | Water Supply | From_River_By_Storage |
| | 5 | dat(5) | real | Water Supply | From_River_By_Exchange |
| | 6 | dat(60 | real | Water Supply | From_Well |
| | 7 | dat(7) | real | Water Supply | From_Carrier_By_Priority |
| | 8 | dat(8) | real | Water Supply | From_Carrier_By_Storage |
| | 9 | dat(9) | real | Water Supply | Carried_Water |
| | 10 | dat(10) | real | Water Supply | From_Soil |
| | 11 | dat(11) | real | Water Supply | Total_Supply |
| | 12 | dat(12) | real | Shortage | Total_Short |
| | 13 | dat(13) | real | Shortage | CU_Short |
| | 14 | dat(14) | real | Water Use | Consumptive_Use |
| | 15 | dat(15) | real | Water Use | To_Soil |
| | 16 | dat(16) | real | Water Use | Total_Return |
| | 17 | dat(17) | real | Water Use | Loss |
| | 18 | dat(18) | real | Station In/Out | Upstream_Inflow |
| | 19 | dat(19) | real | Station In/Out | Reach_Gain |
| | 20 | dat(20) | real | Station In/Out | Return_Flow |
| | 21 | dat(21) | real | Station In/Out | Well_Depletion |
| | 22 | dat(22) | real | Station In/Out | To_From_GW_Storage |
| | 23 | dat(23) | real | Station Balance | River_Inflow |
| | 24 | dat(24) | real | Station Balance | River_Divert |
| | 25 | dat(25) | real | Station Balance | River_By_Well |
| | 26 | dat(26) | real | Station Balance | River_Outflow |
| | 27 | dat(27) | real | Available Flow | Available_Flow |
| | 28 | dat(28) | real | Structure type (Na):<br>• < 0 = Baseflow node (e.g., -10001 indicates a diversion that is a baseflow node).<br>• 0 = Well only.<br>• 1-5000 = Diversion<br>• 5001 – 7500 = Instream flow<br>• 7501 – 10000 = Reservoir | |
| | 29 | dat(29) | real | Number of structures at this node (typically 1). | |

## StateMod B43 Files and Standard Time Series Properties

The standard time series identifier for StateMod binary time series is of the form:

```
Location.StateMod.DataType.Interval~StateModB~PathToFile
```

Time series properties are set using the following guidelines:

- The location part of the time series identifier is taken from the identifier field in the data. The identifier for the specific node type (e.g., diversion) is used, not the river node identifier. The river node identifier is often the same as for the specific node type, but this is not a requirement within StateMod.
- The data source part of the time series identifier is set to `StateMod`, because StateMod has created the output time series.
- The data type is assigned as the parameter name (see record 11 above, without using the group).
- The data interval is assigned as `Month` or `Day`, depending on the file extension.
- The scenario is set to blank.
- The input type is set to `StateModB`.
- The input name is set to the name of the file.
- The units for daily data are assigned as `CFS`. The units for monthly data in the files are average CFS for the month and are converted to `ACFT`, assuming a constant number of days per month, as read from record 4. February normally has 28 days per month in the header and therefore leap years have one fewer days than actual.
- The missing data value is assigned to `-999.0`.
- The description is set to the node name.
- The period is set based on the header information in record 1 (for the year) and record 3 (to determine the start and end months, based on the calendar type).

## Limitations

StateMod binary files have the following limitations:

- The file does not contain a format version; therefore, it is difficult for software to handle changes in the file format.
- The file does not contain header information indicating the source of the file (e.g., the creation date, user, directory, StateMod response file, command line). Therefore, it is difficult to know with certainty how a file was created.
- Leap years are not explicitly handled with 29 days.
- Baseflow nodes in record 9 may have the same identifier as other nodes because any node can be a baseflow node. This can be confusing since software may list the node in more than one list. The software that reads the file filters out duplicate time series identifiers to try to resolve this problem.
- This documentation is limited in that it presents the file format only for the **.b43* file. Additional documentation may be added in the future.

# Appendix
# TSView - Time Series Viewing Tools

Color, 2005-08-05, Original Maintained with TSTool, Acrobat Distiller

## Overview

The TSView package contains integrated software components that can be used with software applications to enable time series viewing capabilities. The main purpose of the TSView package is to provide simple, consistent, and flexible displays that can be used in a variety of applications with little or no reconfiguration. TSView also provides features to configure and process time series products (e.g., graphs), where the time series data are stored separately from the configuration information.

The TSView package has been developed by Riverside Technology, inc., using Java technology. TSView interfaces can be embedded in Java applications and can be used in web pages either as embedded applets or stand-alone windows. TSView tools operate similarly on Microsoft Windows and UNIX operating systems.

This appendix describes general TSView features and can be used as a reference for how to configure and use TSView components. Software program documentation may include specific information about using TSView features.

## Time Series Terminology

The TSView package treats time series as objects that can be read, manipulated, and output in various formats. A time series is defined as having header information (attributes) and data, which usually consists of a series of date/time versus data pairs. Internally, time series are considered to have either regular interval (equal spacing of date/time) or irregular interval (e.g., occasional observations). Regular time series lend themselves to simpler storage and faster processing because date/time information can be stored only for the endpoints. The following basic attributes are stored for each time series:

- data interval as an interval base (e.g., `Month`, `Hour`) and multiplier (e.g., `1` for month, or `24` for hour) - in many cases, the multiplier is `1` and is not shown in output (e.g., `Month` rather than `1Month`),
- data type (e.g., `Streamflow`), which ideally can be checked to determine if a time series contains mean, instantaneous, or accumulated values,
- units (e.g., `CFS`), which ideally can be used to make units conversions and look up precision for output,
- period of record, using dates that are of an appropriate precision for the interval,
- data limits (the maximum, minimum, etc.),
- description (generally a station, structure, or sensor name),
- missing data value (used internally to mark missing data and trigger data filling, often `-999`),
- comments (often station comments, if available),
- genesis history (a list of comments about how the time series was created).

In order to uniquely and consistently identify time series, a multi-part *time series identifier* is employed, having the following parts:

- location (or location-sublocation)
- data source
- data type (or datatype-subdatatype)
- data interval (time step)
- scenario

and optionally:

- sequence number (currently being evaluated)
- input type
- input name

These time series attributes are typically concatenated into a time series identifier string. The following example illustrates how the basic identifier parts can be used (without input type and name):

```
12345678.USGS.Streamflow.DAY.HIST
```

The above example identifies a `USGS` streamflow gage identified as location `12345678`, at which historic average daily flow data are available. If possible, data types appropriate for the input type should be used to avoid confusion; however, time series file input types often do not contain a simple data type abbreviation (see the input type appendices in the TSTool documentation for more information). The above example illustrates that the scenario can be used to qualify the data (in this case as historic data, `HIST`). The scenario is often omitted. When the scenario is used, it often indicates some specific condition (e.g., `FLOOD`, `DROUGHT`, `HIST`, `FILLED`)

The optional input type and input name are used to specify the time series input format and storage location, especially in cases where the identifier is saved in a file and the input type is needed for later processing. For example:

```
12345678.USGS.Streamflow.DAY.HIST~USGSNWIS~C:\data\12345678.txt
12345678.USGS.Streamflow.DAY~HydroBase
```

The first example illustrates a time series identifier for a USGS National Water Information System data file. The second example illustrates the identifier for the same time series, in the HydroBase database. Using the input parts of the identifier allows software to transparently locate the data, and for the above examples, would allow the time series to be read from each input source and compared.

The use of the input type and name is being phased into TSView and related components. Input types that have been added to software more recently (e.g., as of version 05.04.00 of the TSTool application) use the new convention and older input types are being updated accordingly. The TSTool appendices that describe each input type identify issues with compatibility.

Using the above time series identifier convention omits use of time series attributes like the period of record and the units, even though these attributes could conceivably be used to distinguish between time series that are otherwise the same. Instead, it is assumed that the period of record and units can be determined from the input and do not need to be part of the identifier. If necessary, different input files can be used to further differentiate time series.

The TSView components use the time series identifiers extensively to locate and manage time series. For example, graph properties for each time series are cross-referenced to time series by using the identifiers. Perhaps most importantly, the time series identifiers as simple strings can be stored in files and can be used by a variety of software to consistently and reliably locate data for processing.

The following table summarizes important time series terminology.

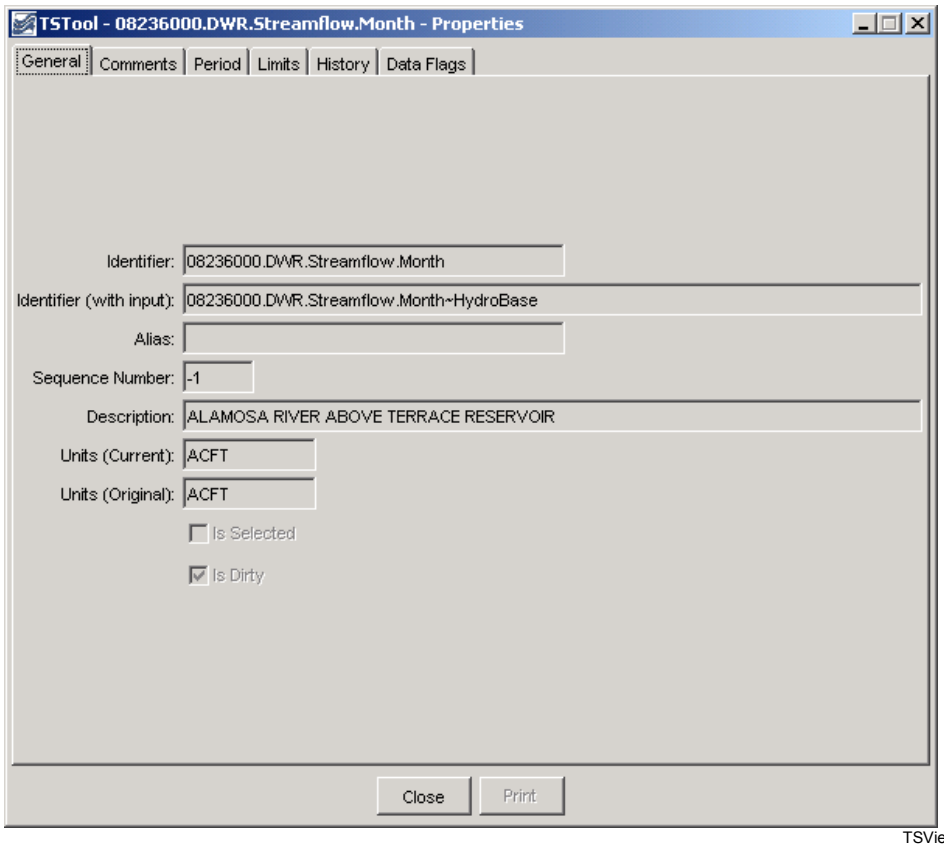
**Time Series Terminology (listed alphabetically)**

| Term | Description |
|---|---|
| *Data Interval* | Time interval between time series data values. If a *regular* time series, the interval is constant. If an *irregular* time series, the interval can vary. Intervals are represented as an optional multiplier followed by a base interval string (e.g., `1MONTH`, `24HOUR`) or `IRREGULAR` for irregular time series. |
| *Data Source* | A string abbreviation for a data source, which is part of the time series identifier and typically indicates the origin of the data (e.g., an agency abbreviation, or a model name if the result of a simulation). |
| *Data Type* | A string abbreviation for a data type, which is part of the time series identifier (e.g., `Streamflow`). |
| *Date/Time Precision* | Date/time objects used with time series have a precision that corresponds to the time series data interval. The precision is typically handled transparently but it is important that the precision is consistent (e.g., monthly data should not use date/time objects with daily precision). Displaying time series with various precision usually results in the smallest time unit being used for labels. |
| *Input Name* | A string input name corresponding to an input type, which is part of a time series identifier. For database input types, the name may be omitted or may be the name of the database connection (e.g., `ARCHIVE`). For input files, the name is typically the name of the file. |
| *Input Type* | A string abbreviation that indicates the input type (persistent format) for a time series, and is part of a time series identifier. This is often the name of a database (e.g., `HydroBase`, `RiversideDB`) or a standard data file format type (e.g., `StateMod`, `MODSIM`, `RiverWare`). |
| *Location* | A string identifier that is part of a time series identifier and typically identifies a time series as being associated with a location (e.g., a stream gage or sensor identifier). The location may be used with certain input types to determine additional information (e.g., station characteristics may be requested from a database table using the location). |
| *Scenario* | A string label that is part of a time series identifier, and serves as a modifier for the identifier (e.g., `HIST` for historical). |
| *Sequence Number* | A number indicating the sequence position of a time series in a series. For example, possible time series traces may be identified with a sequence number matching the historical year for the data. The use of sequence numbers with traces is being evaluated. |
| *Time Series Product* | A graph or report that can be defined and reproduced. See the **Time Series Product Reference** section. |
| *Time Step* | See Data Interval. |

## Time Series Properties Interface

Time series properties are displayed in a tabbed panel as appropriate in applications (e.g., the TSTool application can display the properties after time series are read and listed in the TSTool interface). Differences between time series input types may result in variations in the properties (e.g., some input types do not have descriptions for time series). The following figures describe the properties tabs. The size of each tabbed panel is set to the size of the largest tab; therefore, some tabbed panels are not completely filled.

### *Time Series Properties - General*



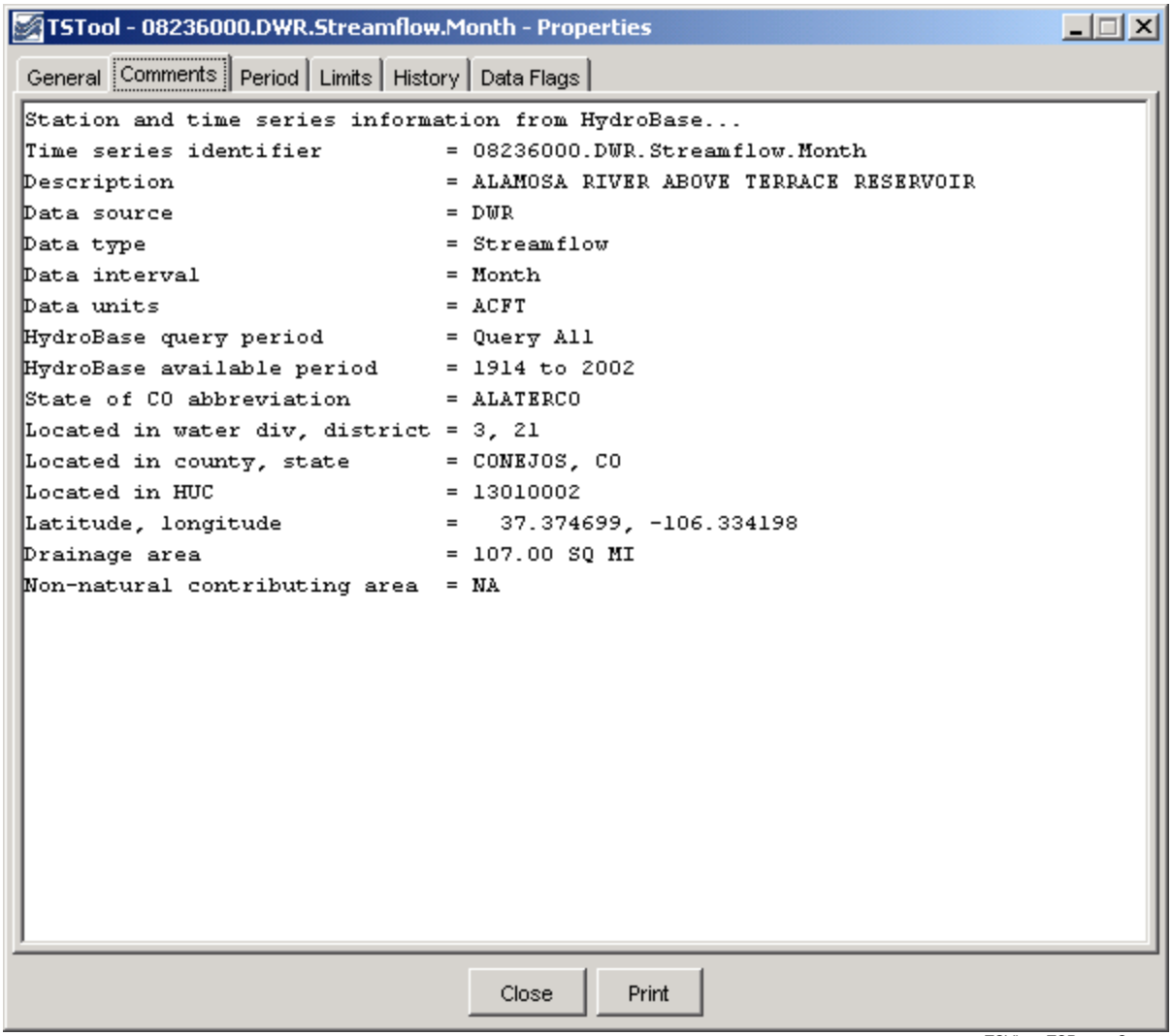

TSView_TSProps_General

**Time Series Properties - General**

General time series properties are as follows:

| | |
|---|---|
| ***Indentifier*** | The five-part time series identifier without the input type and name. This identifier is often used internally in applications to manage time series. See the **Time Series Terminology** section for a complete explanation of time series identifiers. |
| ***Identifier (with input)*** | The full identifier, including the input type and name (if available). The input type and name indicate the format and storage of the data. |
| ***Alias*** | A time series may be assigned an alias to facilitate processing (e.g., the alias is used by the TSTool application in time series commands). |
| ***Sequence Number*** | If the time series is part of a series of traces, the sequence number is used to identify the trace. Often it is the year for the start of the trace. |
| ***Description*** | The description is a mid-length phrase (i.e., longer than the location but shorter than comments) describing the time series (e.g., `XYZ RIVER AT ABC`). |
| ***Units (Current)*** | The units that are currently used for data. The units may have been converted from the original. |
| ***Units (Original)*** | The units in the original data source. |

***Time Series Properties – Comments***



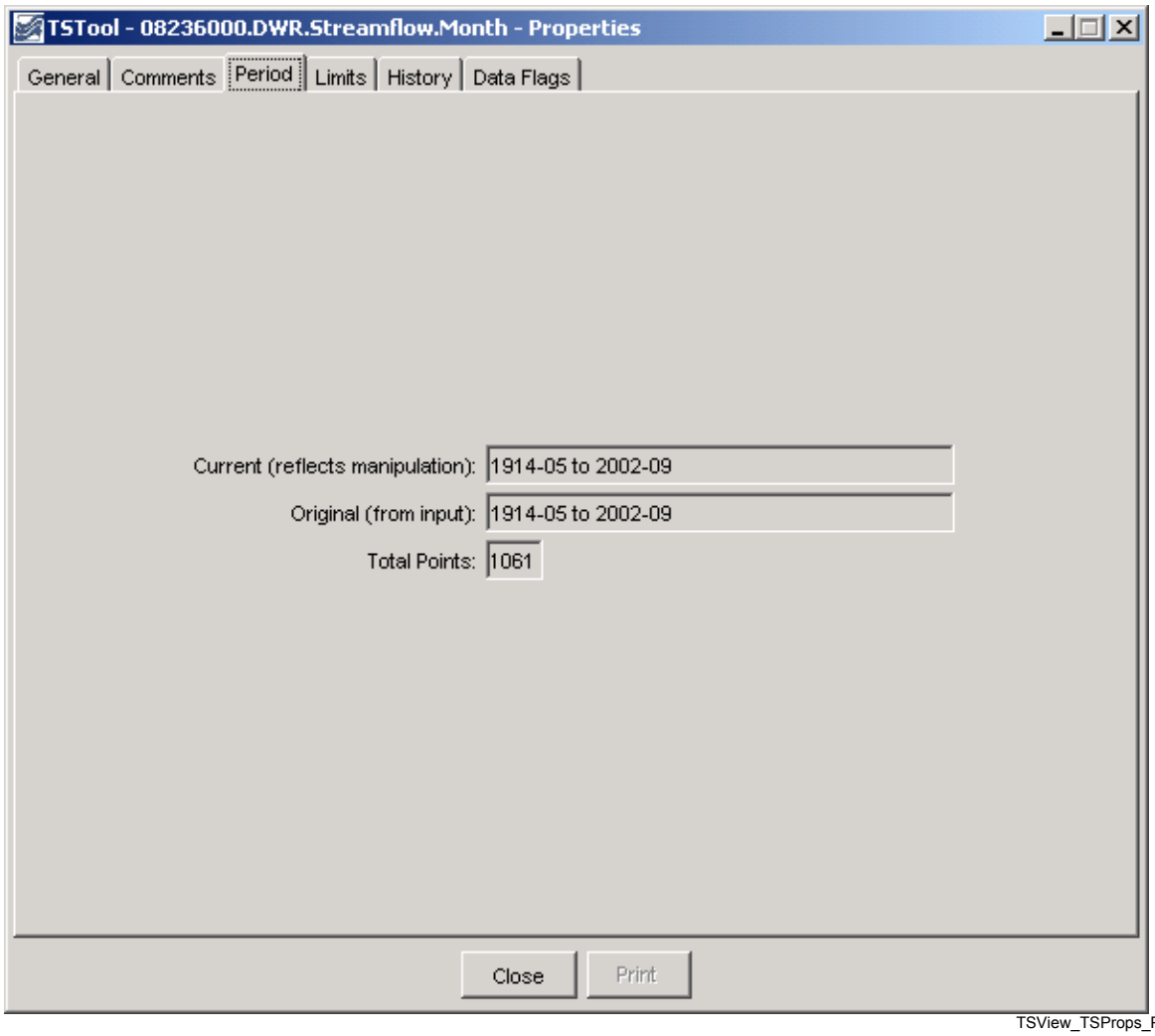

TSView_TSProps_Comments

**Time Series Properties - Comments**

Comments for time series can be created a number of ways and may be formatted specifically for an application. Common ways of creating comments are:

- read comments from the original data source - this is ideal; however, electronic comments are often not available (e.g., the USGS previously published comments for data stations in hard copy water reports; however, comments may no longer available electronically),
- format comments from existing data pieces (e.g., the figure illustrates a standard set of comments for State of Colorado data, using the HydroBase input type).

***Time Series Properties – Period***




TSView_TSProps_Period

**Time Series Properties - Period**

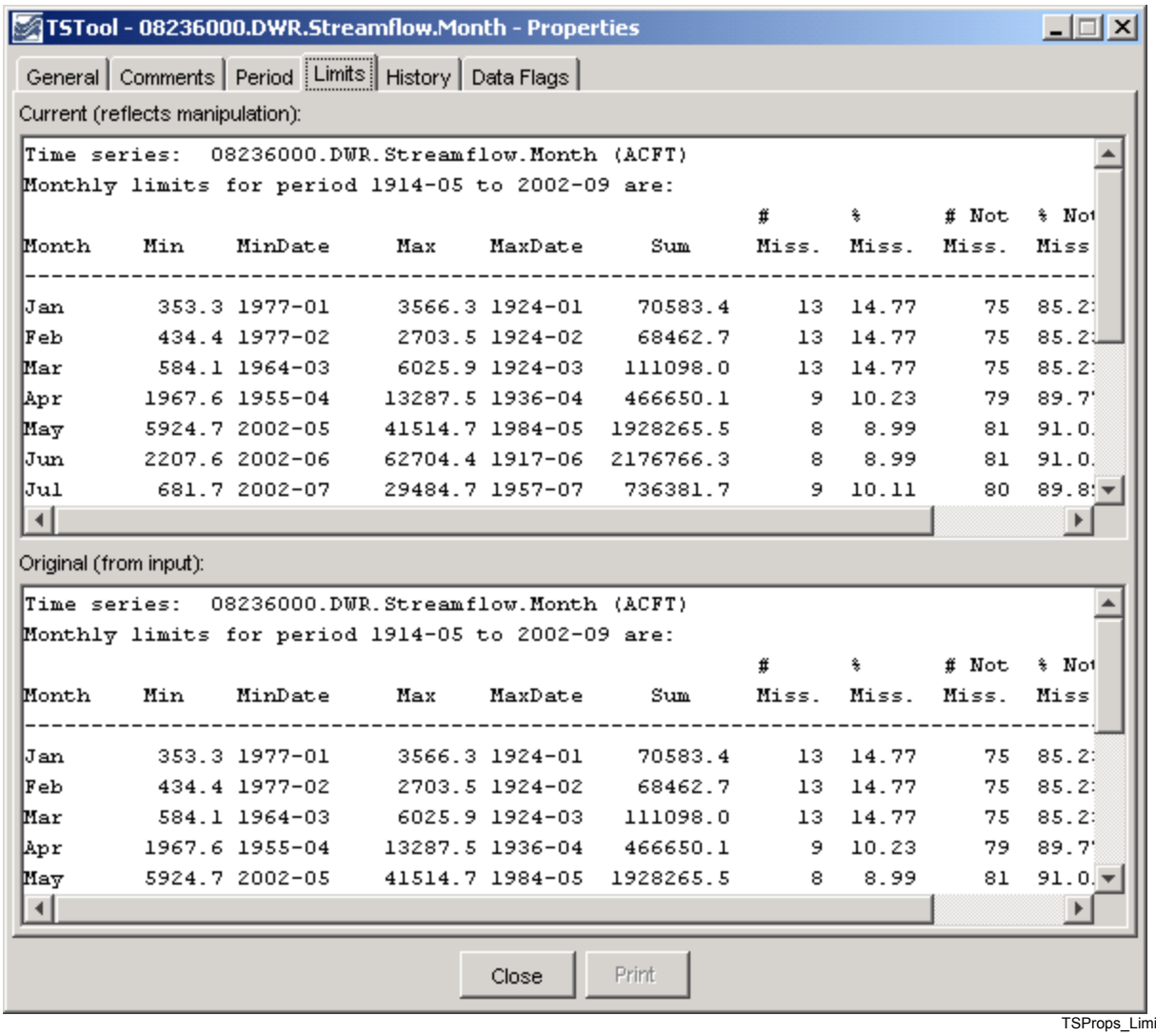
Properties related to the period are as follows:

***Current (reflects manipulation)*** The current period is used to allocate computer memory for the time series data. This period may be set by an application (e.g., when creating model input files a specific period may be used). The precision of the date/time objects should generally be consistent with the time series data interval.

***Original (from input)*** The original period can be used to indicate the full period available from a database. Setting the original period can sometimes be complicated by how missing data are handled (e.g., a database or file may indicate a certain period but a much shorter period is actually available).

***Total Points*** Total number of points in a time series. If a regular time series, this can be computed from the period. If an irregular time series, the number of points is determined from a count of all data values. The data points may include missing data – see the data limits for additional information.

***Time Series Properties – Limits***



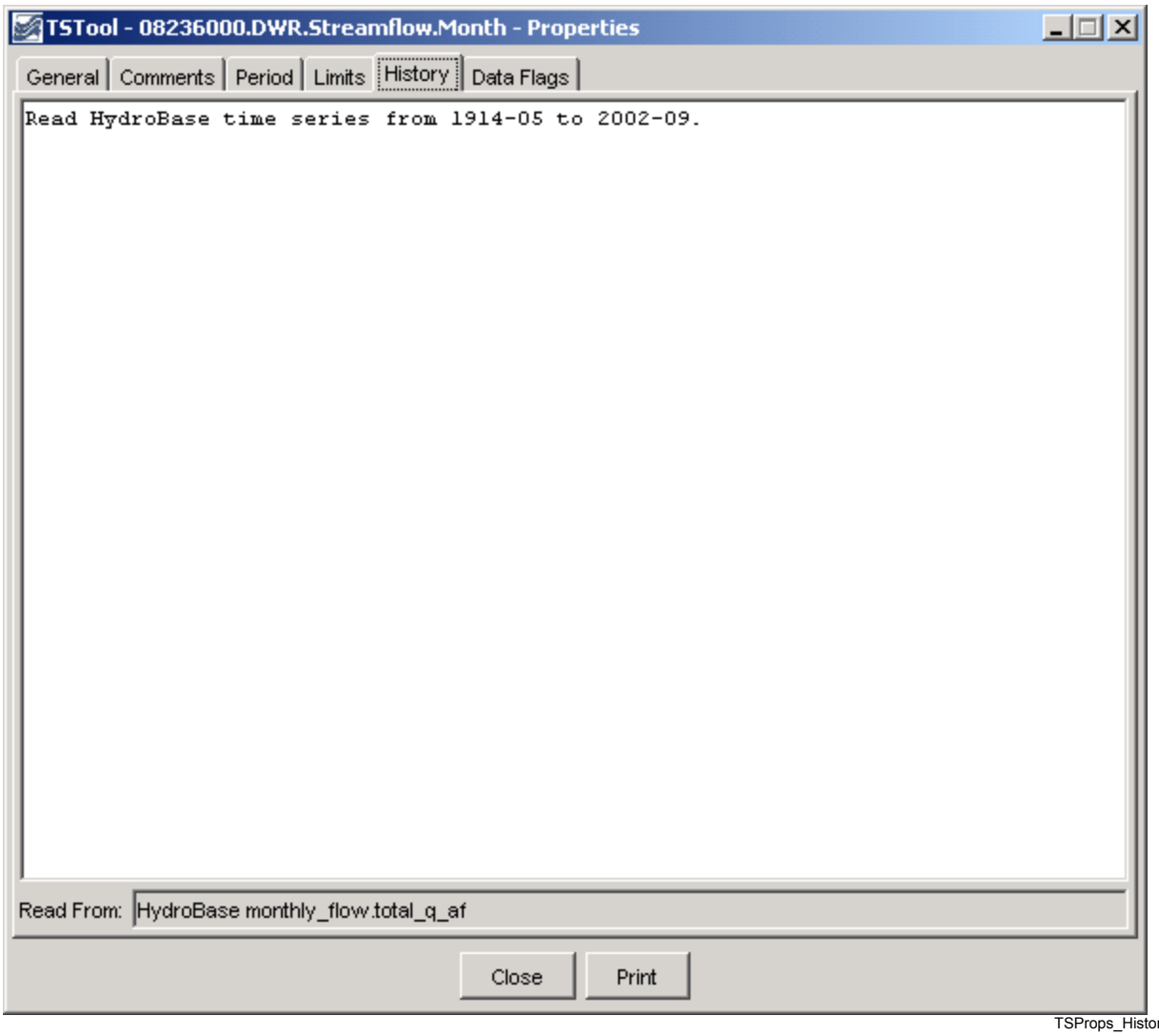

TSProps_Limits

**Time Series Properties - Limits**

Time series limits are determined for both the current data (top in figure) and the original data (bottom in figure). This is useful because the original data may contain missing data, which are later filled. The data limits are displayed consistent with the data interval. In the example shown, limits are computed for each month. For other time series having other intervals, only overall data limits may be computed.

Theoretically, it is possible that a daily time series could have day limits (e.g., max/min values for each day of the year), month limits (e.g., computed as an average of the daily values by month), and year limits (e.g., computed as an average of all daily values in a year). However, automatically including this level of detail decreases performance and it is difficult to automatically make the right decisions (e.g., about whether to average or total values). Consequently, the limits are currently computed in a basic fashion on the raw data (no interval changes).

***Time Series Properties – History***



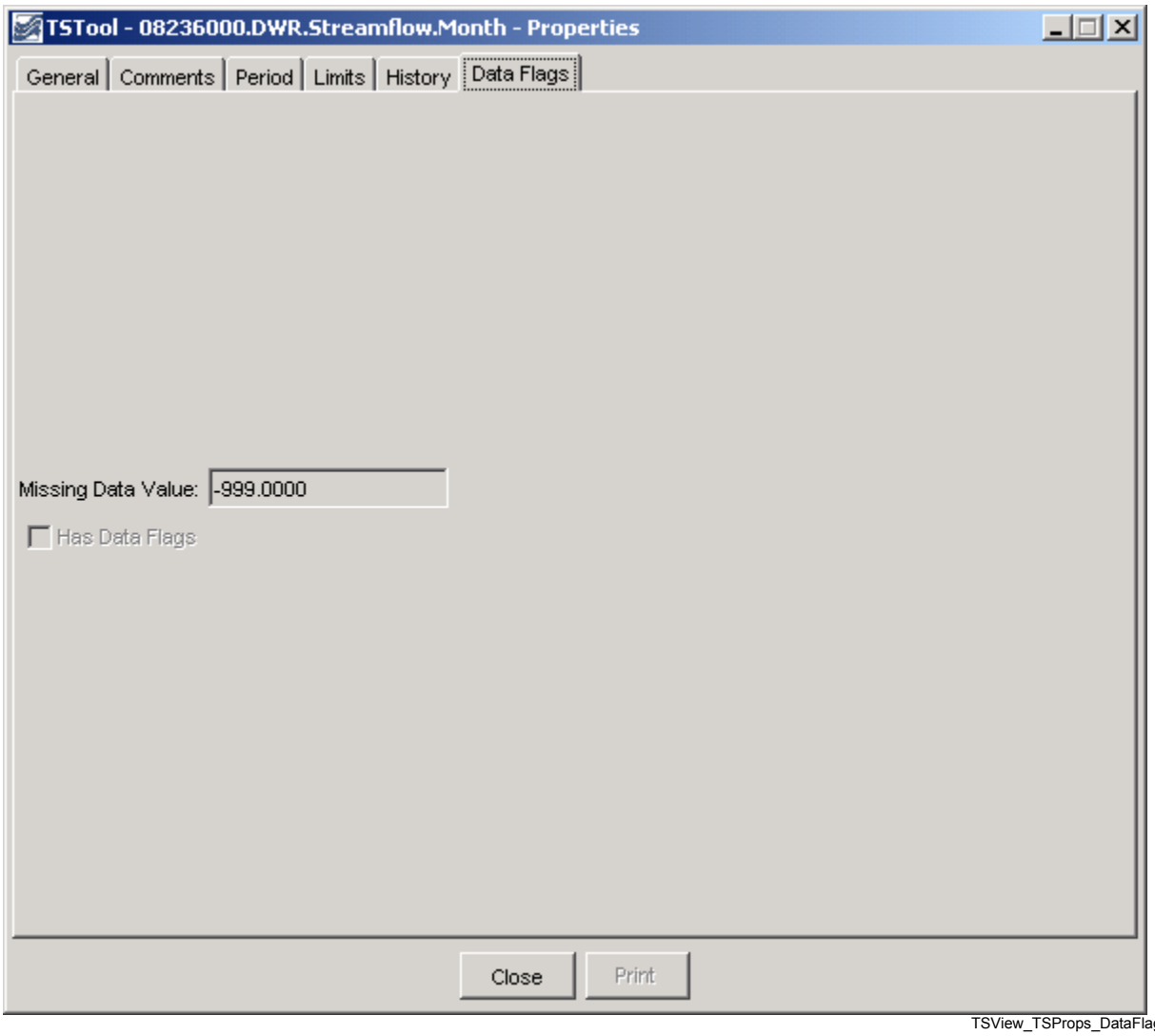

TSProps_History

**Time Series Properties - History**

The time series history (sometimes called the *genesis history*) is a list of comments indicating how the time series has been processed. The completeness of this history is totally dependent on the time series input/output and manipulation software. Although efforts have been made to add appropriate comments as time series are processed, enhancements to the history comments are always being considered.

At the bottom of the history list (see ***Read From***) is the input name that was actually used to read the data. This input name may or may not be exactly the same as the input name in the time series identifier. For example, if reading from a HydroBase database, the time series identifier may specify an input type of `HydroBase` and no input name (because the software knows from the other parts of the time series identifier which database tables to read). However, it is also useful to know the actual table that is read in order to help users and developers understand the data flow. If reading from a file input type, the ***Read From*** information will show the full path to the file; however, the input name in the time series identifier may only include a relative path.

***Time Series Properties – Data Flags***




TSView_TSProps_DataFlags

**Time Series Properties - Data Flags**

Time series data flags contain information that describe the quality of a data point. The missing data value indicates a special number that is used to indicate that a data value is missing at a point. Currently only floating point values are recognized; however the `NaN` (not a number) value is generally supported for input types that use the convention. All time series are typically assigned a missing data value.

The ***Has Data Flags*** checkbox indicates whether the time series has data flags. Full support for data flags is being phased in, based on whether an input type supports data flags. The USGS NWIS file format is an example of an input type that supports data flags (e.g., `e` is used to indicate estimated data).

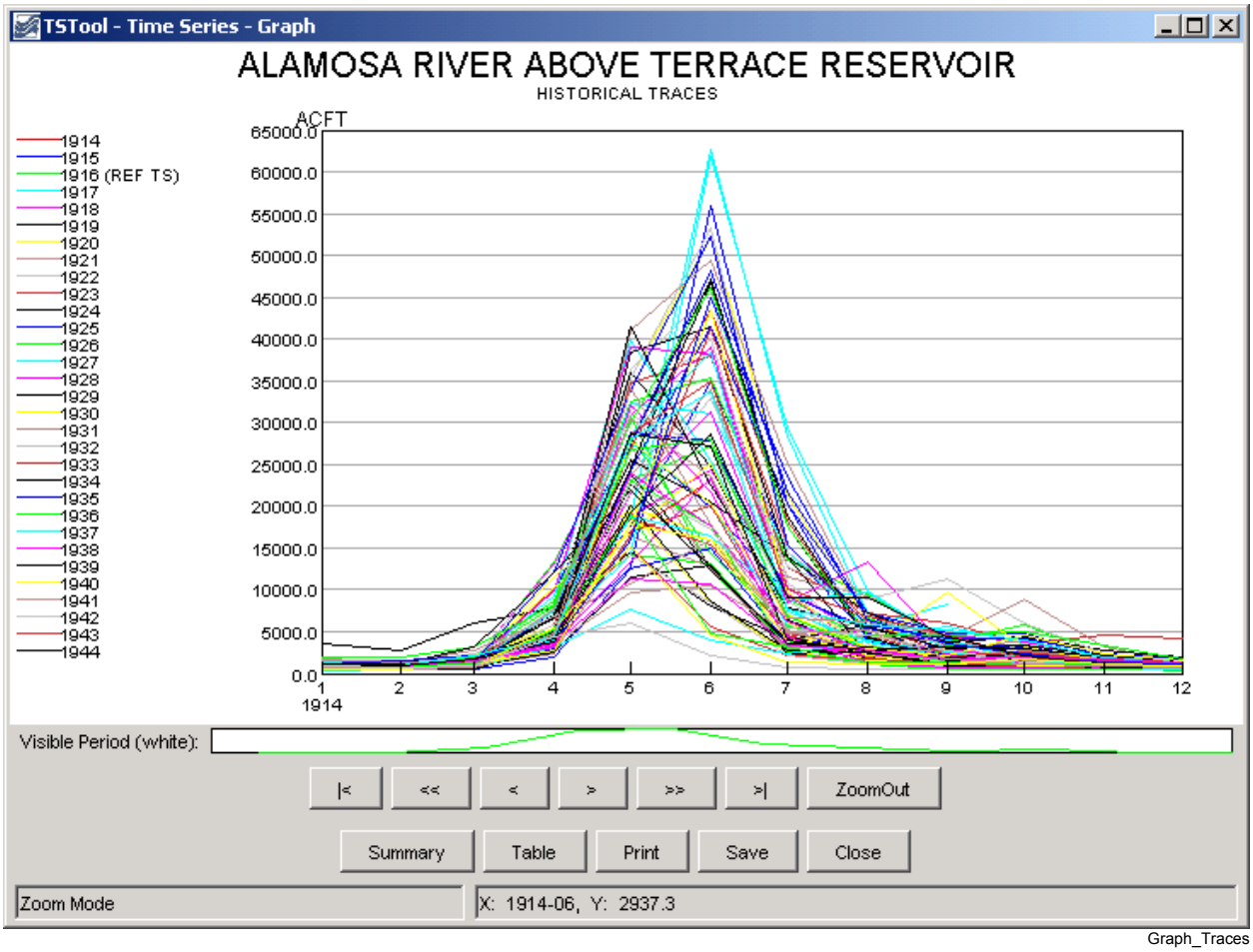
One of the issues with fully supporting data flags is that different input types (and even different data within an input type) treat data flags inconsistently. Therefore, it is easier to add data flags to time series visualization tools (e.g., label points on a graph with the flag) than to integrate data flags in data filling and analysis features. Features related to data flags will continue to be enhanced.

## Time Series Traces

The term *time series traces* in general refers to a group of time series, often shown in overlapping fashion. Common uses of time series traces are:

- separate a full time series into annual traces and plot them on top of each other, shifted so that they all start at the same date/time,
- run a model or analytical tool multiple times, with input being a series of input traces, and generating a series of output traces, in order to produce probabilistic simulations.

The power of using traces is that a large amount of data can be used to visualize and study statistical qualities of the data, as shown in the following figure.




Graph_Traces

**Example Graph for Time Series Traces**

The TSView package supports time series traces at various levels. Time series properties include a sequence number that can be used to identify a time series as being in a group of traces. However, for data management and viewing, time series identifiers often do not indicate whether a time series is in a group of traces (the sequence number is managed internally). Full support of time series traces is being phased in.

Currently, applications like TSTool include features to create time series traces and TSView tools can be used to view the time series as if they were separate time series. Additional visualization features are being enabled as time allows.

The following sections describe the different time series views that are available in TSView. Although most illustrations using simple time series, most features are also available for use with traces.

## Time Series Views

The main components of the TSView package are configured to provide multiple views for time series data. The three main views that are available are:

1. ***Graph*** - line, bar, or other graph
2. ***Summary*** - text report suitable for the data type and interval
3. ***Table*** - spreadsheet-like table with scrolling, suitable for export to other tools

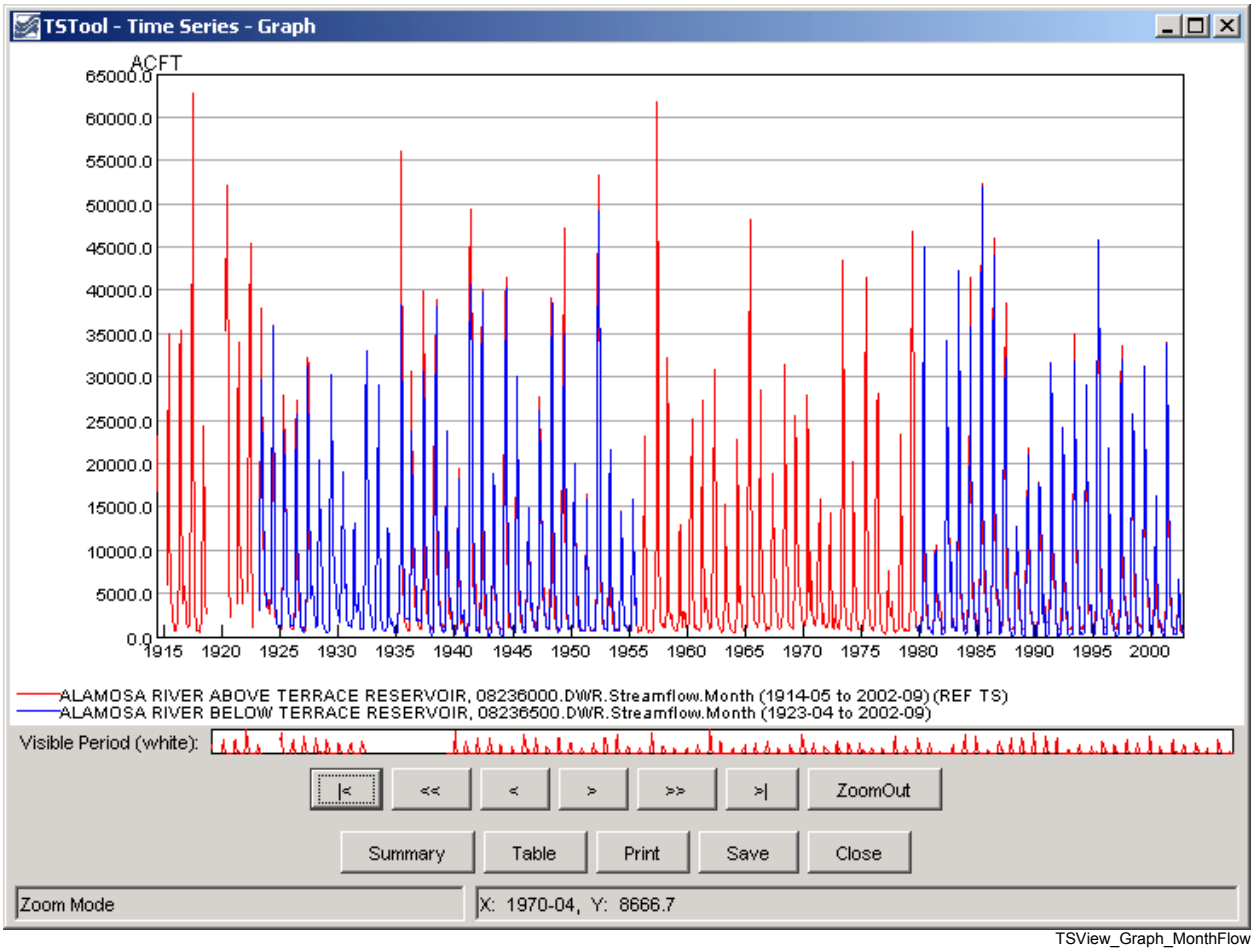
The initial view for a time series list is typically determined from the actions of the software user. For example, a ***Graph*** button may be displayed on a screen, which when pressed will result in a graph being displayed. The time series that are displayed in the view can typically contain one or more time series (some graph types may have a restriction on the number of graph types). To increase performance and capacity, the TSView package as much as possible uses a single copy of the time series data for visualization. For example, to generate graphs, the data for the time series objects are used directly rather than being copied into a graphing tool's data space. This also allows TSView to more easily display different data intervals on the same view because the data do not need to be forced into a consistent grid data structure.

The following sections describe the three time series views. The graph view type requires more extensive explanation due to the variety of graph properties.

## Time Series Graph View

The graph view for time series supports a variety of graph types. The features of the various graph types will be discussed in detail in the following sections, starting with basic graph types, followed by more specific types.

Typically, the graph type is selected in the application (e.g., menus are available in TSTool for selecting the graph type for a list of time series). In many applications, the graph type often defaults to a line graph. The following figure illustrates a line graph for two monthly streamflow time series.




TSView_Graph_MonthFlow

**Example Line Graph for Monthly Streamflow**

The graph view is divided into the following main areas:

***Graph Canvas*** The graph canvas is the area where the graph and legend are drawn. This area is used to interact with the graph (e.g., zoom). More than one graph can be drawn in the canvas (see the **Time Series Product Reference** section for additional details). If zooming is supported for the graph, a box can be drawn with the mouse to zoom in to a shorter period. Right-click over a graph of interest to show the popup menu for graph properties and analysis details (e.g., regression results). The canvas area is essentially a preview of a printed graph.

***Reference Graph***

The reference graph below the main graph canvas shows the current view extent (the white area in reference graph in the figure above). The reference graph is only shown for graph types that support zooming. If shown, it can be used for zooming, similar to the main graph. The time series with the longest period of record is drawn in the reference graph to illustrate variations in the data over time (this time series is noted in the main graph legend with REF TS – this label is not shown in printed output).

***Scroll/Zoom Buttons***

Under the graph areas is a layer of buttons used for zooming. The ***Zoom Out*** button will zoom to the full extent of the data.

The other buttons facilitate scrolling through data as described below. For all scrolling operations, the visible graph extent (or *page*) is maintained during the scroll. Scrolling can use the buttons or keys described below. To use the keyboard, first click in the main graph canvas to shift focus to that area.

| | | |
|---|---|---|
| **\|<** | ***Home*** | Scroll the visible window to the start of the period. |
| **<<** | ***Page Down*** | Scroll the visible window one full page to the left (earlier in time). |
| **<** | ***Left Arrow*** | Scroll the visible window 1/2 page to the left. |
| **>** | ***Right Arrow*** | Scroll the visible window 1/2 page to the right (later in time). |
| **>>** | ***Page Up*** | Scroll the visible window a full page to the right. |
| **>\|** | ***End*** | Scroll the visible window to the end of the period. |

***Main Buttons***

The bottom row of buttons provides features for displaying other views, printing, and exporting:

| | |
|---|---|
| ***Summary*** | Display the summary view for the time series (see the **Time Series Summary View** section). |
| ***Table*** | Display the table view for the time series (see the **Time Series Table View** section). |
| ***Print*** | Print the graph. Because the physical extents of the printed page are different from the visible window, the printed graph may not exactly match the viewed version (e.g., more or less axis labels may be used). |

***Save*** Save the graph as a Portable Network Graphic (PNG) or JPEG graphic, a DateValue file (a useful time series format), or a *Time Series Product* file (see the **Time Series Product Reference** section) by selecting from the choices. Depending on the main application, saving to a database as a time series product may also be enabled.

***Close*** Close the graph window. If related summary or table windows are still visible, the graph view can be quickly re-displayed by pressing the ***Graph*** button on the other view windows. If the graph properties have been changed but have not been saved, a warning will be displayed.

***Status Message Area*** The lower-left status message area is used to provide general user instructions and feedback.

***Mouse Tracker Area*** The lower-right status message area is used to indicate the position of the mouse on a graph, in data units. The coordinates are typically shown using an appropriate precision as determined from the time series date/time precision and data units.

Within each graph canvas it is possible to draw more than one graph, each with its own titles, legend, etc. The **Time Series Product Properties** section (below) provides an example and discusses how to edit graph properties. The **Time Series Product Reference** section describes in detail the format of *Time Series Product* files. These files, when saved from the graph view, can be used to recreate a graph interactively or in batch mode, at a later time.

Because TSView is a general tool, a number of rules are in place when viewing time series in graphs (see the **Time Series Product Properties** section for information on changing specific graph properties to override the defaults):

1. Time series plotted on the same graph should generally have the same units or have units that can be readily converted. If the units are not consistent, you will be warned and the units will be displayed in the legend rather on the axis. (A future enhancement may allow multiple axes, each with different units.)

2. Time series can have different data intervals (e.g., daily data can be plotted with monthly data). However, other output options, such as reports, may not allow the same flexibility. It is important to understand the data type characteristics. For example, some data are instantaneous (e.g., real-time streamflow) whereas other data are accumulated (e.g., precipitation) or mean (e.g., mean temperature). Therefore, the representation of the data may need to be selected with care to ensure consistency. For example, some data intervals and types may be better represented as bars and others as lines or points.

3. Data values are plotted at exactly the point that they are recorded. The plot positions are determined using the year as the whole number and months, days, etc. to determine the fractional part of the plot position. The end-user does not typically see these computed positions because labeling uses data units, including dates. The plot positions are determined from the dates associated with data and no

adjustments are made to plot in the middle an interval. For example, monthly data are plotted at the first day of the month (not day 15). Properties to override this convention are being evaluated. Bar graphs allow you to select whether the bars are drawn to the left or right of the date, or centered on the date. This allows flexibility to show a period over which a value was recorded, if appropriate.

4. The mouse coordinates that are displayed by the mouse tracker are computed by interpolating screen pixels back to data coordinates (which involves a conversion of the plot position to date/time notation). Consequently, the values shown may be rounded off (depending on the zoom extent and data precision). The mouse coordinates are displayed based on the precision of the time series data. When moved, the mouse will display the same date until the date changes within the given precision. For example, for monthly data, moving the mouse left to right, the mouse coordinate will display as `1999-01` as soon as the date changes from `1998-12` to `1999-01`. The label will remain at `1999-01` until `1999-02` is encountered. Because data values are drawn at points, you should therefore always position the mouse slightly to the right of the point to see the date corresponding to the value. This is very important for bar graphs because the bar may extend over several dates. If specific values need to be determined, use the summary or table views.

5. Labels for axes are determined automatically in most cases based on the font requirements, available display space, and data range. Major and minor tic marks are drawn to help determine the data coordinates. Labels are redrawn as the visible period is changed.

6. Graphs that can be zoomed do not allow the vertical axis to be re-scaled on the fly. This capability is being evaluated.

7. Currently, graph types cannot be mixed for time series on a graph. In other words, a graph cannot contain a bar graph for one time series and a line graph for another time series. This ability may be added in the future. A work-around is to use multiple graphs on a page (see the **Time Series Product Properties** section for an example).

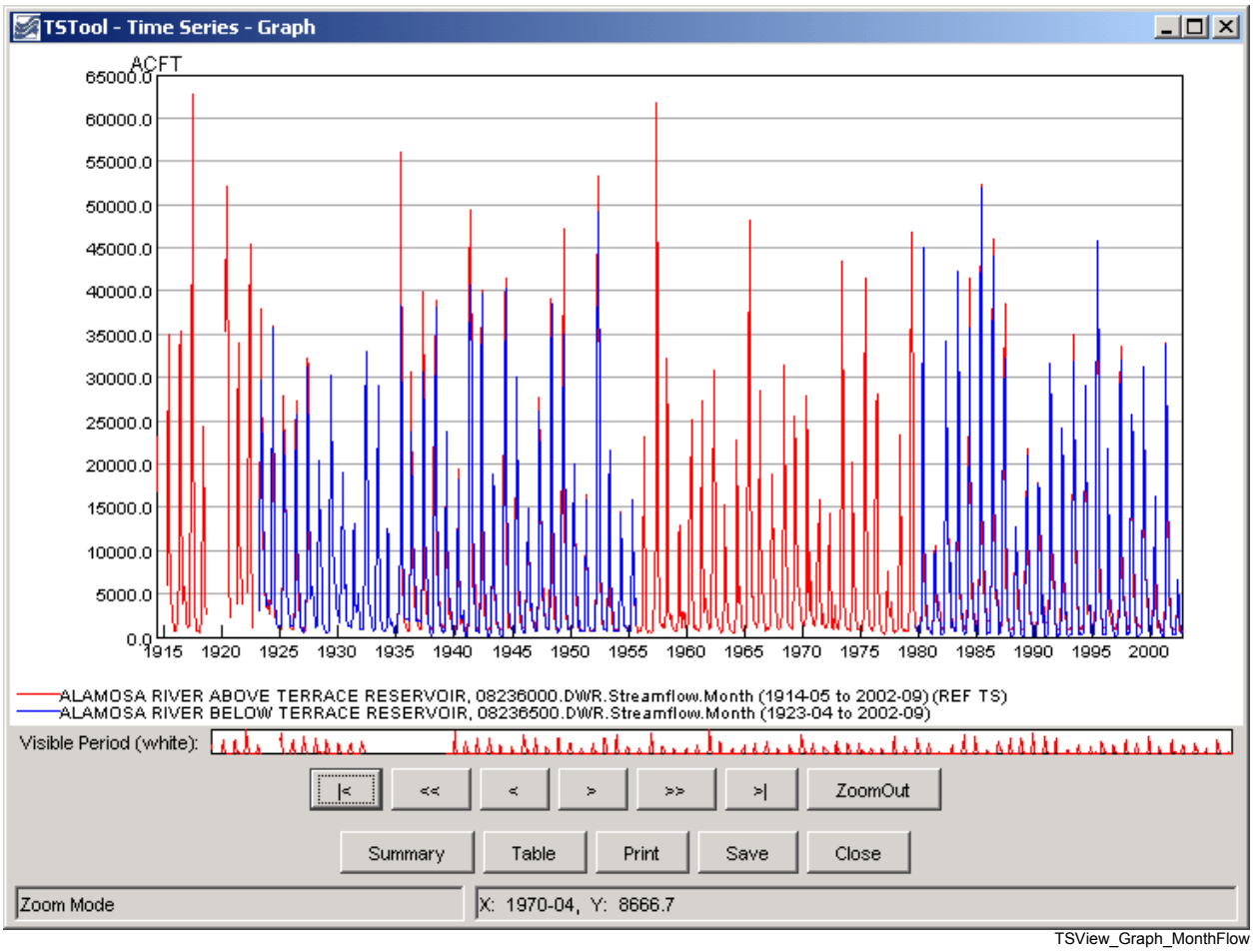
8. The precision used to format graph labels is determined from data unit information provided by the application. This generally produces acceptable graphs. However, in some cases, the range of values being plotted results in inappropriate labels where label values are truncated and/or repeated.

9. Graph types can be changed after the initial display, with limitations. Graphs can be switched between simple types like line and bar graphs; however, simple graphs cannot be changed to more complex types.

The following sections describe various graph types supported by the TSView package. Graph properties are mentioned in some sections. The discussion of how to change graph properties is included in the **Time Series Product Properties** section after the graph type descriptions.

### *Line Graph*

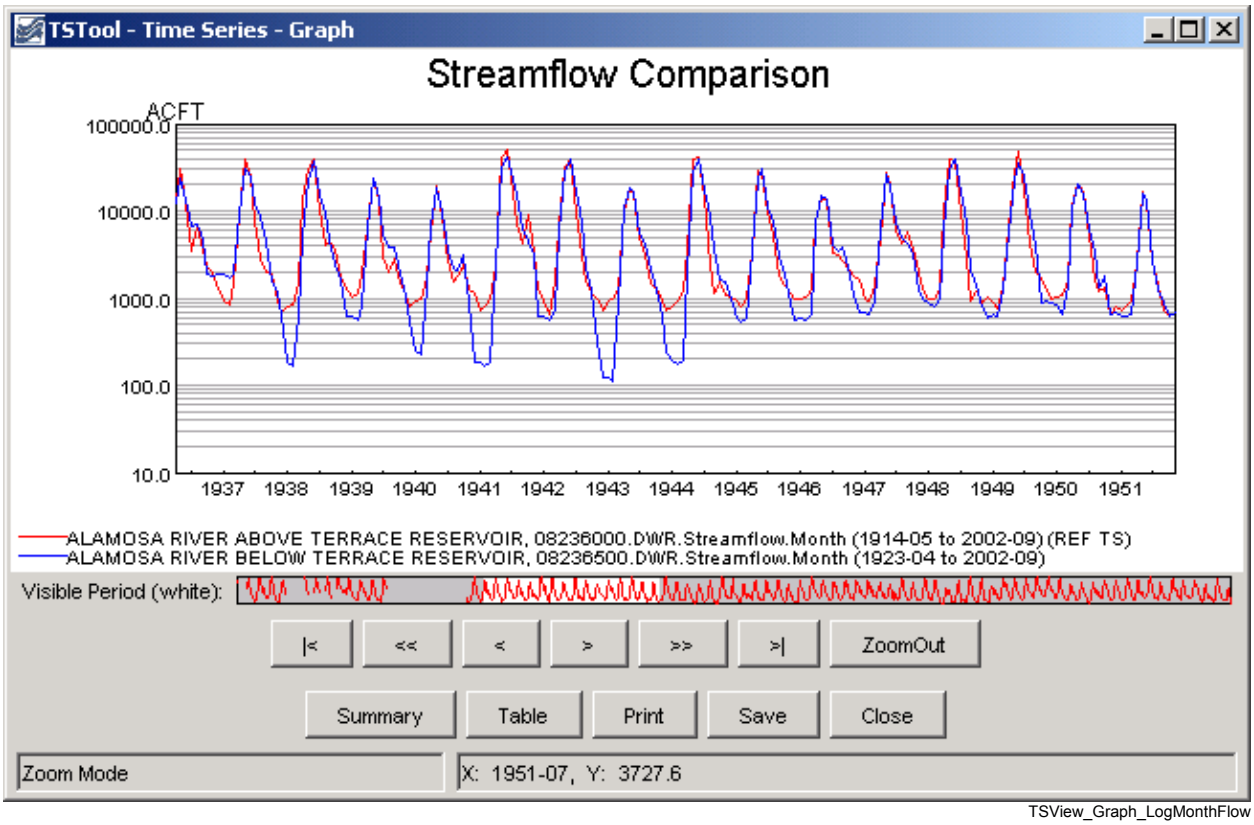
Line graph features have been illustrated in previous discussion. The line graph type is also used to generate graphs with only points by setting the line style to `None` (for example, software that displays daily data where gaps are expected may default to using symbols and no line).




TSView_Graph_MonthFlow

### *Line Graph - Log Y Axis*

Log-axis line graphs are similar to simple line graphs. The following figure illustrates a typical graph.




TSView_Graph_LogMonthFlow

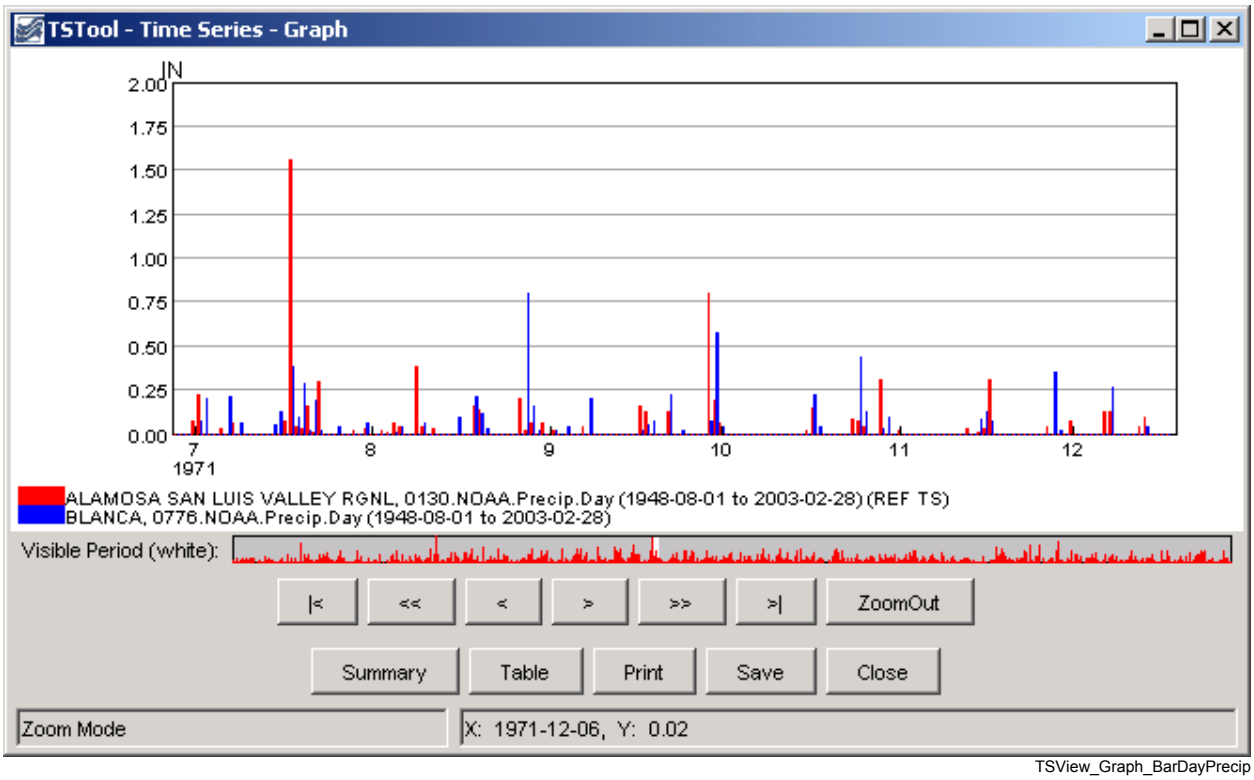
**Example Log Y Axis Graph showing Monthly Streamflow**

Characteristics of the log plot are:

- If the minimum data value is <= `0.0`, then `.001` is used for the minimum plotting value.

### *Bar Graph*

The bar graph type produces a graph with parallel vertical bars, as shown in the following example:




TSView_Graph_BarDayPrecip

**Example Bar Graph showing Daily Precipitation**

The above example illustrates that at the given zoom extent (which is a small part of the full period - see the white area in the reference graph), labels are drawn for months. Zooming in more would display the day in the labels. The mouse tracker in all cases shows days since that is the precision of the data. Characteristics of the bar graph are as follows:

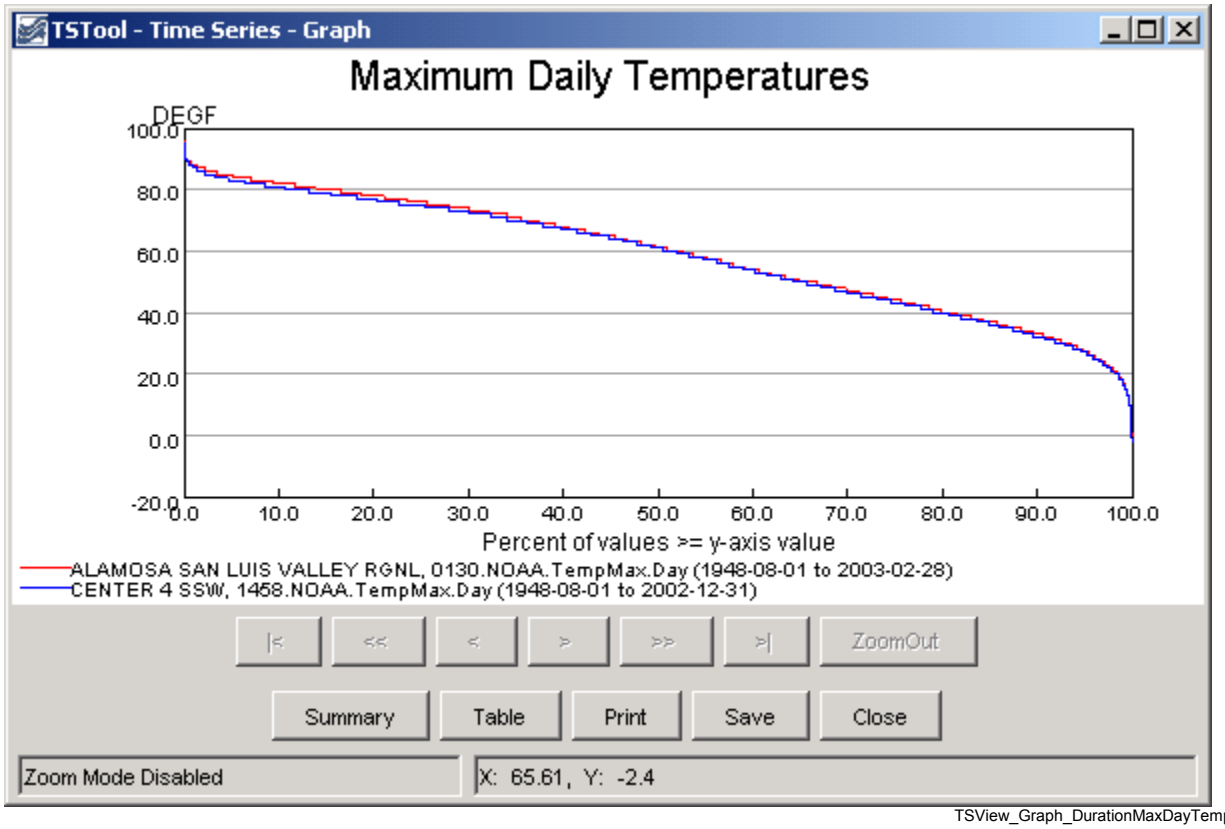
- Bars can be plotted centered on, to the left of, or to the right of the dates. If multiple time series are plotted, the overall total width of the bars will correspond to one data interval. If drawn to the left of the date, the bars for all graphed time series are drawn to the left of the date. If drawn to the right of the date, the bars for all graphed time series are drawn to the right of the date. If centered on the date, half the bars are drawn to the left of the date, and half to the right
- Bar widths are determined based on the number of time series being plotted. Monthly time series use a slightly narrower bar (larger gap between bars) because the number of days in a month varies. To make bars stand out better, a white line may be drawn to separate adjacent bars. If bars are very narrow the line is not drawn. Bars will always be drawn at least one pixel wide, even if this obscures neighboring bars (zoom in to see more detail). Round-off in drawing bars may result in some bars being slightly wider or narrower than other bars.
- Bars always end at the zero value on the Y axis. In other words, bars extend up or down from zero.
- The mouse cursor display dates relative to the axis and does not determine the data value relative to edges of the bars. For example, if bars are plotted centered on dates, 1/2 of the bar will actually be in the previous date, according to the mouse tracker.

### *Double Mass Curve*

Double mass curves are currently disabled. An alternative is to use the TSTool application and generate cumulative time series, which can be viewed in a line graph.

### *Duration Graph*

A duration graph indicates the range of values in a time series and how often they occur, as shown in the following example:




TSView_Graph_DurationMaxDayTemp

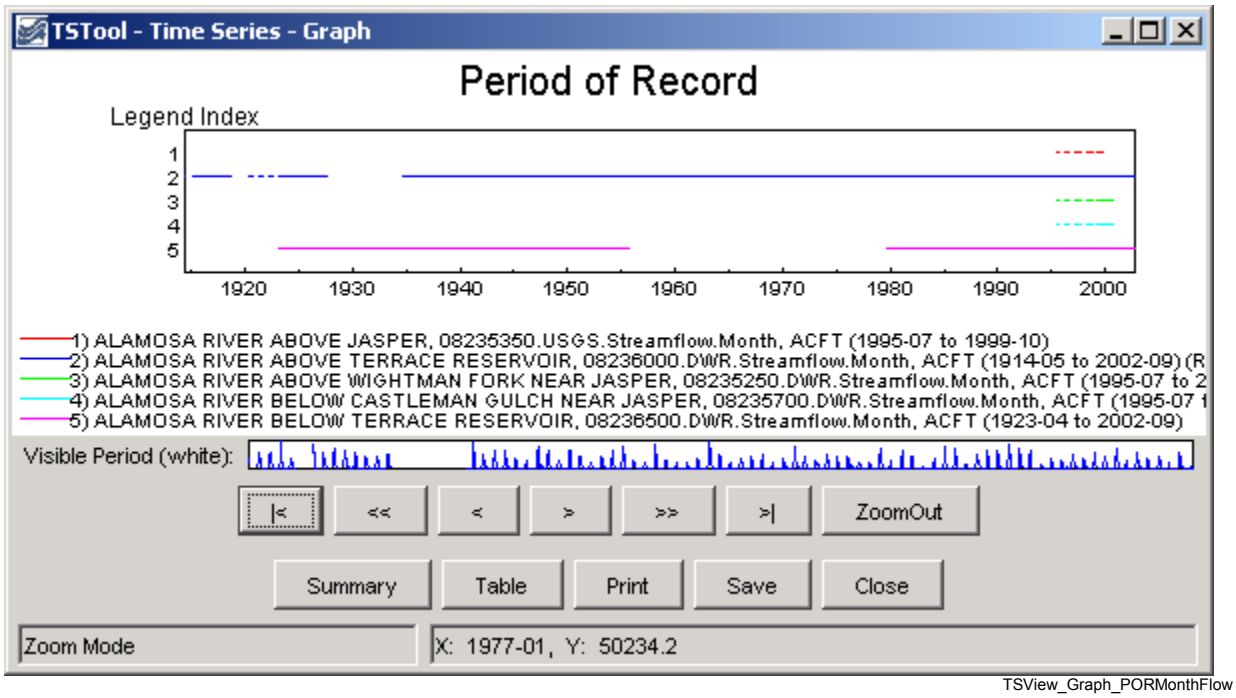
**Example Duration Graph showing Maximum Monthly Temperatures**

The algorithm for calculating and graphing a duration curve was taken from the book **Handbook of Applied Hydrology** (edited by Ven Te Chow): *"When the values of a hydrologic event are arranged in the order of their descending magnitude, the percent of time for each magnitude to be equaled or exceeded can be computed. A plotting of the magnitudes as ordinates against the corresponding percents of time as abscissas results in a so-called duration curve. If the magnitude to be plotted is the discharge of a stream, the duration curve is known as a flow-duration curve."* Features of duration graphs are as follows:

- The zoom feature is disabled for this graph type.
- Although duration curves have traditionally been applied to streamflow or reservoir data, duration graphs can be created for any time series data.
- Noticeable breaks in the curve are caused by a limited number of data points and/or values that are measured as rounded values.

### *Period of Record Graph*

The period of record graph is used to display the availability of data over a period, as shown in the following figure:




TSView_Graph_PORMonthFlow

**Example Period of Record Graph showing Monthly Streamflow**

Characteristics of the period of record graph type are:

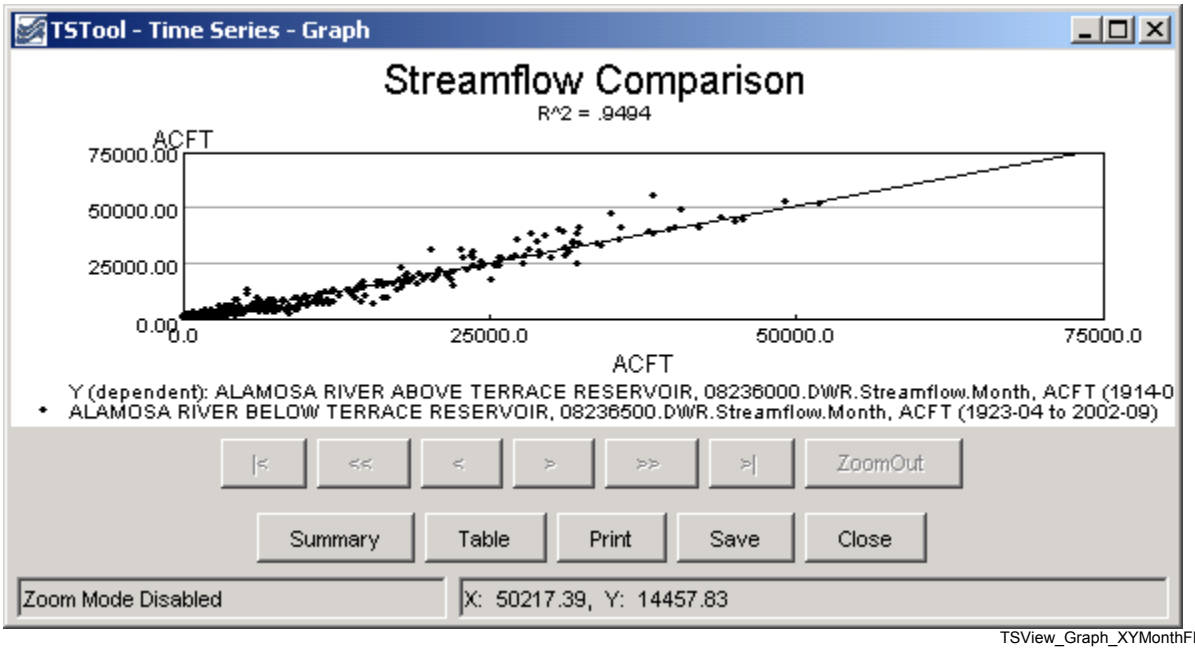
- Horizontal lines are drawn for each time series, with breaks in the line indicating missing data.
- Zooming is fully enabled, however, it may be difficult to see small breaks in the lines – it may be necessary to display symbols at the data points. The data limits properties of each time series can also be used to check for missing data. The TSTool application provides reporting features to summarize data coverage.
- Because data values are not plotted, the y-axis is labeled with a legend index number. This also allows the graph window to be compressed vertically, if desired.

### *XY-Scatter Graph*

The XY-Scatter graph type can be used to compare data having the same data interval (units can be different). This graph type is often used for the following comparisons:

1. The dependent time series (Y) requires filling and multiple independent time series (X) are analyzed to find the best time series to use as the independent time series. One or more independent time series can be plotted on the same graph.
2. The dependent time series (Y) contains observed data and one or more independent simulated time series (X) are analyzed to determine which simulation is closed to actual observations.
3. The independent (X) and dependent (Y) time series are compared to determine whether any time of relationship exists between data points. In this case, a single dependent time series may be compared with multiple independent time series on the same graph.

Currently the XY-Scatter graph can have only a single dependent time series but can have one or more independent time series. The following figure shows a typical graph.



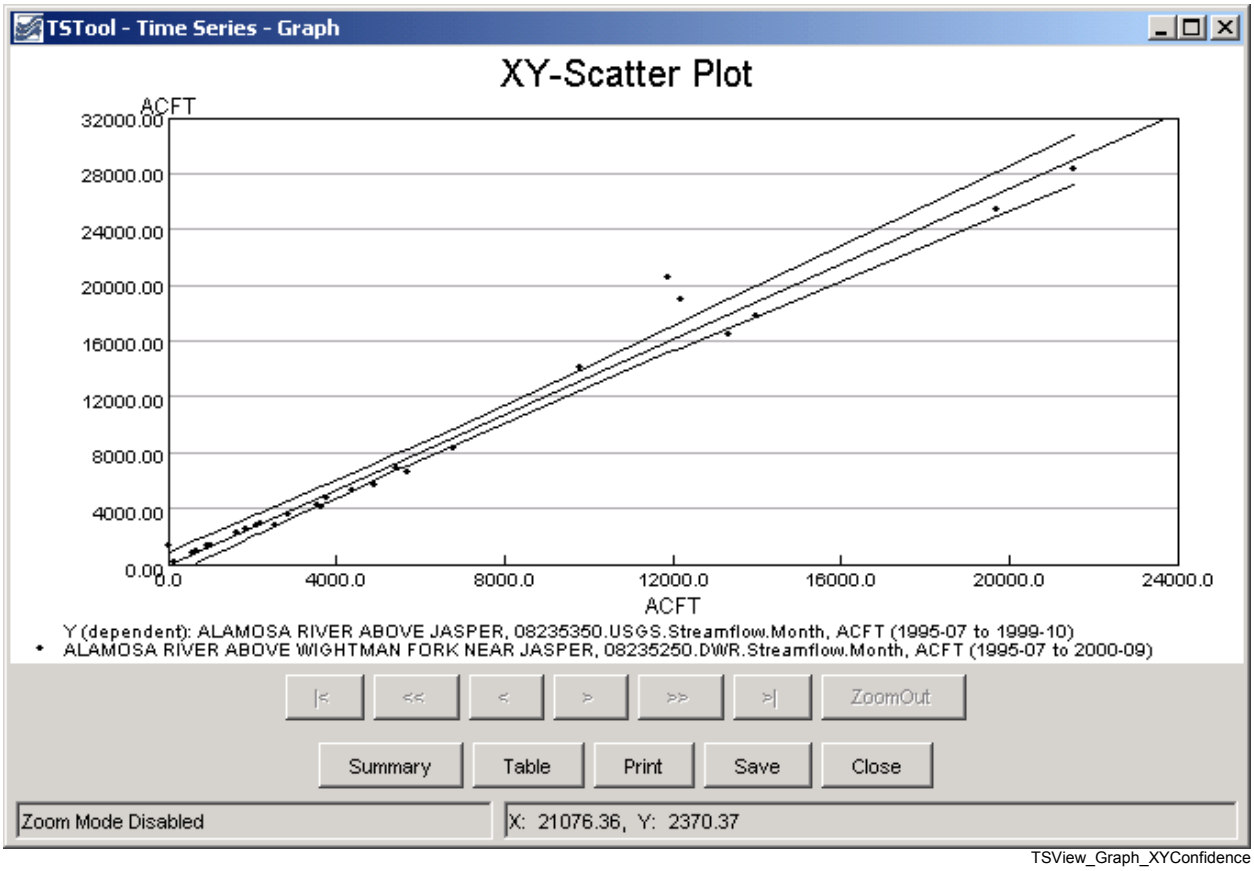

TSView_Graph_XYMonthFlow

**Example XY Scatter Graph showing Monthly Streamflow**

Characteristics of the XY Scatter graph are:

- Labels and legend are automatically generated. See the **Time Series Product Properties** section below for information about changing the appearance of the graph.
- Simple linear regression is initially performed to determine a line of best fit. See the ***Analysis*** tab in the **Time Series Product Properties** section below for information about curve fit methods.
- A 45 degree line is currently **not displayed** because time series of different types and units may be compared. Graph properties do allow the line of best fit to be forced to zero. The limits on the axes are not automatically set to equal values for the same reason; however, a property to force the values to be the same will be added.
- Zooming is disabled.
- Two or more time series must be specified and must have the same interval.

- Confidence intervals can be turned on, as shown in the following figure:




TSView_Graph_XYConfidence

The confidence intervals provide a useful way for assessing the quality of a point estimate. When a regression line is of interest, the confidence interval on the line as a whole permits one to make confidence statements about a number of values of the predictor variables simultaneously. Confidence limits for the line are a function of the level of confidence (e.g., gamma = 95% or 99%), and the F-test statistic (2, n-2 degrees of freedom, and level of significance =1-gamma). The equations used to plot the confidence intervals are shown below (note that because the curves depend on the data points, the shape and smoothness of the curves will depend on the number of points; the points are sorted to generate a continuous line).

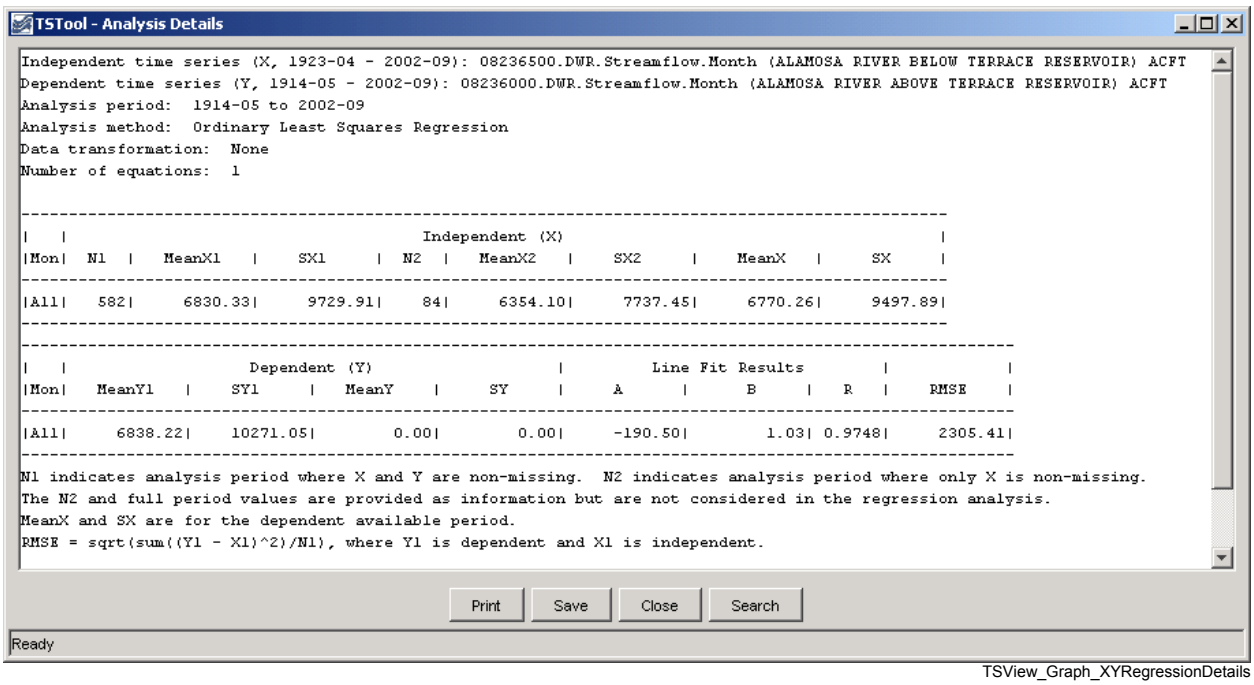
$$\hat{y}_{CI_i} = \left[\bar{y} + B\left(x_i - \bar{x}\right)\right] \pm \sqrt{2F}\left[\frac{1}{n-2}\sum_{j=1}^{n}\left(\hat{y}_j - y_j\right)^2\right]^{1/2}\left[\frac{1}{n} + \frac{\left(x_i - \bar{x}\right)^2}{\sum_{j=1}^{n}\left(x_j - \bar{x}\right)^2}\right]^{1/2}$$

where:

$\hat{y}_{CI_i}$ = confidence interval $y$ value at $x_i$
$\bar{y}$ = mean of $y$
$B$ = slope of regression line equation $y = A + Bx_i$
$x_i$ = $x$ value where $Y_{CIi}$ is being computed
$\bar{x}$ = mean of $x$

$F$ = F distribution at ($2$, $n$-$2$) degrees of freedom and gamma significance
$n$ = number of points with $x$ and $y$ values
$\hat{y}_i$ = $y$ predicted by the equation $y = A + Bx_i$
$y_i$ = $y$ value of data point corresponding to $x_i$

- The best fit line can be turned off.
- Right-clicking on the graph displays the ***Analysis Details*** menu, that, if selected, displays curve fit information about the time series, as illustrated in the following figure:



TSView_Graph_XYRegressionDetails

**Example Analysis Details**

The RMS error (or *RMSE*) is calculated in the following way:

$SSE = \Sigma(X_i - Y_i)^2$ = Sum of Square Errors
$MSE = SSE/N$ = Mean of Sum of Square Errors
$RMSE = \sqrt{MSE}$ = Square Root of the MSE

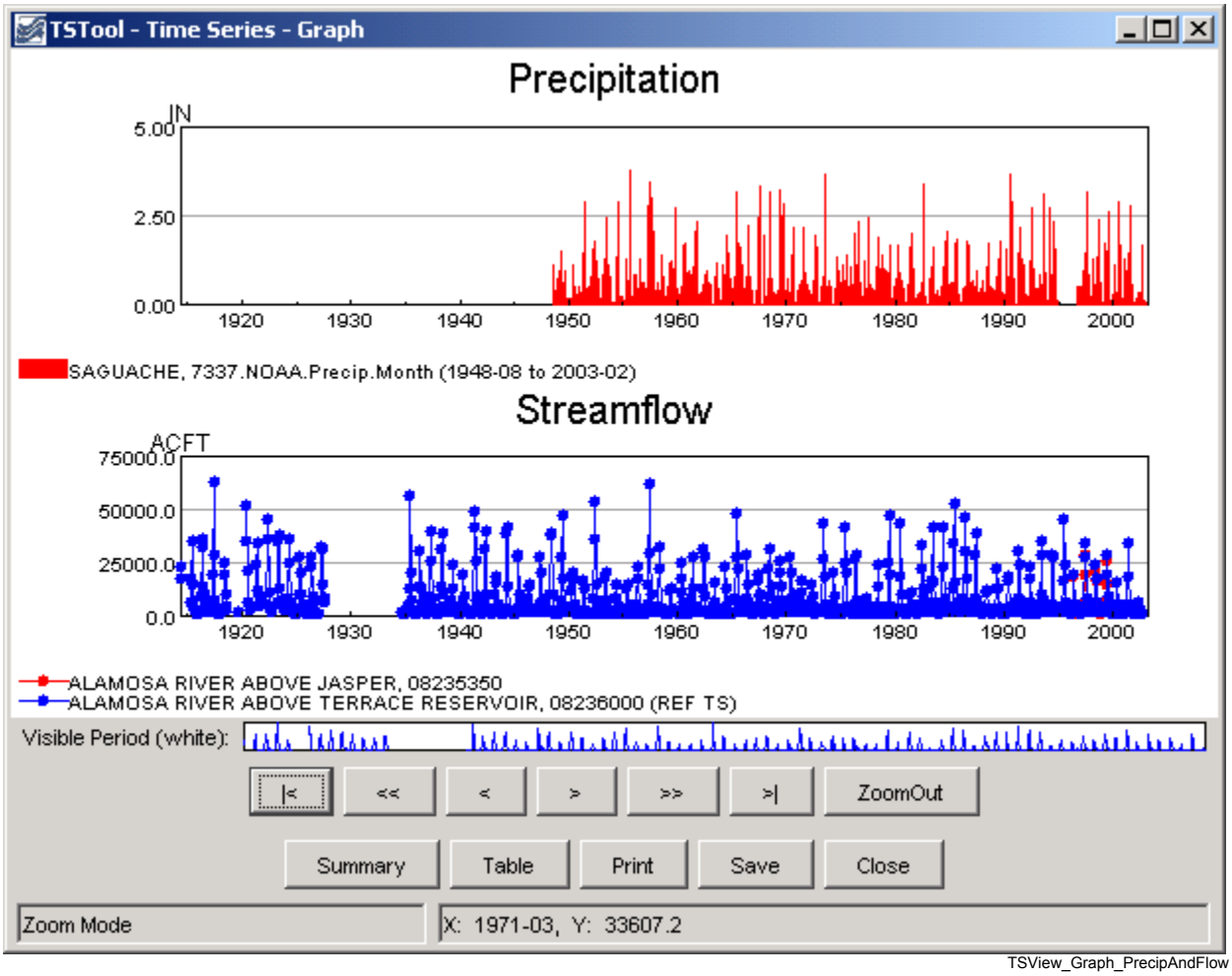
The *RMSE* can have different meanings, depending on how the data are being analyzed:

1. If a measured ($X$) and a simulated ($Y$) time series are being compared to determine, for example, to determine how well a model is simulating actual observations, then the *RMSE* indicates the error of a simulation when compared to the actual (comparing the values).
2. If two time series are evaluated to determine if the relationship between the time series can be used to estimate missing values in one of the time series, then the difference between estimated values (*Yest*) and the line of best fit (e.g., $A + BX$) is used to compute the *RMSE*. For a perfect fit, the *RMSE* would be zero. Values of *RMSE* can be used to evaluate the estimator for data filling.

To provide as much information as possible for multiple uses, the ***XY-Scatter Graph Analysis Details*** provides both *RMSE* values. The default is to display a line of best fit, which is usually desirable information. The graph properties allow the analysis to be done for data filling, if desired.

## Time Series Product Properties

A time series product is one or more time series graphs, tables, or reports on a "page", although currently TSView focuses on graph products. Time series product properties can be displayed by right-clicking on a graph of interest and selecting the ***Properties*** menu item from the popup menu. Interactively changing properties allows graphs to be configured as desired. The following figure illustrates a time series product that has two graphs (see the ***Time Series Product Reference*** section for information about how to define time series product files, which can be used to save a product).




TSView_Graph_PrecipAndFlow

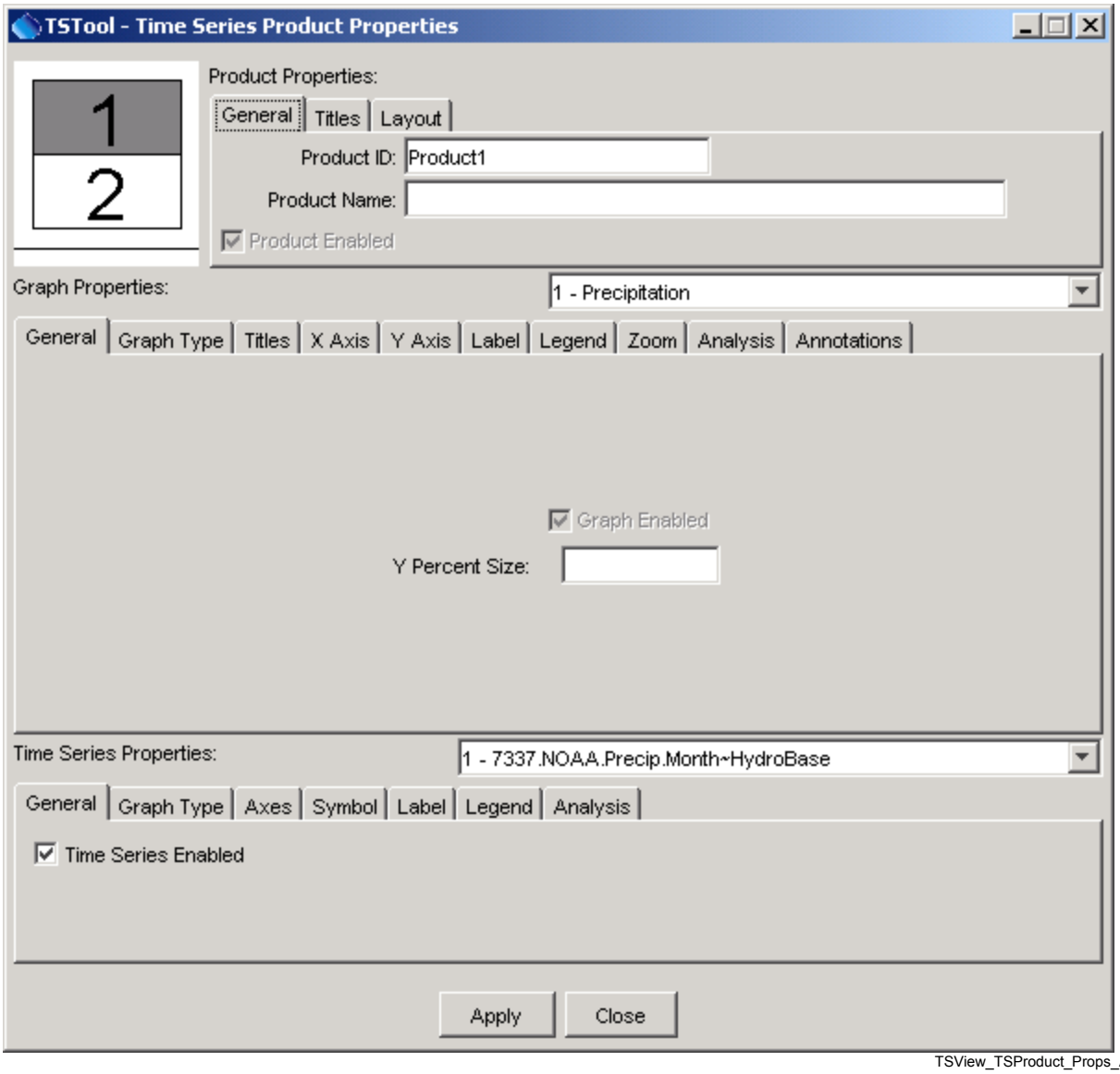
**Example Graph Product showing Precipitation and Streamflow**

In many cases, a graph product will consist of only a single graph (which may show one or more time series). However, it is also useful to display multi-graph products, especially when related data types are used. The TSView interface includes features to construct multi-graph products interactively, and the product files described in the **Time Series Product Reference** section can be created and processed. The TSTool application, for example, can interactively create or read a product file and display a graph similar to the one shown above. Important considerations for multi-graph products are:

- The product page has its own set of properties (e.g., titles and size).
- Each graph area has its own properties (e.g., titles, labels, graph type, legend). These properties comprise most of the properties for a product.

- Each time series has its own properties (e.g., symbol, color).
- If zooming is enabled, then zooming in one graph causes the same zoom to occur in related graphs. Each graph (and the reference graph) is assigned a *zoom group* number. This is used to indicate which graphs should zoom together. Currently, all graphs are in the same zoom group.

Right-clicking on a graph and pressing the ***Properties*** item in the popup menu will display the properties for the graph. The following figures illustrate the properties tabbed panel:




TSView_TSProduct_Props_All

**Tabbed Panel to Edit Time Series Product Properties**

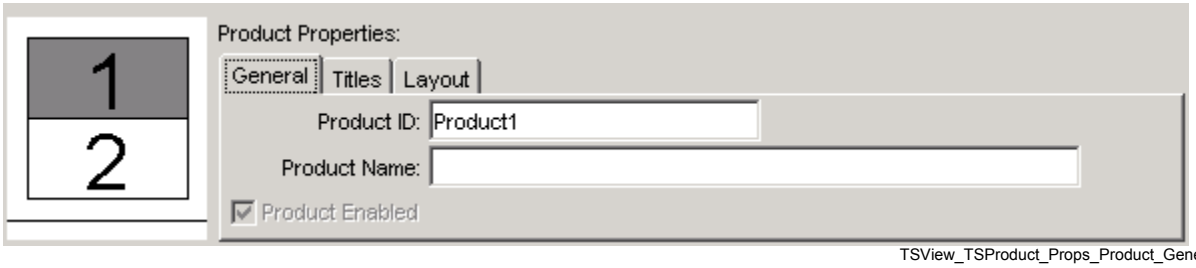
The time series product properties display as three layers of tabbed panels. Characteristics of the properties window are:

- The window is divided into a layout area (top-left) and tabs for different groups of properties. The layout window shows the overall layout of graphs on a page and allows manipulation of the time series product by dropping time series onto the layout.

- The top layer of tabs (***Product Properties***) are associated with product properties (the page).
- The middle layer of tabs (***Graph Properties***) are associated with subproduct properties (graphs on the page). The graph of interest is selected using the drop-down choice that shows the graph number and graph main title. When initially displayed, the selected graph is the one that was clicked on to display the ***Properties*** menu.
- The bottom layer of tabs (***Time Series Properties***) are associated with data (time series) properties. A time series within a graph is selected using the drop-down choice that shows the time series number within the graph, and the time series identifier. When initially displayed, the first time series for the selected graph is selected.
- The ***Apply*** button will apply the current properties and update the graph(s). **Warning - when changing between graphs and time series (where multiple graphs and/or time series exist for a product), properties that are changed are applied automatically. This behavior is being evaluated.**
- The ***Close*** button will apply the current properties, update the graph(s), and close the properties window.
- Only properties read from an original time series product file or that are set by the user will be saved if a time series product is saved. Internal defaults are not saved. This minimizes the size and complexity of product definition files.

The remaining discussion in this section illustrates each of the tabbed panels. The text-based properties that are displayed in the panels are described in the **Time Series Product Reference** section.

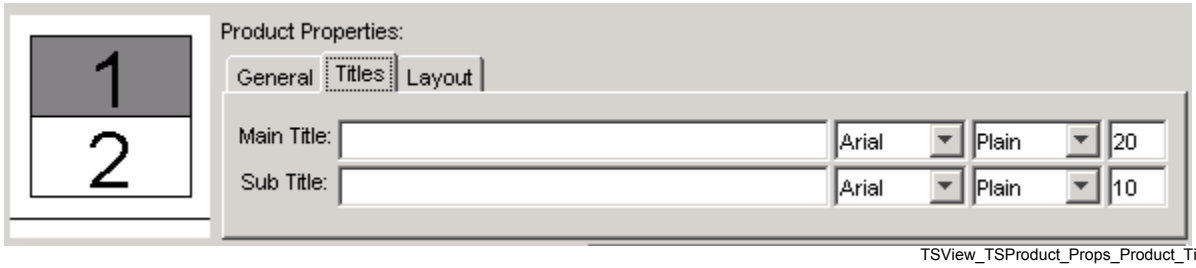
***Product Properties - General***




**Example Product General Properties**

The above figure illustrates the product ***General*** properties. The ***Product Enabled*** checkbox indicates whether the product is enabled (currently view-only). The ***Product ID*** is used when saving the product definition to a database. The ***Product Name*** is also used to when displaying lists of products.

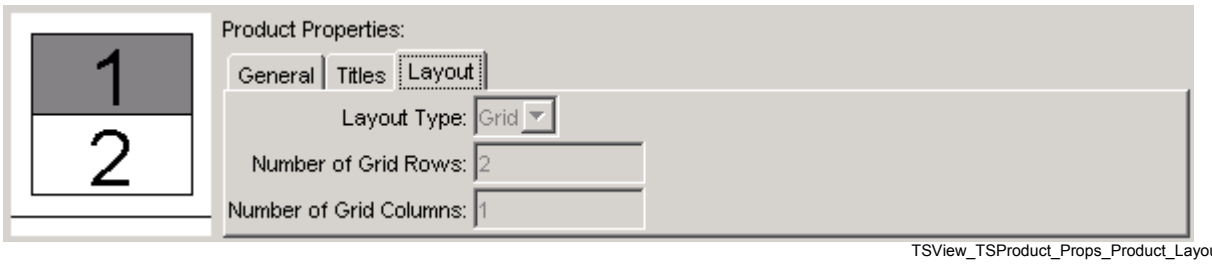
***Product Properties - Titles***




**Example Product Title Properties**

Product ***Titles*** properties include title and subtitle. If blank, no title will be shown. Because graphs (subproducts) also have a title and subtitle, the product titles are often only used when multiple graphs are included on a page.

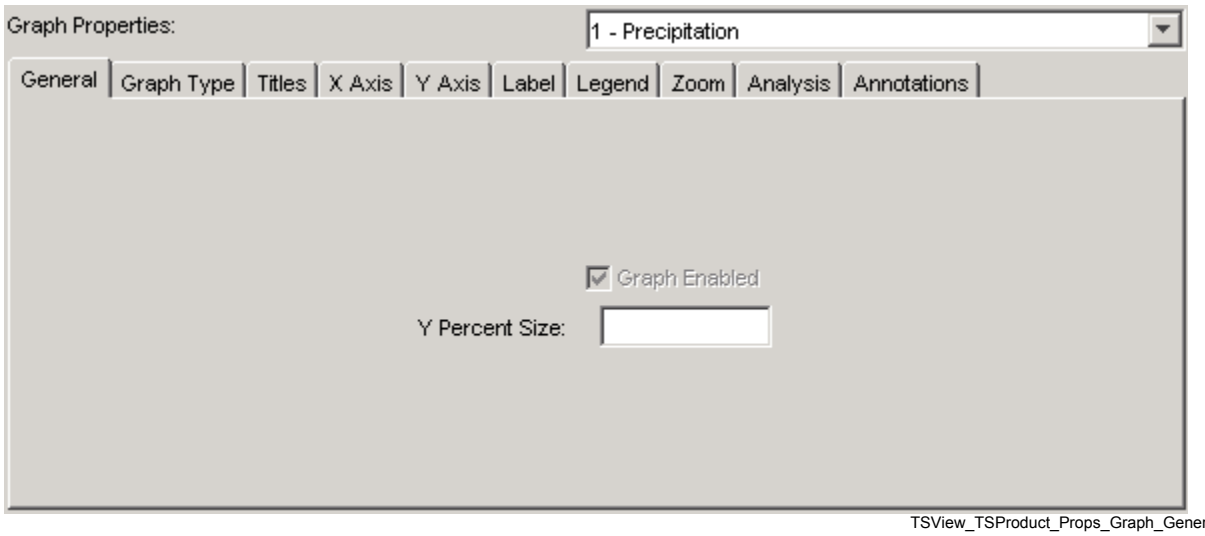
***Product Properties - Layout***




**Example Product Layout Properties**

Product ***Layout*** properties describe how graphs are laid out on the page. Currently, graphs can only be organized in a vertical stack, although the design will support multiple columns. The layout properties are updated automatically as graphs are added to or deleted from the layout window at the left. The relative size of each graph on the page is controlled by using the `LayoutYPercent` general property for each graph on the page (see below).

***Graph Properties - General***

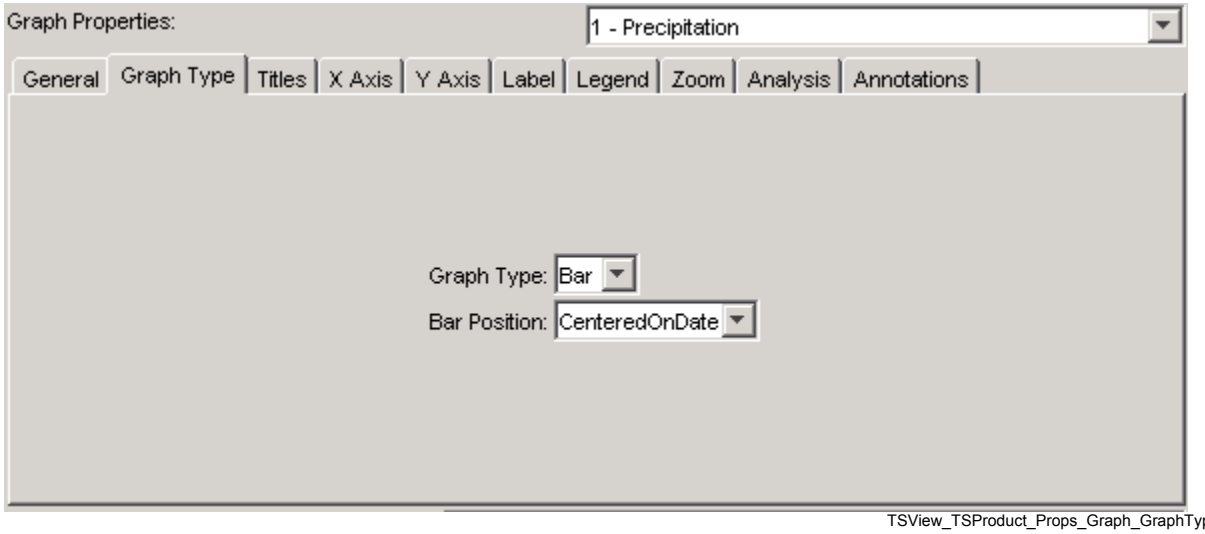



TSView_TSProduct_Props_Graph_General

**Example Graph General Properties**

The above figure illustrates graph ***General*** properties. The ***Graph Enabled*** checkbox indicates whether the graph is enabled (currently view-only). The vertical size of the graph on the page (percent) can also be specified (the default is to size all the graphs on the page equally).

***Graph Properties - Graph Type***

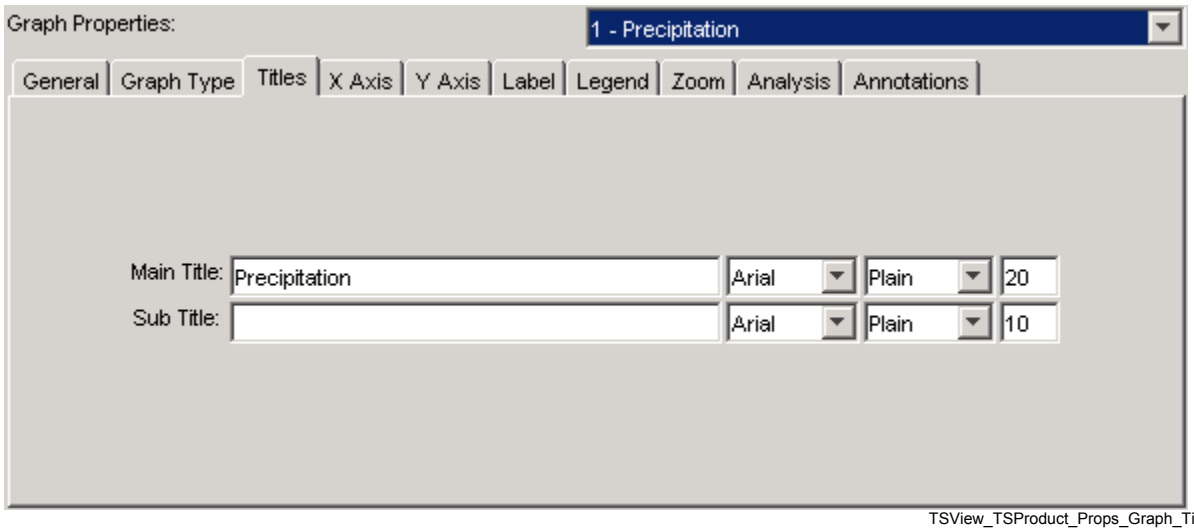



TSView_TSProduct_Props_Graph_GraphType

**Example Graph Graph Type Properties**

***Graph Type*** properties control the overall display of the data. The graph type can be changed after the initial display only when switching between simple graph types (e.g., line and bar graphs). Some graph types may have specific properties (e.g., bar width for bar graphs). If necessary, to change the graph type, you can usually select the type from a main application, and generate a new graph.

***Graph Properties - Titles***

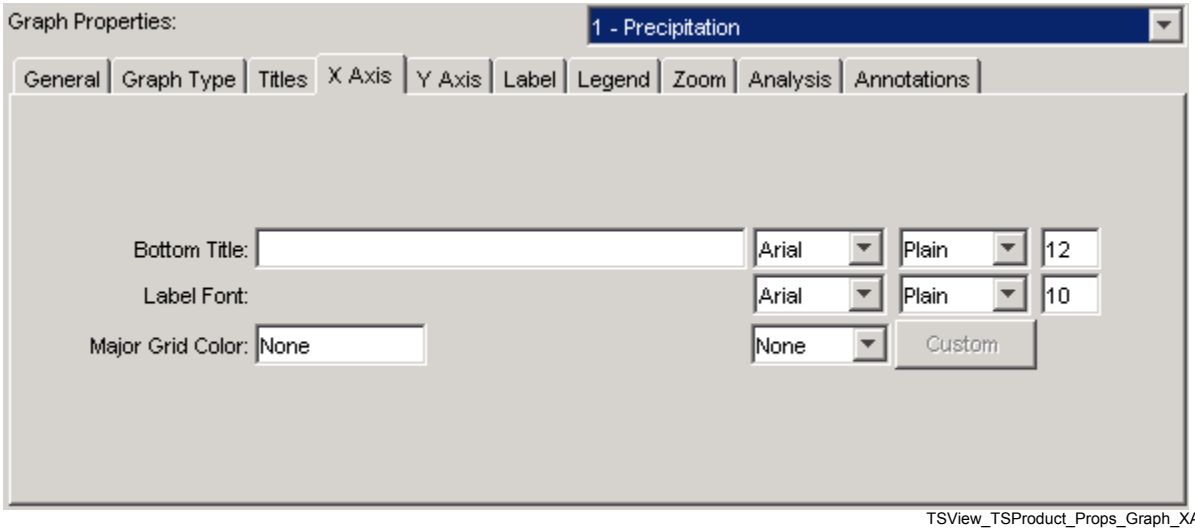



TSView_TSProduct_Props_Graph_Titles

**Example Graph Title Properties**

Graph ***Titles*** properties include title and subtitle. If blank, no title will be shown. Font properties can also be specified. After applying the a change to the main title, the title will be added in the list of graphs.

***Graph Properties - X Axis***

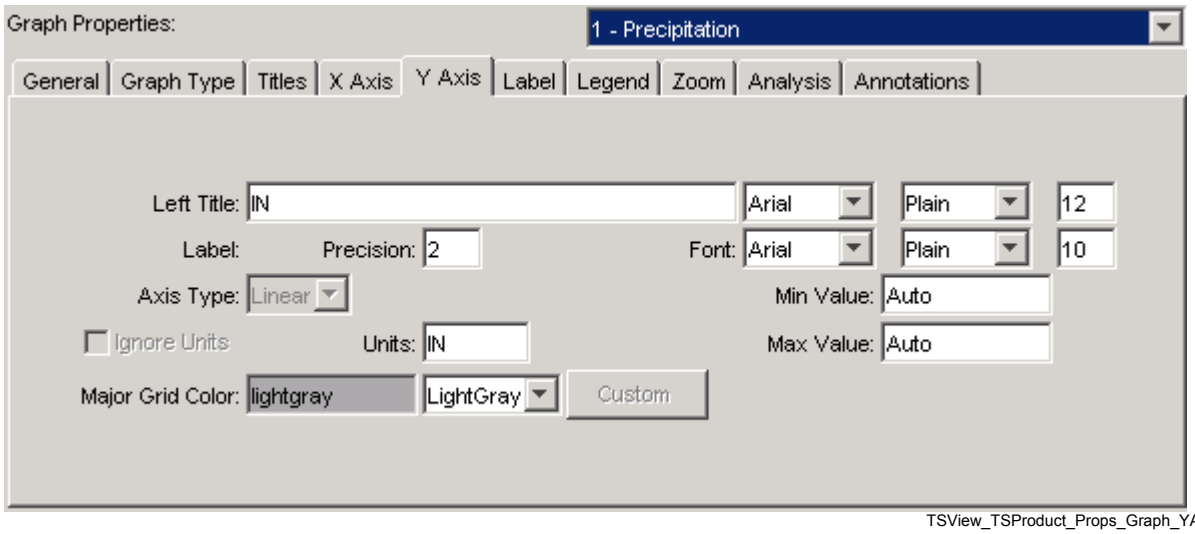



TSView_TSProduct_Props_Graph_XAxis

**Example Graph X Axis Properties**

Graph ***X Axis*** properties include title, label, and grid properties. The ***Major Grid Color*** can be specified by selecting from the available choices, which then fill in the text field with the given color selection.

***Graph Properties - Y Axis***



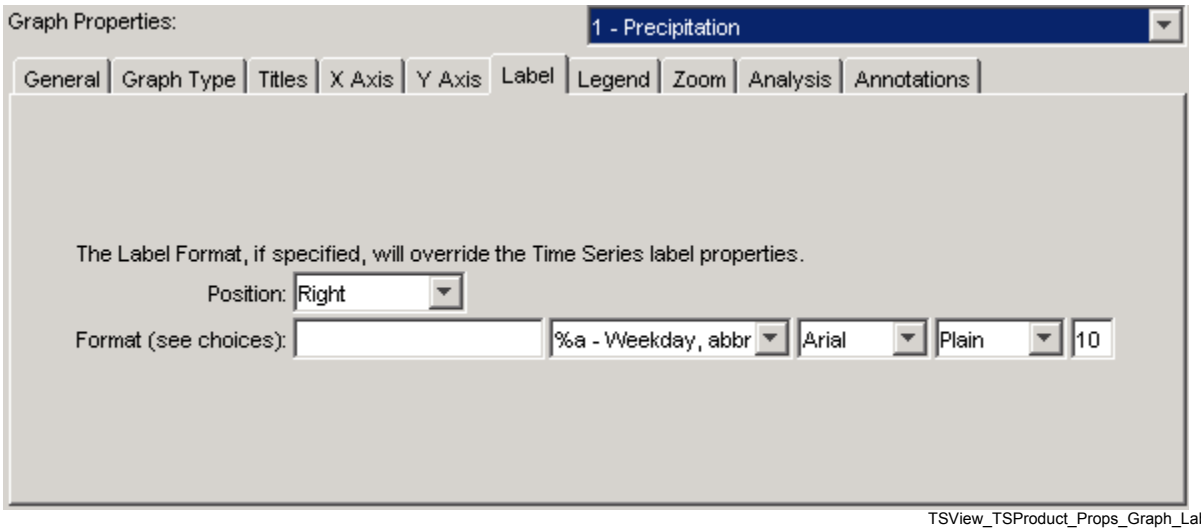

TSView_TSProduct_Props_Graph_YAxis

**Example Graph Y Axis Properties**

Graph ***Y Axis*** properties include the following:

- ***Left Title*** - this may be set to the data units but can be specified (the Y axis title is currently always placed at the top of the Y axis).
- ***Label*** - the font for labels and precision of numerical labels can be specified.
- ***Axis Type*** - currently this is view-only.
- ***Min Value***, ***Max Value*** - currently this is view-only but can be set in time series product definition files (see the **Time Series Product Reference** section).
- ***Units***, ***Ignore Units*** - currently these are view-only. If time series with incompatible units are graphed, ***Ignore Units*** will be checked and the units may be shown in the legend.

***Graph Properties - Label***

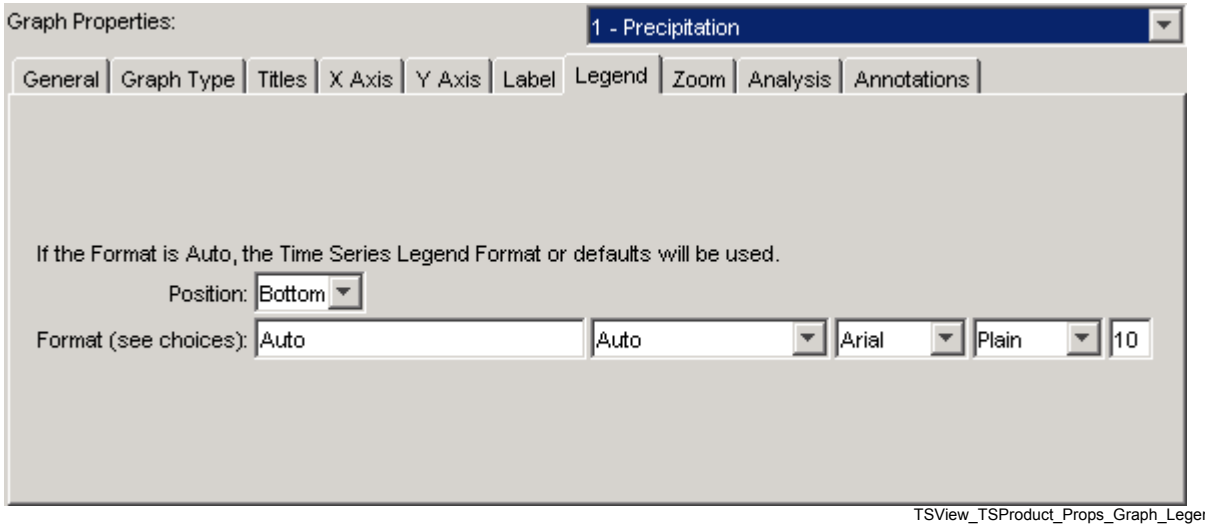



TSView_TSProduct_Props_Graph_Label

**Example Label Properties**

Data points are not labeled by default because there are usually too many data labels to be legible. However, for plots with limited data, or after zooming in, labels can be useful to identify points without referring to tabular data. The label format can be defined using the choices next to the text field or by entering literal text. For an XY Scatter plot, repeat the `%v` format (e.g., `%v, %v`) to show the independent (X) and dependent (Y) data values. See the `DataLabel` properties in the **Time Series Product Reference** section for label options.

***Graph Properties - Legend***

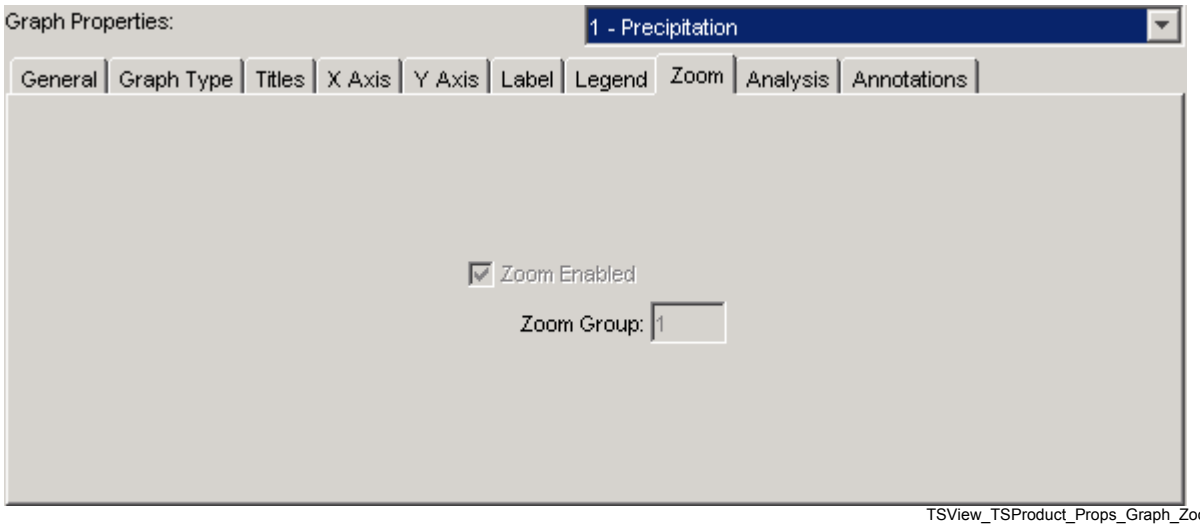



TSView_TSProduct_Props_Graph_Legend

**Example Graph Legend Properties**

Graph ***Legend*** properties include format and font properties. If the ***Legend Format*** is `Auto`, a default legend format will be constructed from the time series description, identifier, and period of record. See the `LegendFormat` property in the **Time Series Product Reference** section for legend formatting options.

***Graph Properties - Zoom***

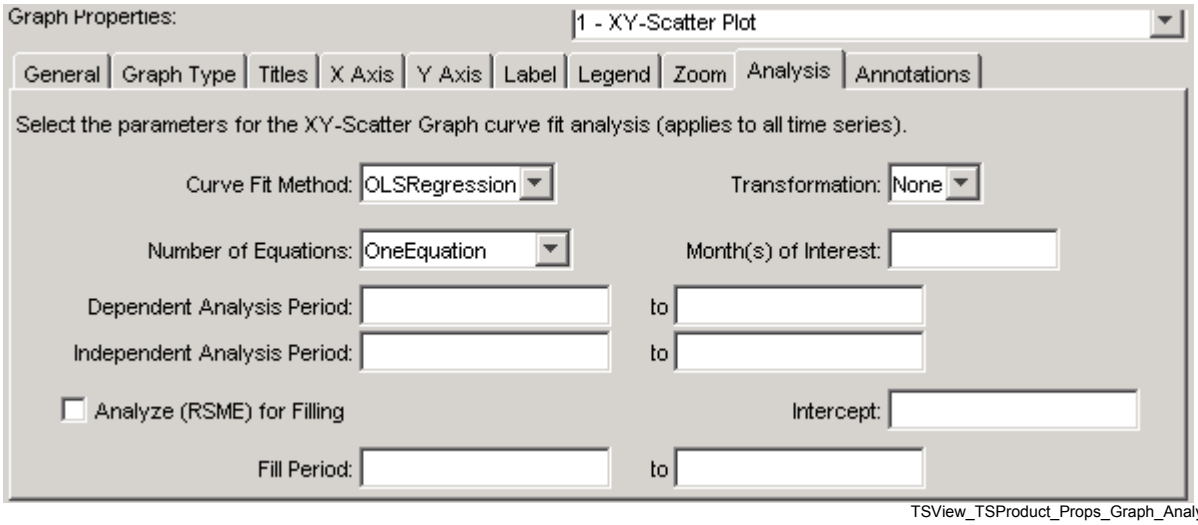



TSView_TSProduct_Props_Graph_Zoom

**Example Graph Zoom Properties**

Graph ***Zoom*** properties are currently view-only. Zoom will be enabled for graph types that support it (e.g., duration graphs do not). The zoom group indicates how graphs should respond when other related graphs on a page are zoomed and currently defaults to `1` for all graphs.

***Graph Properties - Analysis***

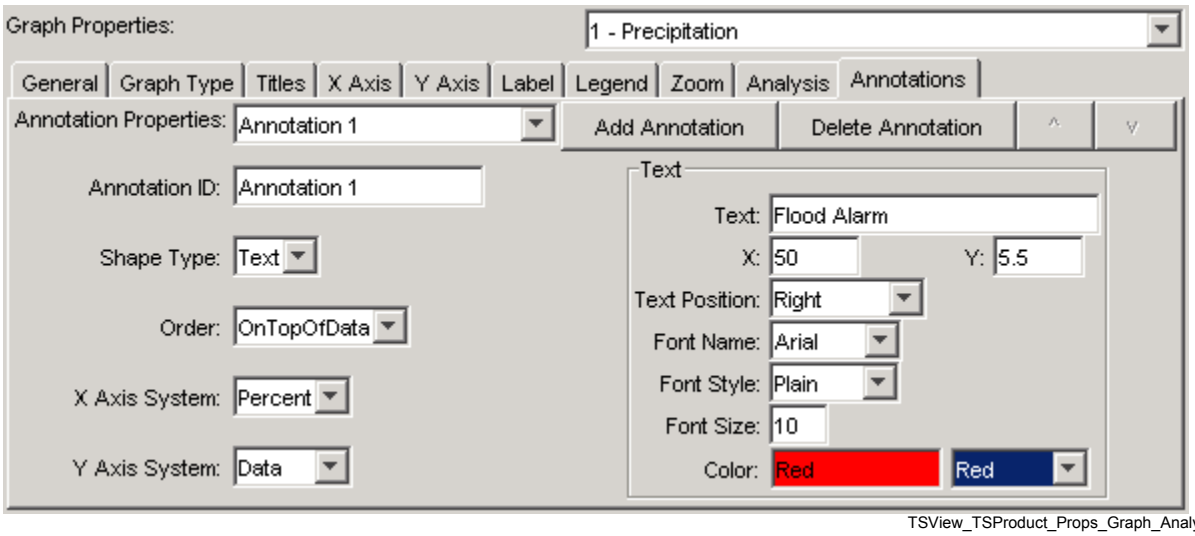



TSView_TSProduct_Props_Graph_Analysis

**Example Graph Analysis Properties**

Graph ***Analysis*** properties are available if the graph requires some type of analysis to produce the result (e.g., curve fitting). See also the analysis tab for individual time series. For help with input, place the mouse cursor over a field and a tool tip will be shown.

***Graph Properties - Annotations***



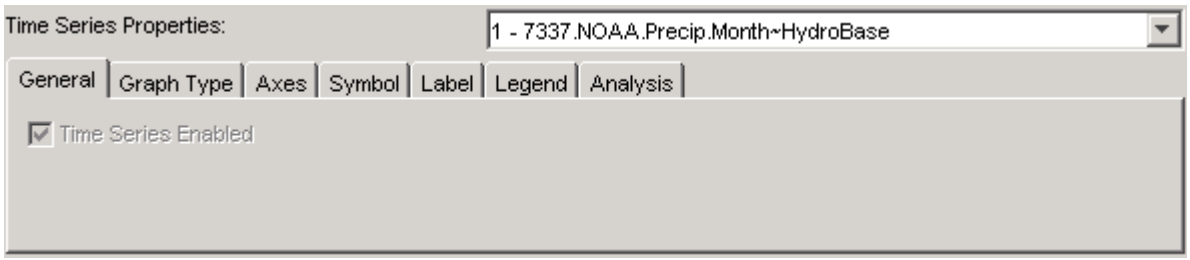

TSView_TSProduct_Props_Graph_Analysis

**Example Graph Analysis Properties**

Graph ***Annotations*** properties are used to add annotation objects to a graph. Annotations are text, line, or other simple shapes and are stored as simple text properties in time series products (see the **Time Series Product Reference** section below for more information). Annotations are placed on a graph using data units or a percent of the graph dimension. This allows annotations to move if a graph uses real-time data.

To add an annotation, press the ***Add Annotation*** button. Then select the ***Shape Type*** and specify annotation properties, as appropriate. The example shown in the above figure places the string "Flood Alarm" at the horizontal (X) center of the graph at a Y-coordinate of 5.5. A horizontal annotation line could also be drawn using 0 to 100 percent on the X axis at the same Y-coordinate.

***Time Series Properties - General***

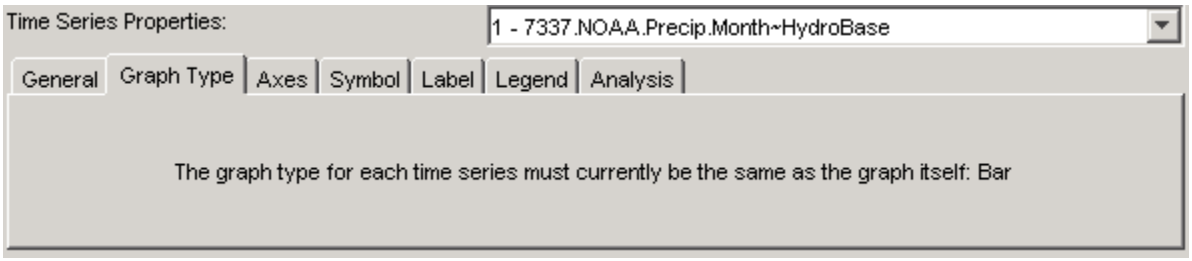



TSView_TSProduct_Props_TS_Gemeral

**Example Time Series General Properties**

Time series ***General*** properties are currently view-only and indicate whether the time series is enabled for the graph.

***Time Series Properties - Graph Type***

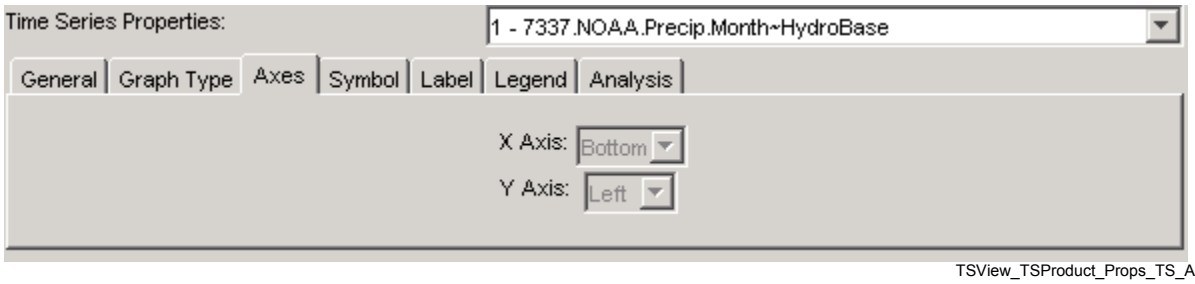



TSView_TSProduct_Props_TS_GraphType

**Example Time Series Graph Type Properties**

Time series ***Graph Type*** properties are currently disabled. Currently all time series in a graph must have the same graph type.

***Time Series Properties - Axes***

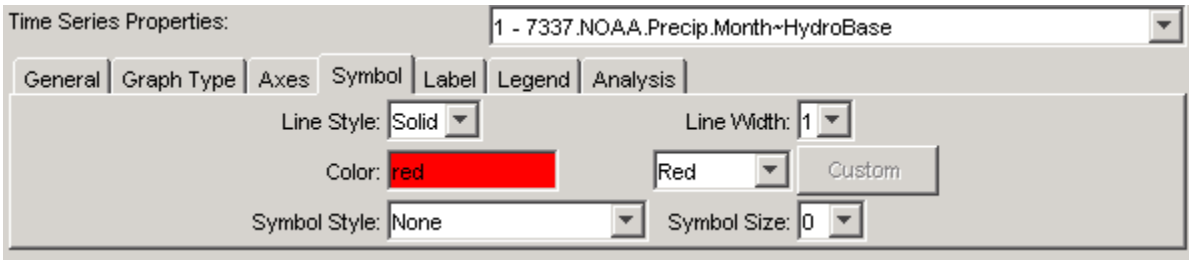



TSView_TSProduct_Props_TS_Axes

**Example Time Series Axes Properties**

Time series ***Axes*** properties are currently view-only and show the graph axes to which a time series is associated.

***Time Series Properties - Symbol***

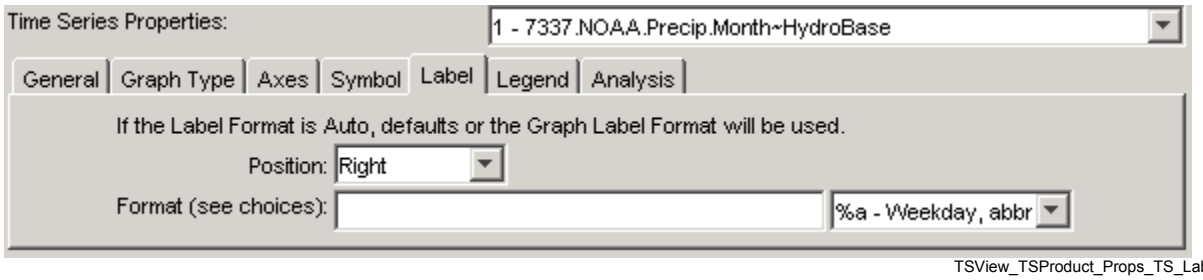



TSView_TSProduct_Props_TS_Symbol

**Example Time Series Symbol Properties**

Time series ***Symbol*** properties define the graphical appearance of time series data. Properties are enabled/disabled based on the graph type (e.g., the ***Symbol Style*** will be disabled if the graph type is `Bar`). The symbol properties are consistent with the GeoView tools used for maps.

***Time Series Properties - Label***

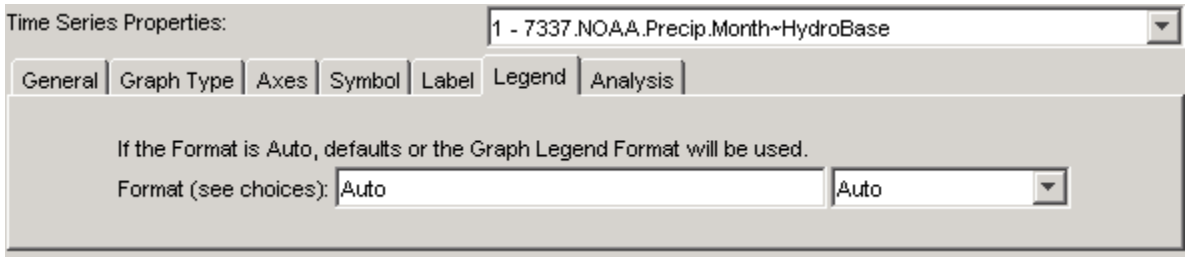



TSView_TSProduct_Props_TS_Label

**Example Time Series Label Properties**

Time series ***Label*** properties allow the data label to be changed. Data points are not labeled by default because there are usually too many data labels to be legible. However, for plots with limited data, or after zooming in, labels can be useful to identify points without referring to tabular data. The label format can be defined using the choices next to the text field or by entering literal text. For an XY Scatter plot, repeat the `%v` format to show the independent (X) and dependent (Y) data values. See the `DataLabel` properties in the **Time Series Product Reference** section for label options.

***Time Series Properties - Legend***



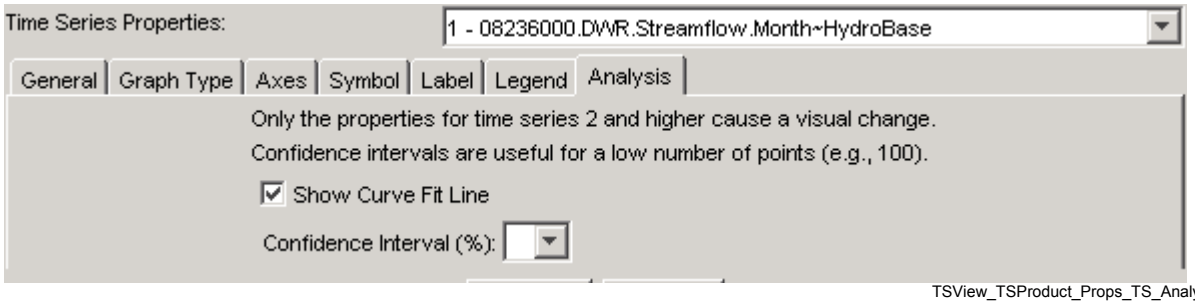

TSView_TSProduct_Props_TS_Legend

**Example Time Series Legend Properties**

Time series ***Legend*** properties allow the legend format to be changed. This is useful if the time series is to have different legend labeling that the other time series in the graph. If the ***Legend Format*** is `Auto`, a default legend format will be constructed from the time series description, identifier, and period of record.

See the `LegendFormat` property in the **Time Series Product Reference** section for legend formatting options.

***Time Series Properties - Analysis***



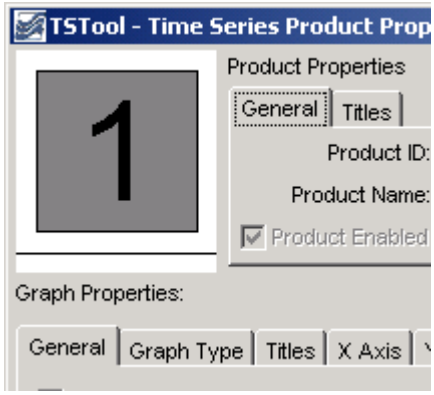

TSView_TSProduct_Props_TS_Analysis

**Example Graph Analysis Properties**

Time Series ***Analysis*** properties are available if the graph requires some type of analysis to produce the result (e.g., curve fitting).

## Changing a Graph Page Layout

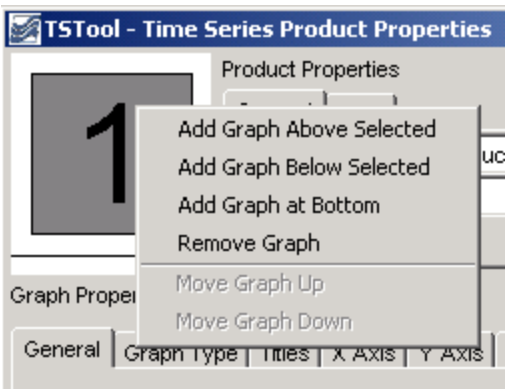
The default page layout for graphs is to display all time series in one graph. In this configuration, the layout area at the top-left corner of the time series product window will display as shown below:




TSView_Layout_1Graph

**Layout Window Showing One Graph**

The layout area can be used to split the single graph into multiple graphs on the page. For example, two graphs may be needed because of different units, time step, or graph type. Left-clicking on a graph in the layout area will select the graph – the selected graph is shown in gray. Right-clicking on the layout area displays a menu with available options:




TSView_Layout_Menu

**Layout Window Menu**

The actions taken by the menus are described below:

| | |
|---|---|
| ***Add Graph Above Selected*** | Add a new graph above the selected graph, renumbering the graphs as needed. |
| ***Add Graph*** | Add a new graph below the selected graph, renumbering the graphs as needed. |
| ***Add Graph at Bottom*** | Add a new graph below all existing graphs, giving the new graph the next number in the sequence. |
| ***Remove Graph*** | Remove the selected graph, renumbering the graphs as needed. |
| ***Move Graph Up*** | Move the graph up one in the sequence, renumbering the graphs as needed. The menu is enabled only when multiple graphs are available. |
| ***Move Graph Down*** | Move the graph down one in the sequence, renumbering the graphs as needed. The menu is enabled only when multiple graphs are available. |

When a new graph is added, it will not have any specific properties, time series data, or annotations, other than the default properties that are assigned (e.g., the default graph type is `Line`), and when drawn it will appear as a blank area. To see the graph, it will be necessary to set the graph's properties and provide it with data (and optionally, annotations). Properties and annotations are defined using the properties tabs as documented in previous sections – use the ***Apply*** button to apply and view the changes. To set graph properties, the graph to be modified should be selected from the choices at the top of the ***Graph Properties*** tab panel (or by selecting the graph in the layout window).

To add time series data to the new graph (or an existing graph), two approaches can be taken:

1. Find the time series to be moved using the list in the time series properties panel. It may be necessary to select a graph to find the time series – selecting a graph will not impact the ability to move the time series to a different graph. In the list of time series, hold the left mouse button down over a time series choice and drag the time series to a graph on the layout area. During this process, the cursor will change to a new shape, as shown below:

TSView_Layout_MoveTS

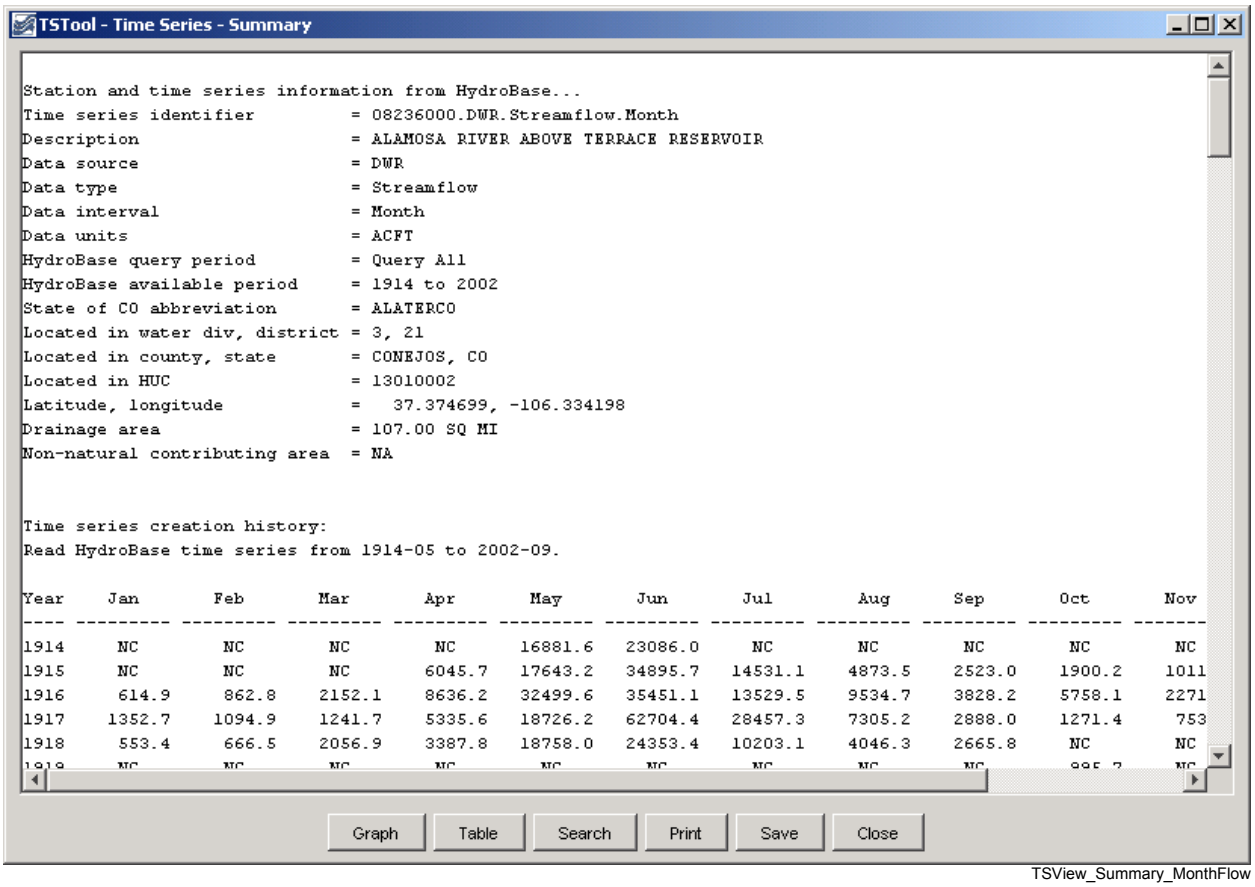
Release the mouse over the graph in the layout area that is to receive the time series. The time series will then be removed from the original graph and will be inserted into the new graph as the last time series in the list.

2. Some software programs will allow dragging a time series from a display to the time series product properties window. Similar to above, drag the time series onto the receiving graph in the layout area. Refer to documentation for the specific software program for additional information about whether this feature is available.

After adding a new graph and moving time series, it may be necessary to change the graph type for a graph. For example, the top graph may show precipitation and the bottom graph may show streamflow resulting from the precipitation. Precipitation is normally shown as bars and streamflow as a line. The graph will initially be shown using the graph type that was originally selected. Change the graph type in the new configuration, as appropriate, by selecting the graph to be changed and then use the ***Graph Type*** tab.

## Time Series Summary View

The time series summary view can be selected from the graph or table view using the ***Summary*** button. Additionally, applications that use the TSView package may allow displaying a summary from a menu or button option. A time series summary view can usually be produced quickly, whereas the table view uses more resources. The following figure illustrates a typical summary view.



TSView_Summary_MonthFlow

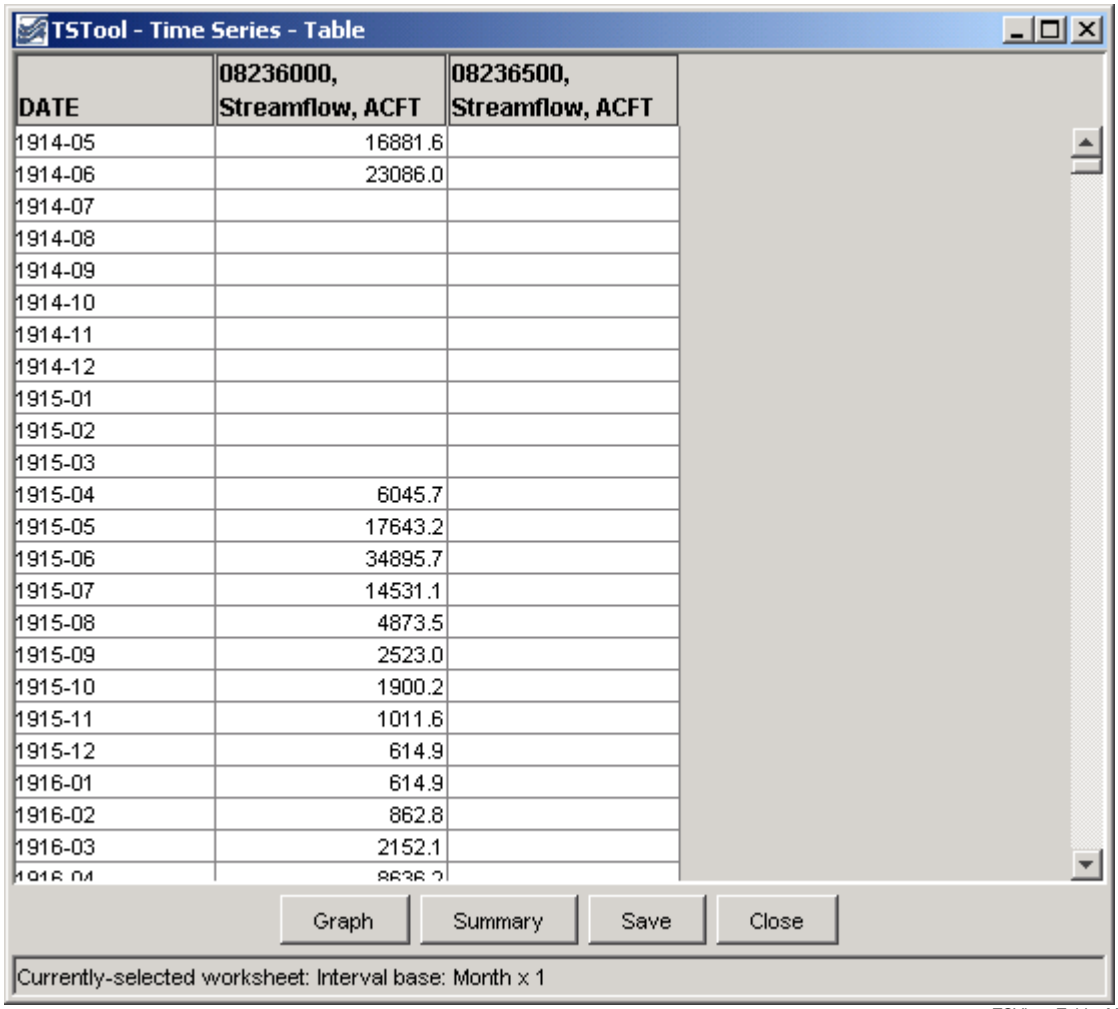
**Example Summary View showing Monthly Streamflow**

The summary view has the following characteristics:

- The graph view can be displayed using the ***Graph*** button and the table view can be displayed using the ***Table*** button.
- Each time series interval (e.g., `Month`, `Day`, `Hour`) has a default summary report format suitable for the interval. This format may be made more specific if time series are read from specific data types (e.g., if daily diversion time series are read from the HydroBase input type, the summary report will use the State of Colorado diversion coding report format).
- The contents of the view can be printed.
- The summary can be saved as a text file or DateValue time series file.
- Limited search capabilities are available to search for a string in the text area.

## Time Series Table View

The time series table view can be selected from the graph or summary view using the ***Table*** button. Additionally, applications that use the TSView package may allow displaying a table from a menu or button option. The table view is useful for viewing date and data values in a spreadsheet-like display. A time series table view for a long period or many time series may require extra time to display, but usually only a few seconds are required. The following figure illustrates a typical table view.




TSView_Table_MonthFlow

**Example Table View showing Monthly Streamflow**

Characteristics of the table view are:

- The summary view can be displayed using the ***Summary*** button and the graph view can be displayed using the ***Graph*** button.
- The precision of dates matches the data interval for the time series.
- If time series with different intervals are selected, multiple tables will be displayed in the window.
- The table contents can be saved as a DateValue file, which is a useful delimited format file.

## Time Series Product Reference

A *time series product* is a report, table, or graph, although currently TSView focuses on graph products. Examples of time series products and their use are:

- Reports and graphs generated from a database to perform quality checks.
- Reports and graphs generated from model input and output to check a calibration or model results.
- Reports and graphs generated from a database for real-time data products, to monitor current conditions or to create products for a web site.

The TSView package contains features to process time series product files in interactive and batch mode to produce time series products. Currently, only graph products are supported. The time series graph view allows a graph to be saved as a time series product file. This file describes the layout and contents of the product but does not include the time series data itself; therefore, the time series product is relatively small.

### *Time Series Product File Format*

The time series product definition file format consists of comments (lines that start with #), sections (indicated by `[ ]`), and simple `property=value` pairs. The following example illustrates the parts of a product file:

```
# Example Time Series Product file
# Comments start with #
# Sections are enclosed in [] and must be included

[Product]

# product properties - surround with double quotes if values contain spaces
xxxxx="xxxxxx   xx"

[SubProduct 1]

# "sub-product", e.g., a graph on a page (page is product and may have
# multiple graphs)

[Data 1.1]

# First data item in the SubProduct (e.g., first time series).
TSID = ...

[Data 1.2]

# Second data item in the SubProduct (e.g., first time series).
TSID = ...

[SubProduct 2]

[Data 2.1]

# Annotations are associated with a SubProduct
[Annotation 2.1]

Annotation properties…
... etc. ...
```

**Example Time Series Product File**

Most properties, if not specified in the file, will default to reasonable values. The most important property is TSID, which indicates time series identifier to be read for data. The time series identifier follows the conventions described in the **Time Series Terminology** section. Some tools, like TSTool, will match the TSID against time series that have already been read into memory, or, if necessary, read the time series from a file or database if not in memory. The normal convention is to use a *.tsp* extension for time series product file names.

The list of properties that can be used in a time series product definition file is quite extensive and new properties are added as new features are enabled. As shown in the previous section, properties are defined as simple `variable=value` pairs. These properties are used internally by the graph view (and its properties window) regardless of whether the graph originated from a product file or interactively. The following tables list the properties that are currently supported or envisioned to be enabled in the future. The first set of properties are used to define the overall product (the full page).

**Top-level Time Series Product Properties**

| Product Property | Description | Default |
|---|---|---|
| `Current DateTime` | The current date and time to be drawn as a vertical line on all graphs. If the property is not specified, no current date/time line will be drawn. If specified as `Auto`, the current system time will be used for the date/time. If specified as a valid date/time string (e.g., `2002-02-05 15`), the string will be parsed to obtain the date/time. **This property is often specified internally by the application at run time.** | Not drawn. |
| `Current DateTime Color` | Color to use to draw the current date and time. Colors can be specified as named colors (e.g., `red`), hexadecimal RGB values (e.g., `0xFF0000`), integer triplets (e.g., `255,0,0`) or floating point triplets (e.g., `1.0,0.0,0.0`). | `Green` |
| `Enabled` | Indicates whether the product should be processed. Specify as `True` or `False`. | `True` |
| `LayoutNumber OfColumns` | The number of columns in the product. | Currently always `1`. |
| `LayoutNumber OfRows` | The number of rows in the product. | Currently equal to the number of graphs. |
| `LayoutType` | Indicates how the graphs in a product are laid out. Only `Grid` is supported. | `Grid` |
| `MainTitle FontName` | Name of font to use for main title (`Arial`, `Courier`, `Helvetica`, `TimesRoman`). | `Arial` |
| `MainTitle FontSize` | Size, in points, for main title. | `20` |
| `MainTitle FontStyle` | Font style (`Bold`, `BoldItalic`, `Plain`, `PlainItalic`). | `Plain` |
| `MainTitle String` | Main title for the product, centered at the top of the page. | No main title. |
| `OutputFile` | Output file when graph product is generated in batch mode. **This property is often set at run time by the application.** | *C:\TEMP\tmp.jpg* on windows, */tmp/tmp.jpg* on UNIX |
| `Owner` | An identifier that indicates the owner of the TSProduct, used internally when saving TSProduct definitions to a database that implements permissions. | None – can be blank if permissions are not important. |

**Top-level Time Series Product Properties (continued)**

| Product Property | Description | Default |
|---|---|---|
| PeriodEnd | Ending date for time series data in the product. The date should be formatted according to common conventions (e.g., YYYY-MM-DD HH:mm), and should ideally be of appropriate precision for the data being queried. **This property is often set at run time by the application.** | Full period is read. |
| PeriodStart | Starting date for time series data in the product. The date should be formatted according to common conventions (e.g., YYYY-MM-DD HH:mm), and should ideally be of appropriate precision for the data being queried. **This property is often set at run time by the application.** | Full period is read. |
| PreviewOutput | Indicates whether the product should be visually previewed before output. **This property is often set at run time by the application and is used to override generation of the OutputFile.** | false |
| ProductType | Currently only Graph is supported. | Graph |
| SubTitleFontName | Name of font to use for subtitle (see MainTitleFontName for font list). | Arial |
| SubTitleFontSize | Size, in points, for subtitle. | 10 |
| SubTitleFontStyle | Font style (see MainTitleFontStyle for style list). | Plain |
| SubTitleString | Subtitle for the product. | No subtitle. |
| TotalHeight | Height of the total drawing space, which may include multiple graphs, pixels. | 400 |
| TotalWidth | Width of the total drawing space, which may include multiple graphs, pixels. | 400 |

The subproduct properties are associated with the graphs on a page. There can be one or more graphs on a page, each with different properties. It is envisioned that graphs can be grouped into several zoom groups, where zooming in on one graph will cause all graphs to scale similarly. However, at this time, all graphs in a product are placed in a single zoom group. It is also envisioned that graphs will could be placed anywhere on the page; however, at this time, multiple graphs on a page can only be stacked vertically, each using the full width of the page.

The following tables describe the subproduct (graph) properties.

**Subproduct (Graph) Properties**

| Subproduct (Graph) Property | Description | Default |
|---|---|---|
| BarPosition | For use with bar graphs. This property controls how bars are positioned relative to the date and can have the values CenteredOnDate, LeftOfDate, or RightOfDate. | CenteredOnDate |
| BottomXAxisLabelFontName | Name of font for bottom x-axis labels (see Product MainLabelFontName). | Arial |
| BottomXAxisLabelFontSize | Bottom x-axis labels font size, points. | 10 |
| BottomXAxisLabelFontStyle | Bottom x-axis labels font style (see Product MainLabelFontStyle). | Plain |
| BottomXAxisTitleFontName | Name of font for bottom x-axis title (see Product MainTitleFontName). | Helvetica |
| BottomXAxisTitleFontSize | Bottom x-axis title font size, points. | 12 |
| BottomXAxisTitleFontStyle | Bottom x-axis title font style (see Product MainTitleFontStyle). | Plain |
| BottomXAxisLabelFormat | Format for X axis labels. Currently this is confined to date/time axes and only MM-DD is recognized. | Determined automatically. |
| BottomXAxisMajorGridColor | Color to use for the major grid. | Most graph types automatically set to None. |
| BottomXAxisMinorGridColor | Color to use for the minor grid. **This property is not implemented**. | None |
| BottomXAxisTitleString | Bottom X axis title string. | As appropriate for the graph type (often none if dates). |
| DataLabelFontName | Name of font for data labels (see Product MainLabelFontName). | Arial |
| DataLabelFontSize | Data label font size, points. | 10 |
| DataLabelFontStyle | Data label font style (see Product MainLabelFontStyle). | Plain |

**Subproduct (Graph) Properties (continued)**

| Subproduct (Graph) Property | Description | | Default |
|---|---|---|---|
| DataLabelFormat | Format specifiers to use for labeling data points. If blank, no labels will be drawn. If specified, labels are drawn for line graphs and XY scatter plots. The following format specifiers are available (all other text in the format is treated literally). The last three specifiers are related to time series data and all others are related to the date for a point. The %v specifier can be specified twice for XY Scatter plots to display the X and Y values. If specified and the time series data property is not specified, the graph property will be used. | | Blank (no data point labels). |
| | %% | Literal percent. | |
| | %a | Weekday name abbreviation. | |
| | %A | Weekday name. | |
| | %B | Month name. | |
| | %b | Month name abbreviation. | |
| | %d | Day number. | |
| | %H | Hour (0-23), 2-digit. | |
| | %I | Hour (1-12), 2-digit. | |
| | %J | Day of year. | |
| | %m | Month 2-digit. | |
| | %M | Minute, 2-digit. | |
| | %p | AM, PM. | |
| | %S | Second, 2-digit. | |
| | %y | Year, 2-digit. | |
| | %Y | Year, 4-digit. | |
| | %Z | Time zone. | |
| | %v | Data value, formatted according to units. | |
| | %U | Data units. | |
| | %q | Data flag (e.g., quality). | |

**Subproduct (Graph) Properties (continued)**

| Subproduct (Graph) Property | Description | Default |
|---|---|---|
| DataLabelPosition | Indicates the position of data labels, relative to the data point: UpperRight, Right, LowerRight, Below, LowerLeft, Left, UpperLeft, Above, Center. If specified and the time series data property is not specified, the graph property will be used. | Right |
| Enabled | Indicates whether the sub-product should be processed. Specify as true or false. | true |
| GraphHeight | Graph height in pixels. Currently this property is ignored (use Product TotalHeight instead). | Product TotalHeight (minus space for titles, etc.) if one graph, or an even fraction of Product TotalHeight (minus space for titles, etc.) if multiple graphs. |
| GraphType | Indicates the graph type for all data in a graph product. Available options are: Bar, Duration, Line, PeriodOfRecord, Point, XY-Scatter. | Line |
| GraphWidth | Graph width in pixels. Currently this property is ignored (use Product TotalWidth instead). | Product TotalWidth (minus space for titles, etc.). |
| LayoutXPercent | For the product grid layout, the width of the graph as a total width of the product, percent. | 100 divided by the number of columns in the layout. |
| LayoutYPercent | For the product grid layout, the height of the graph as a total width of the product, percent. | 100 divided by the number of rows in the layout. |
| LeftYAxisIgnoreUnits | Indicates whether to ignore units for the lefty Y axis. Normally, units are checked to make sure that data can be plotted consistently. If this property is set, then the user will not be prompted at run-time to make a decision. Specify as true or false. | If not specified, the units will be checked at run-time and, if not compatible, the user will be prompted to indicate whether to ignore units in the graphs. The property will not be reset automatically but will be handled internally using the interactively supplied value. |

**Subproduct (Graph) Properties (continued)**

| Subproduct (Graph) Property | Description | Default |
|---|---|---|
| LeftYAxisLabelFontName | Name of font for left y-axis labels (see Product MainLabelFontName). | Arial |
| LeftYAxisLabelFontSize | Left y-axis labels font size, points. | 10 |
| LeftYAxisLabelFontStyle | Left y-axis labels font style (see Product MainLabelFontStyle). | Plain |
| LeftYAxisLabelPrecision | If numeric data, the number of digits after the decimal point in labels. | Automatically determined from graph type and/or data units. |
| LeftYAxisMajorGridColor | Color to use for the major grid. | Most graph types automatically set to lightgray. |
| LeftYAxisMax | Maximum value for the left Y Axis. | Auto, automatically determined. If the actual data exceed the value, the property will be ignored. |
| LeftYAxisMin | Minimum value for the left Y Axis. | Auto, automatically determined. If the actual data exceed the value, the property will be ignored. |
| LeftYAxisMinorGridColor | Color to use for the minor grid. **This property is not implemented**. | None |
| LeftYAxisTitleFontName | Name of font for left y-axis title (see Product MainTitleFontName). | Arial |
| LeftYAxisTitleFontSize | Left y-axis title font size, points. | 12 |
| LeftYAxisTitleFontStyle | Left y-axis title font style (see Product MainTitleFontStyle). | Plain |
| LeftYAxisTitleString | Left y axis title string. **Note that due to limitations in Java graphics, the left y-axis title is placed at the top of the left y-axis so that it takes up roughly the same space as the y-axis labels. The top-most label is shifted down to make room for the title.** | As appropriate for the graph type (often the data units). |
| LeftYAxisType | Left y-axis type (Log, or Linear). | Linear |
| LeftYAxisUnits | Left y-axis units. **This property is currently used internally and full support is being phased in.** See also LeftYAxisIgnoreUnits. | Units from first valid time series, or as appropriate for the graph type. |
| LegendFontName | Name of font for legend (see Product MainTitleFontName). | Arial |
| LegendFontSize | Legend font size, points. | 10 |
| LegendFontStyle | Legend font style (see Product MainTitleFontStyle). | Plain |

**Subproduct (Graph) Properties (continued)**

| Subproduct (Graph) Property | Description | | Default |
|---|---|---|---|
| LegendFormat | The legend format is composed of literal characters and/or time series data format specifiers, as follows. | | Auto, which uses Description, Identifier, Units, Period |
| | Blank | No legend will be displayed. | |
| | %% | Literal percent | |
| | %A | Time series alias | |
| | %D | Description (e.g., RED RIVER BELOW MY TOWN) | |
| | %F | Full time series identifier (e.g., XX_FREE.USGS.QME.24HOUR.Trace 1) | |
| | %I | Full interval part of the identifier (e.g., 24Hour). | |
| | %b | Base part of the interval (e.g., Hour). | |
| | %m | Multiplier part of the interval (e.g., 24). | |
| | %L | Full location part of the identifier (e.g., XX_FREE). | |
| | %l | Main part of the location (e.g., XX). | |
| | %w | Sub-location (e.g., FREE). | |
| | %S | The full source part of the identifier (e.g., USGS). | |
| | %s | Main data source (e.g., USGS). | |
| | %x | Sub-source (reserved for future use). | |
| | %T | Full data type (e.g., QME). | |
| | %t | Main data type. | |
| | %k | Sub-data type. | |
| | %U | Data units (e.g., CFS). | |
| | %z | Sequence number (used with traces). | |
| | %Z | Scenario part of identifier (e.g., Trace1). | |
| LegendPosition | Position of the legend relative to the graph: Bottom, InsideLowerLeft, InsideLowerRight, InsideUpperLeft, InsideUpperRight, Left, None, Right. | | Bottom |

**Subproduct (Graph) Properties (continued)**

| Subproduct (Graph) Property | Description | Default |
|---|---|---|
| MainTitleFontName | Name of font to use for graph main title (see Product MainTitleFontName). | Arial |
| MainTitleFontSize | Size, in points, for graph main title. | 10 |
| MainTitleFontStyle | Graph main title font style (see Product MainTitleFontStyle). | Plain |
| MainTitleString | Main title for the graph. | None, or appropriate for graph type. |
| PeriodEnd | Ending date for time series data in the sub-product. The date should be formatted according to common conventions (e.g., YYYY-MM-DD HH:mm), and should ideally be of appropriate precision for the data being queried. **This property is often set at run time.** | Full period is read. |
| PeriodStart | Starting date for time series data in the sub-product. The date should be formatted according to common conventions (e.g., YYYY-MM-DD HH:mm), and should ideally be of appropriate precision for the data being queried. **This property is often set at run time.** | Full period is read. |
| RightYAxisLabelFontName | Name of font for right y-axis labels (see Product.MainLabelFontName). **This property is not enabled.** | Arial |
| RightYAxisLabelFontSize | Right y-axis labels font size, points. **This property is not enabled.** | 10 |
| RightYAxisLabelFontStyle | Right y-axis labels font style (see Product MainLabelFontStyle). **This property is not enabled.** | Plain |
| RightYAxisTitleFontName | Name of font for right y-axis title (see Product MainTitleFontName). **This property is not enabled.** | Arial |
| RightYAxisTitleFontSize | Right y-axis title font size, points. **This property is not enabled.** | 12 |
| RightYAxisTitleFontStyle | Right y-axis title font style (see Product MainTitleFontStyle). **This property is not enabled.** | Plain |
| RightYAxisTitleString | Right y axis title string. **This property is not enabled.** | |

**Subproduct (Graph) Properties (continued)**

| Subproduct (Graph) Property | Description | Default |
|---|---|---|
| SubTitleFontName | Name of font to use for graph Sub title (see Product MainTitleFontName). | Arial |
| SubTitleFontSize | Size, in points, for graph sub title. | 10 |
| SubTitleFontStyle | Graph sub title font style (see Product MainTitleFontStyle). | Plain |
| SubTitleString | Sub title for the graph. | No subtitle. |
| TopXAxisLabelFontName | Name of font for Top x-axis labels (see Product.MainLabelFontName). **This property is not enabled.** | Arial |
| TopXAxisLabelFontSize | Top x-axis labels font size, points. **This property is not enabled.** | 10 |
| TopXAxisLabelFontStyle | Top x-axis labels font style (see Product MainLabelFontStyle). **This property is not enabled.** | Plain |
| TopXAxisTitleFontName | Name of font for Top x-axis title (see Product MainTitleFontName). **This property is not enabled.** | Arial |
| TopXAxisTitleFontSize | Top x-axis title font size, points. **This property is not enabled.** | 12 |
| TopXAxisTitleFontStyle | Top x-axis title font style (see Product MainTitleFontStyle). **This property is not enabled.** | Plain |
| TopXAxisTitleString | Top X axis title string. **This property is not enabled.** | As appropriate for the graph type. |
| XYScatterAnalyzeForFilling | Indicate whether the analysis should be used to analyze for filling. If true, then the XYScatterIntercept, XYScatterFillPeriodStart, and XYScatterFillPeriodEnd properties may be specified. | false |
| XYScatterDependentAnalysisPeriodEnd | Specify the ending date/time for the period to analyze the dependent time series data, to determine the best-fit line. | Blank (analyze full period). |
| XYScatterDependentAnalysisPeriodStart | Specify the starting date/time for the period to analyze the dependent time series data, to determine the best-fit line. | Blank (analyze full period). |
| XYScatterFillPeriodEnd | When XYScatterAnalyzeForFilling=true, indicates the ending date/time of the period to fill, using standard date/time string. | Blank (fill full period). |

**Subproduct (Graph) Properties (continued)**

| Subproduct (Graph) Property | Description | Default |
|---|---|---|
| XYScatterFillPeriodStart | When XYScatterAnalyzeForFilling=true, indicates the starting date/time of the period to fill, using standard date/time string. | Blank (fill full period). |
| XYScatterIndependentAnalysisPeriodEnd | Specify the ending date/time for the period to analyze the independent time series data, to determine the best-fit line. | Blank (analyze full period). |
| XYScatterIndependentAnalysisPeriodStart | Specify the starting date/time for the period to analyze the independent time series data, to determine the best-fit line. | Blank (analyze full period). |
| XYScatterIntercept | The value of A in the best-fit equation A + bX. If specified, the value of B is adjusted accordingly. This property cannot be used with transformed data and if specified must be 0. | Blank (do not force the intercept). |
| XYScatterMethod | Curve fit method used when analyzing data for the XY Scatter graph (OLSRegression or MOVE2). | OLSRegression |
| XYScatterMonth | One or more month numbers used when analyzing data for the XY Scatter graph, separated by commas or spaces (1=Jan). | Blank (analyze all) |
| XYScatterNumberOfEquations | Number of equations used when analyzing data for the XY Scatter graph (OneEquation or MonthlyEquations). | OneEquation |
| XYScatterTransformation | Data transformation used when analyzing data for the XY Scatter graph (None or Log). This property is not enabled. | None |
| ZoomEnabled | Indicates whether the graph can be zoomed (true) or not (false). | Graph types are evaluated and the property is automatically set. XY-Scatter and Duration graphs can't zoom. |
| ZoomGroup | Indicate a group identifier that is used to associate graphs for zooming purposes. For example, there may be more than one distinct group of graphs, each with its own overall period or data limits. The graph types may also be incompatible for zooming. **This is an experimental feature and should currently not be specified in product files.** | All graphs are assigned to zoom group 1. |

### *Time Series Properties*

Each subproduct (graph) includes time series data, and the presentation of each time series can be configured using data (time series) properties. In some cases, properties are layered, allowing a property to be defined for the subproduct (graph) for use by all time series (e.g., legend text).

The following tables list data (time series) properties.

**Data (Time Series) Properties**

| Data (Time Series) Property | Description | Default |
|---|---|---|
| `Color` | Color to use when drawing the data. Examples are named colors (e.g., `red`), RGB triplets (e.g., `255,0,128`), and hexadecimal RGB (e.g., `0xFF0088`). | Repeating, using common colors. |
| `DataLabelFormat` | Data label format specifiers. See the graph `DataLabelFormat` property. If the graph property is specified and the time series property is not, the graph property will be used. | Blank (no labels). |
| `DataLabelPosition` | Data label position. See the graph `DataLabelPosition` property. If the graph property is specified and the time series property is not, the graph property will be used. | `Right` |
| `Enabled` | Indicates whether the data should be processed. Specify as `true` or `false`. | `true` |
| `GraphType` | Indicates the graph type for the data in a graph product. Available options are: `Bar`, `Duration`, `Line`, `PeriodOfRecord`, `Point`, `XY-Scatter`. **Currently the sub-product property is used for all data. It is envisioned that this propery will be enabled in the future to allow different data representations to be plotted together (e.g., monthly as bars, daily as line).** | Property not enabled. |
| `LegendFormat` | The legend for the data can be specified and will override the SubProduct `LegendFormat` property (see that property for details). | `Auto` |
| `LineStyle` | Line style. Currently only `None` (e.g., for symbols only) and `Solid` are allowed. | `Solid` |
| `LineWidth` | Line width, pixels. Currently a line width of 1 pixel is always used. | `1` |

**Data (Time Series) Properties (continued)**

| Data (Time Series) Property | Description | Default |
|---|---|---|
| PeriodEnd | Ending date for time series data in the data item. The date should be formatted according to common conventions (e.g., YYYY-MM-DD HH:mm), and should ideally be of appropriate precision for the data being queried. **This property is often set at run time.** | Full period is read. |
| PeriodStart | Starting date for time series data in the data item. The date should be formatted according to common conventions (e.g., YYYY-MM-DD HH:mm), and should ideally be of appropriate precision for the data being queried. **This property is often set at run time.** | Full period is read. |
| RegressionLineEnabled | Indicates whether the regression line should be shown (currently only used with the XY-Scatter graph type). The line is drawn in black (there is currently not a property to set the line color). | true |
| SymbolSize | Symbol size in pixels. | 0 (no symbol) |
| SymbolStyle | Symbol style. Recognized styles are:<br>• None<br>• Arrow-Down, Arrow-Left, Arrow-Right, Arrow-Up<br>• Asterisk<br>• Circle-Hollow, Circle-Filled<br>• Diamond-Hollow, Diamond-Filled<br>• Plus, Plus-Square<br>• Square-Hollow, Square-Filled<br>• Triangle-Down-Hollow, Triangle-Down-Filled, Triangle-Left-Hollow, Triangle-Left-Filled, Triangle-Right-Hollow, Triangle-Right-Filled, Triangle-Up-Hollow, Triangle-Up-Filled<br>• X, X-Cap, X-Diamond, X-Edge, X-Square | None |

**Data (Time Series) Properties (continued)**

| Data (Time Series) Property | Description | Default |
|---|---|---|
| TSID | Time series identifier. | Must specify. |
| XAxis | X-axis to use (Bottom or Top). **This currently always defaults to bottom.** | Bottom |
| XYScatterConfidenceInterval | This property is only used with XY scatter plots. If not blank, the value indicates that confidence level lines should be drawn on the XY Scatter plot for the given confidence interval, percent. Currently only 99 and 95 percent confidence intervals are supported. The lines will only be drawn if the curve fit line is drawn (see RegressionLineEnabled). | Blank (do not draw). |
| YAxis | Y-axis to use (Left or Right). **This currently always defaults to left.** | Left |

***Annotation Properties***

Annotations are associated with subproducts (graphs) and are implemented as simple shapes that are drawn on normal graphs. It is envisioned that all shapes supported by the drawing package will eventually be supported but currently only text labels and lines can be specified as annotations.

To allow flexibility, annotations can be placed using two coordinate systems. For example, if a product is generated using real-time data, the date/time axis will have a different range over time. Therefore, placing an annotation using a fixed coordinate would cause the annotation to scroll off the graph as time passes. To resolve this issue and still allow absolute positioning of annotations, as appropriate, the following coordinate systems are supported, as specified by the XAxisSystem and YAxisSystem properties:

Data — When using the data coordinate system, it is expected that the coordinates used to define the annotation will agree with the data units being drawn. For example, for a normal time series graph, the x-axis coordinate would be specified as a date/time to the necessary precision and the y-axis coordinate would be specified using data values.

It is envisioned that a notation +NNN and –NNN will be implemented in the future to allow offsets from the edges of the graph, in data units.

Percent — When using the percent coordinate system, it is expected that the coordinates used to define the annotation are specified as a percent of the graph width or height, with 0 being the lower/left and 100 being the upper/right.

Each axis can have a different coordinate system (e.g., the y-axis value can be set using data units and the x-axis value can be set using percent).

The following tables list annotation properties.

**Annotation Properties (All Shapes)**

| Annotation Property | Description | Default |
|---|---|---|
| `AnnotationID` | A string that identifies the annotation, to be used in software displays. If there are many annotations, this helps identify them when editing. | `Annotation` + annotation number (1+) (e.g., `Annotation1`). |
| `Color` | Color to use when drawing the annotation. Examples are named colors (e.g., `red`), RGB triplets (e.g., `255,0,128`), and hexadecimal RGB (e.g., `0xFF0088`). | Black |
| `Order` | The drawing order for the annotation, either `BehindData` to draw behind time series data or `OnTopOfData` to draw on top of time series data. | `OnTopOfData` |
| `ShapeType` | The type of shape to be drawn for the annotation. Currently accepted values are `Text` and `Line`. | None – must be specified. |
| `XAxisSystem` | Indicates the system for X coordinates:<br>• If `Data`, the X coordinates that are specified will be in data units.<br>• If `Percent`, the X coordinates are percent of the graph (0% is left and 100% is right). | `Data` |
| `YAxisSystem` | Indicates the system for Y coordinates:<br>• If `Data`, the Y coordinates that are specified will be in data units.<br>• If `Percent`, the Y coordinates are percent of the graph (0% is bottom and 100% is top). | `Data` |

**Annotation Properties (Line Shape)**

| Annotation Property | Description | Default |
|---|---|---|
| `LineStyle` | Line style. Currently only `None` and `Solid` are allowed. | `Solid` |
| `LineWidth` | Line width, pixels. Currently a line width of 1 point (pixel) is always used. | `1` |
| `Points` | X and Y coordinates for the line endpoints, as follows: `X1,Y1,X2,Y2` or `X1,Y2 X2,Y2`. | None – must be specified. |

**Annotation Properties (Text Shape)**

| Annotation Property | Description | Default |
|---|---|---|
| `FontSize` | Annotation text font size, points. | `10` |
| `FontStyle` | Annotation text font style (see Product `MainLabelFontStyle`). | `Plain` |
| `FontName` | Annotation font name (see Product `MainTitleFontName`). | `Arial` |
| `Point` | X and Y coordinates for the text position, as follows: `X1,Y1` | None – must be specified. |
| `Text` | The string to display. | `Blank` |
| `TextPosition` | Indicates the position of text, relative to the point: `UpperRight`, `Right`, `LowerRight`, `Below`, `LowerLeft`, `Left`, `UpperLeft`, `Above`, `Center`. | `Right` |

This page is intentionally blank.

# Appendix

# GeoView Mapping Tools

Color, 2004-05-27, Original Maintained with TSTool, Acrobat Distiller



## Overview

The GeoView package contains integrated software components that can be used with software to enable map-based interfaces. The main purpose of the GeoView package is to provide simple and flexible map displays that can be used in a variety of software applications with little or no reconfiguration.

The GeoView package has been developed by Riverside Technology, inc., using Java technology. GeoView interfaces can be embedded in Java applications (e.g., use GeoView as the main window interface), can be enabled as a separate floating window (e.g., to support an application's features without being embedded in the main window), and can be used in web pages either as embedded map applets or stand-alone map windows. GeoView tools operate similarly on Microsoft Windows and UNIX operating systems.

This appendix describes general GeoView features and can be used as a reference for how to configure and use GeoView components. Specific uses of GeoView in a software program are discussed in the documentation for the specific software. Some of the figures shown in this documentation were generated using GeoView with the TSTool software and consequently title bars include TSTool in the wording.

## GeoView Terminology

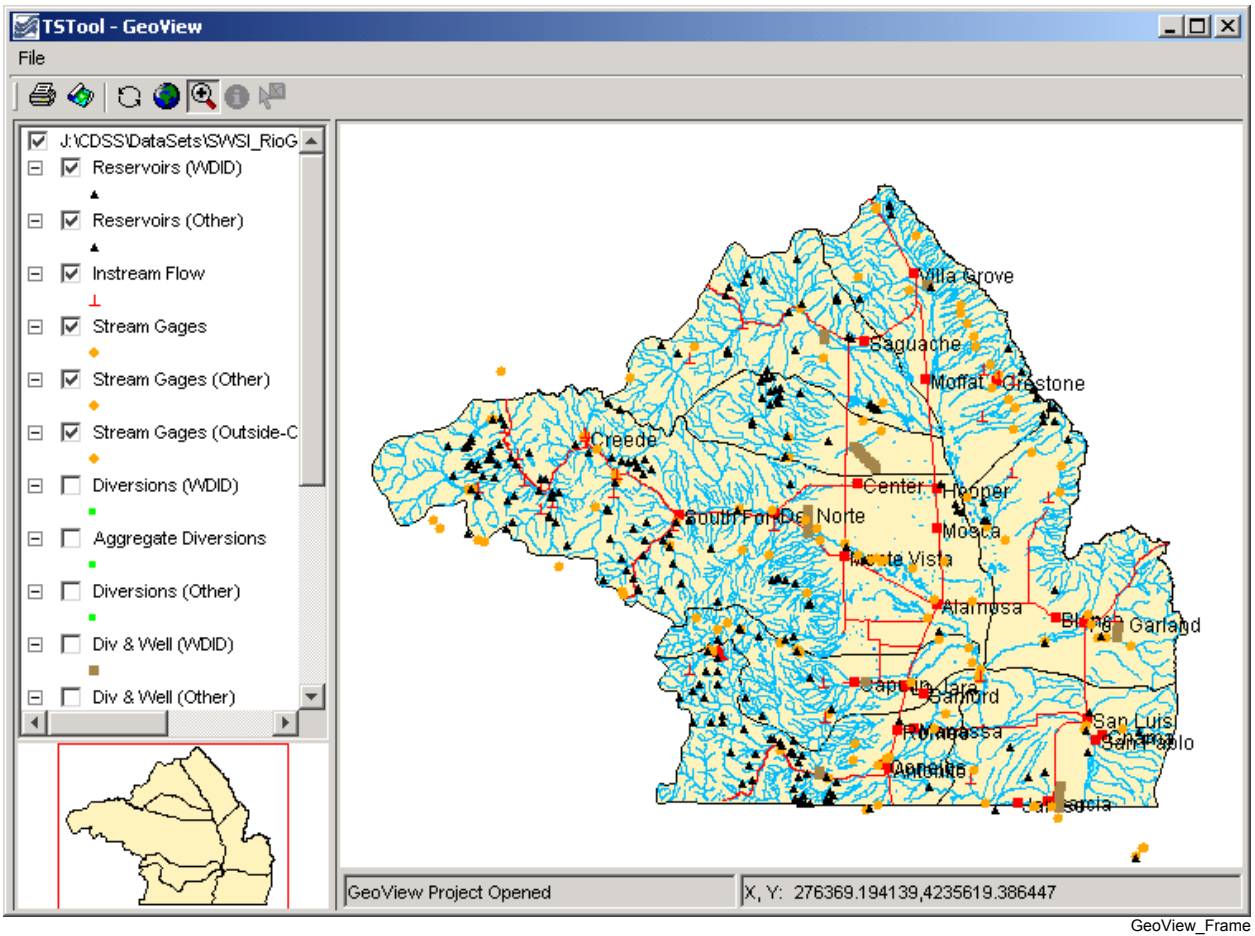
GeoView terminology is similar to other GIS product terminology. Important terms are shown in the following table. These terms are used infrequently in most user interfaces and applications but are visible at times in property dialogs and configuration files.

**GeoView Terminology**

| Term | Description |
|---|---|
| *GeoView* | The visible map window where maps are displayed. Currently there can be only one main GeoView. A reference GeoView may be used to show the zoom extents in the main GeoView. |
| *GeoLayer* | A data layer, in its "raw" form (e.g., an ESRI Shapefile). The more generic term "layer" is often used. |
| *GeoLayerView* | A view of a GeoLayer, with symbol properties for visualization. The more generic term "layer view" is often used. This is equivalent to a "theme" in some software packages. |
| *Feature* | A general term describing an item on the map, consisting of shape and attribute data. |
| *Shape* | A general term defining the type of feature (e.g., point, polygon). The shape type defines the ways that a feature can be symbolized and used in analysis. Shape types can typically be determined automatically from input data. |
| *Attributes* | A general term defining non-shape data that are associated with a feature. Often, attributes are stored in a tabular form, such as a relational database table. Attributes are usually associated with a shape using some type of index number (shape index). |
| *Symbol* | The combination of properties used to visualize a layer (e.g., symbol style, color, labeling, classification). The feature shape type controls how the feature can be symbolized. |
| *GeoView Project* | A GeoView Project file (*.gvp*) can be used to define the layers and global viewing properties for a GeoView. The contents of this file are described in more detail in the **GeoView Configuration – the GeoView Project File** section at the end of this appendix. |
| *Application Layer Type* | Because GeoView is a generic tool, it has no implicit understanding of the types of data that are important to an application. The `AppLayerType` is a property that can be assigned to layers in a GeoView Project file to help an application know that a layer is important to the application. An application layer type of "BaseLayer" indicates that a layer should be used for background information and is not specific to the application. See the **Using GeoView with a Software Application** section below for more information. |

## The GeoView Panel

An example GeoView interface is shown in the following figure. This example uses a floating GeoView window. Some programs use a GeoView that is embedded in the main application window, and some rely on secondary map windows (as shown below). If the map is in the main window, the menus at the top of the window will be those specific to the software (whereas below the single GeoView ***File*** menu is shown).




GeoView_Frame

**Example GeoView Interface (from TSTool)**

The GeoView Panel is a self-contained component that offers a standard map-based interface that can be used in many applications. In the above figure, the GeoView Panel includes everything shown, except for the top menu bar (with the ***File*** menu). The general purpose GeoView Frame includes the menu bar and a GeoView Panel. The GeoView Panel contains the following components:

| | |
|---|---|
| Table of Contents (left edge) | The table of contents displays a list of layer views, showing the top-most layer at the top of the legend. Layers can be enabled/disabled by toggling the check box. A layer can be selected/deselected by clicking on the layer in the table of contents. Layers that are selected can be acted on (e.g., properties can be viewed). The table of contents also indicates the symbol for the layer. |
| Main GeoView (large map) | The main GeoView displays the enabled layers and allows you to interact with the map using the mouse and keyboard (e.g., zoom, select). |

| | | | |
|---|---|---|---|
| Reference GeoView (lower left) | The reference GeoView displays layers that have the property `ReferenceLayer` set to `true`. This view shows the current zoom extent relative to the maximum extent of the data and can also be used to initiate a zoom to a region on the main map (the reference map is always in zoom mode). | | |
| Standard Controls (top, below menu bar) | Standard controls perform actions on the visible map as follows: | | |
| |  | ***Print*** | Print the visible map. You will be able to pick the printer and orientation. |
| |  | ***Save Image*** | Saves the map as a Portable Network Graphics (PNG), JPEG, or other supported graphic file format. |
| |  | ***Refresh*** | Refresh the map display by redrawing features in enabled layers that are in the visible window. This does **not** re-read the original data. GeoView normally refreshes automatically as needed. |
| |  | ***Zoom Out*** | Zoom to the maximum data extents. |
| |  <br>  <br>  | ***Select the Mode as Zoom, Info, or Select*** | Select the interaction mode. The ***Zoom*** mode allows a rectangle to be drawn on the map to zoom to the specified region and is the default mode if no layer is selected. The ***Info*** mode allows features to be selected (by clicking on or drawing a box around), after which geographic information about the features is displayed. The ***Select*** mode is similar to ***Info***; however, its purpose is to select features for an additional action (e.g., exporting data or performing a query). The ***Info*** and ***Select*** modes are only enabled if one or more layers are selected in the table of contents.<br><br>See the next section for more information about using these features. |
| Message Areas (bottom) | The message areas are used to display the mouse coordinates and provide other feedback. | | |

The GeoView Panel components work with each other to provide interaction with the maps, as described below.

## Interacting with the GeoView Map

The layers shown on the map are initially displayed according to the GeoView Project settings. Once displayed, you can interact with the map in the following ways:

**Disable/Enable a layer view**

Layers can be enabled/disabled to make the map more readable or useful:

1. Use the check boxes in the table of contents to disable and enable layer views, as appropriate. The map will automatically refresh, resulting in a slight delay as the map is redrawn.
2. If necessary, use the ***Refresh*** tool (  ) to cause the map to be updated (automatic refresh may be disabled for some applications, due to performance reasons).

**Change layer view order**

**Currently the layer view order can only be changed by editing the GeoView Project file.**

**Zoom in/out**

Zooming is useful make symbols and labels more readable. To zoom in:

1. Set the GeoView interaction mode to "Zoom" by selecting the zoom tool (  ) at the top of the window.
2. Use the mouse to draw a box around an area of interest (left mouse button down to start, move the mouse, and then release). The main GeoView map will zoom to the selected region and the reference map will show the zoom extent.
3. Use the ***Zoom Out*** tool (  ) to zoom to the full extent or use the reference GeoView to zoom to a different region.

**Change symbols for a layer view**

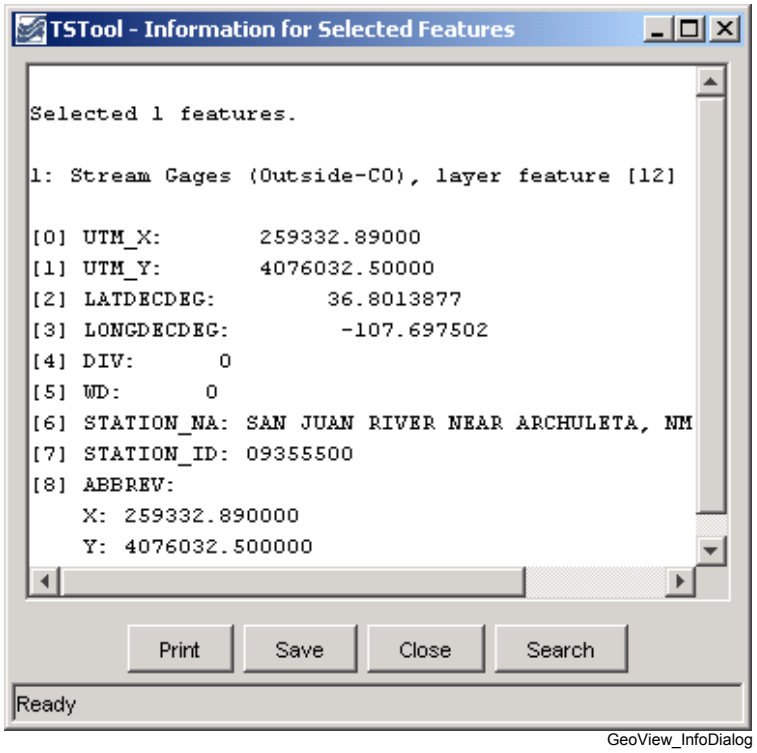
To change the symbols and labels for a layer view:

1. Select the layer view in the table of contents
2. Right-click and select the ***Properties*** menu. See the **Setting GeoView Properties** section below for information about the properties.

**Display geographic information for features**

The GeoView interface can display information about geographic features (shape and attribute data) from the original geographic data. To do so:

1. Select layer views in the table of contents that are to be searched for information.
2. Set the GeoView interaction mode to Info (  ).
3. Click near the feature or draw a box around multiple features. The layers will be searched and the following dialog will be shown.



GeoView_InfoDialog

The resulting dialog will show information about the selected features, including basic layer information, and information about the specific shapes and attributes. **The display is for geographic data only. Attribute names and values are as they appear in the original data. Additional application-specific data are typically provided by a separate software interface.**

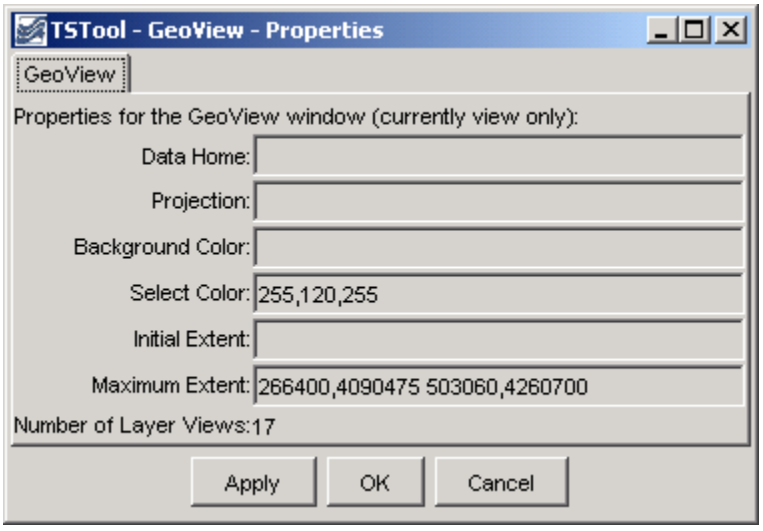
Select features

Features can be selected for a number of reasons. Currently, GeoView has limited select tools, which are mainly used internally when integrated with an application (e.g., an application can select features internally, which are then highlighted on the map). In the future, interfaces to select features from the GeoView interface using query criteria may be added.

Features can be selected (  ) similar to the ***Info*** mode described above. The selected features are highlighted on the map. In the past, yellow, or cyan have been used to highlight selected features. However, yellow is not clearly visible when earth-tone colors are used for background layers and cyan is not clearly visible when water-tone colors are used for background layers. Therefore, GeoView is phasing in a magenta/pink selection color, which is rarely used for background layers.

## Setting GeoView Properties

GeoView properties are initially set in a GeoView Project file or are assigned internally by the software. Most properties control how layers are displayed (colors, labels, etc). To view general GeoView properties, right click on the GeoView map and select the ***Properties*** menu. Some properties are currently view-only. Refer to the **GeoView Configuration – the GeoView Project File** section below for a complete list of properties that can be defined in a GeoView Project file.



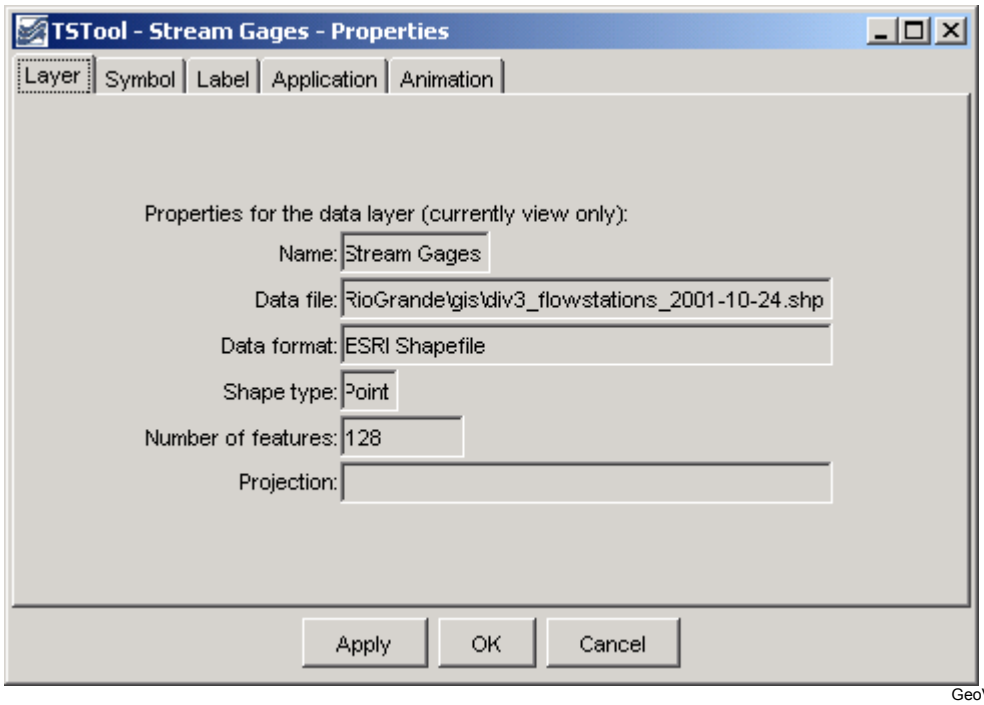

GeoView_Props

**Main GeoView Properties**

GeoView properties, as shown in the above figure, apply to the main GeoView and are shared between layers. These properties are typically not edited by end users. One important property is the projection property. If all data layers are projected consistently (e.g., for ESRI shapefiles) then a projection does not need to be defined. However, if the layers have different projects, a GeoView projection and projections for each layer can be defined to allow the GeoView to project data consistently for visualization.

If the ***OK*** or ***Apply*** buttons are pressed, the GeoView properties will be updated in memory (the GeoView Project file is not updated) and the map will redraw. Pressing ***OK*** will additionally close the properties dialog. The ***Cancel*** button causes the dialog to close, without updating the properties.

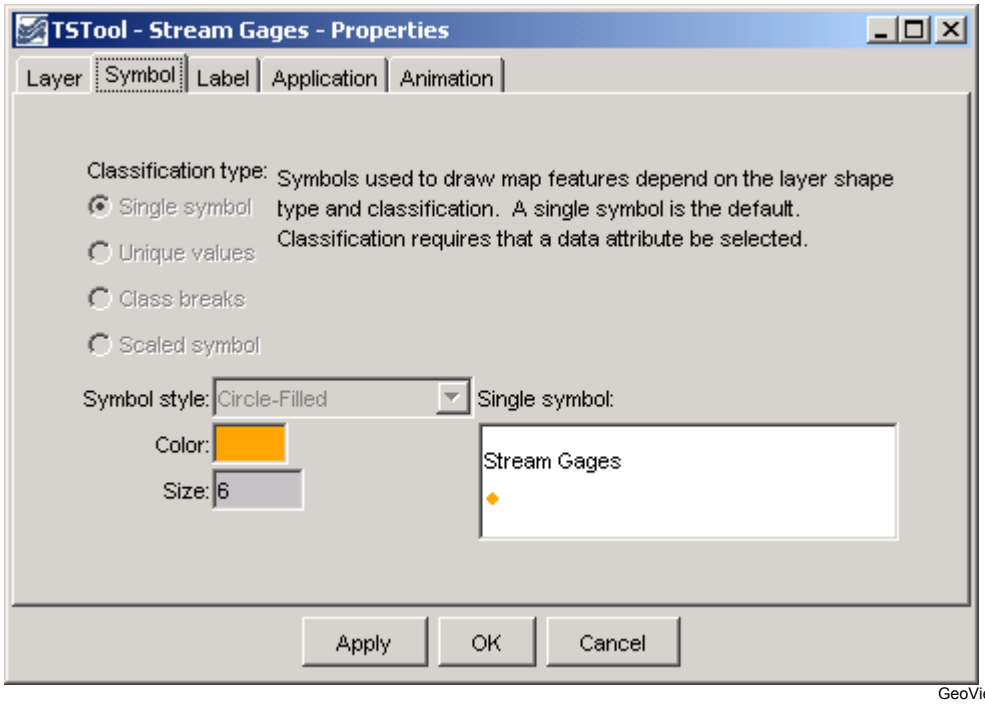
To view or change properties for a layer, select a layer view in the table of contents, right-click and select the ***Properties*** menu item. The following tabbed dialog will be displayed for the first selected layer view. The tabbed panels are discussed below the each figure.




GeoView_Props_Layer

**GeoView Layer Properties**

GeoView layer properties, as shown in the above figure, apply to the input source. Currently these properties are used for information purposes and cannot be interactively edited.

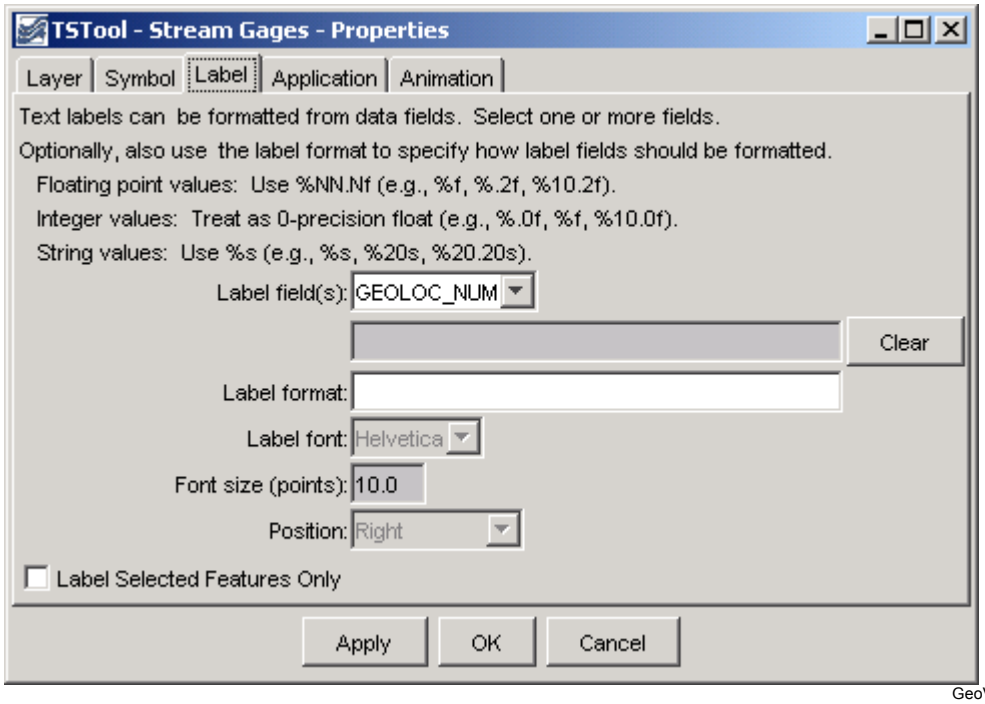



GeoView_Props_Symbol

**Layer View Symbol Properties**

Symbol properties, as shown in the above figure, indicate how the layer is to be drawn (symbolized) on the map and in the table of contents. A sample of the symbol is shown in the dialog, although it may appear slightly different on the map and table of contents.

Symbol terminology corresponds to standard GIS tools.



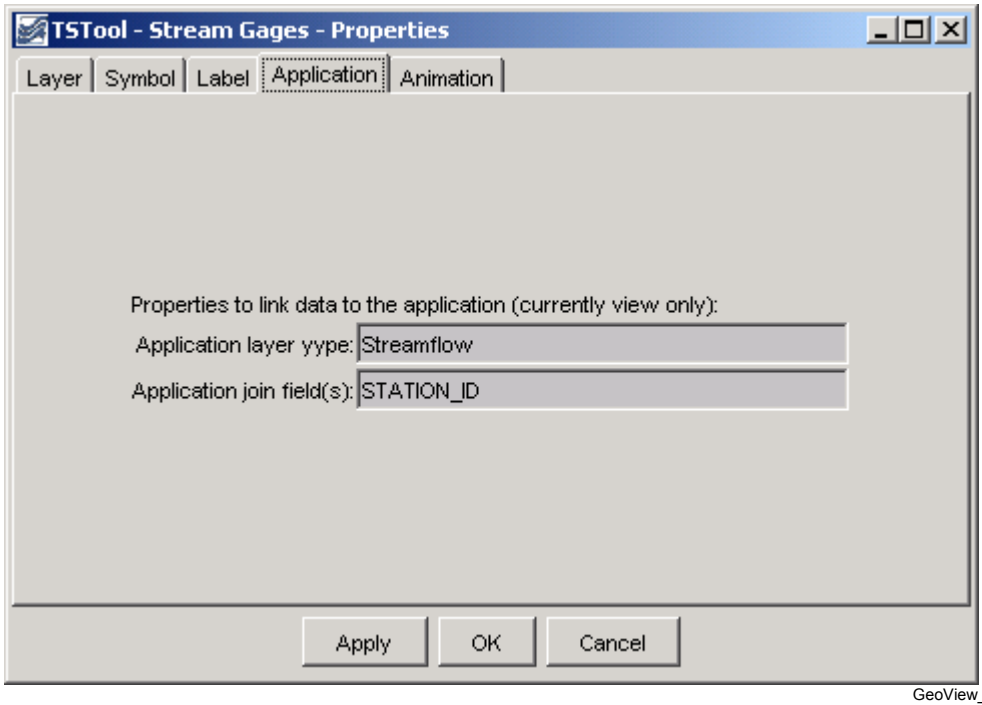

GeoView_Props_Label

**Layer View Label Properties**

Label properties, as shown in the above figure, can be modified to label features with attribute data or literal strings. Currently, only point features can be labeled. Labels can consist of a combination of attribute values. To label features, select the attribute fields from the available choices, in the order that they should appear in the label.

The label format, if not specified, defaults to the use the full field with of the attribute. For example, if an attribute field is defined as being twenty characters wide, the label may be the full width, including leading and trailing spaces. More often, it is desirable to omit the spaces. To do so, or to format numbers using a more appropriate format than the full. width default, use the ***Label Format*** information. The dialog box notes illustrate valid formats. For example, if a string field and an integer field are available, the following label format would show the labels with only a comma and one space between the values:

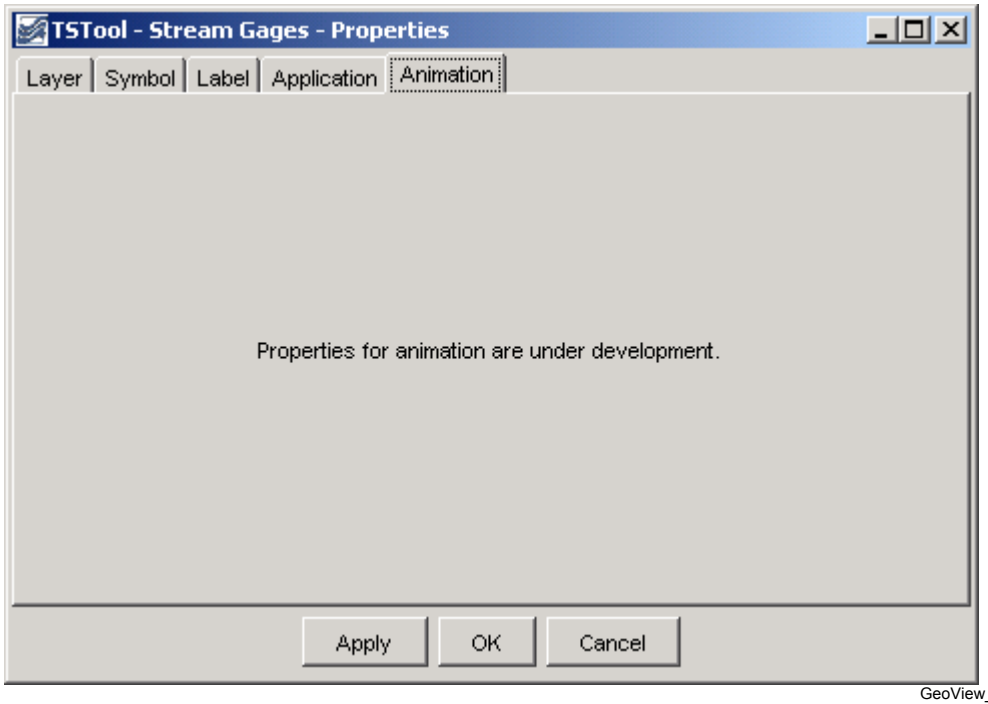
```
%s, %d
```




GeoView_Props_Application

**Application Layer Type**

Layer application properties (above) are used to link a layer's data to an application. This process allows general GeoView features to be used more specifically by specific software programs. The **Using GeoView with a Software Application** section (see below) describes this functionality in more detail.



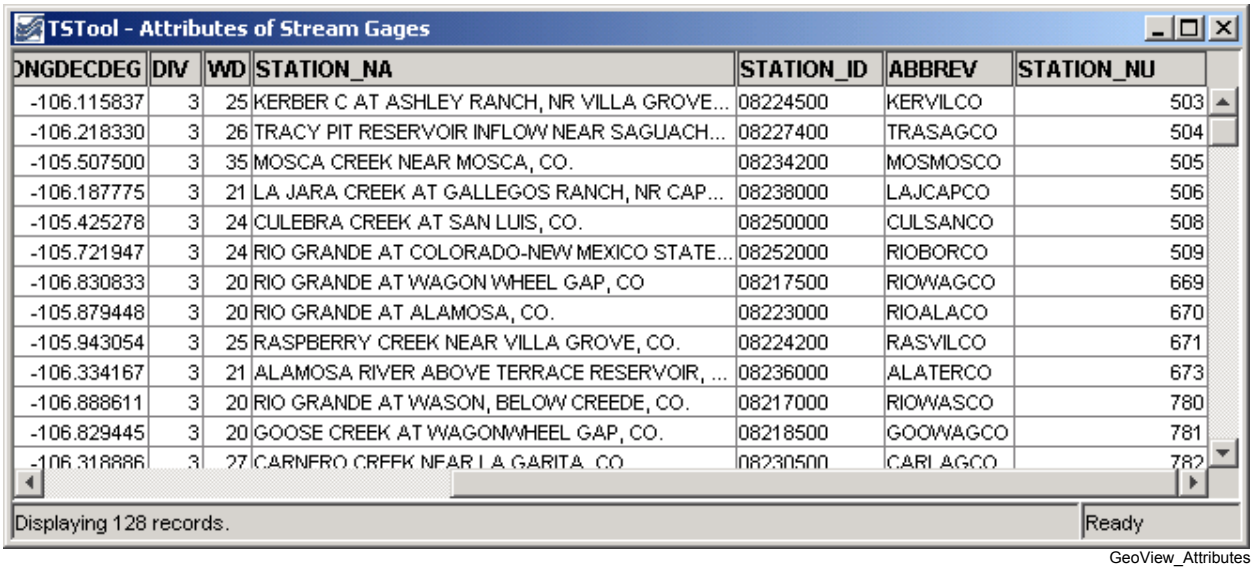

GeoView_Props_Animation

**Layer View Animation Properties**

Layer view animation properties are currently under development. Animation properties will define, for example, the time series data that are used for symbolization during animation.

## Viewing a Layer's Attributes

Each feature in a layer includes geographic shape information (e.g., the coordinates that define a polygon). Each feature also can have attribute data, which are typically represented in a tabular fashion. To view the attributes for a layer, first select the layer in the table of contents, then right-click and press the ***View Attribute Table*** menu choice. A window similar to the following will be shown:




GeoView_Attributes

**Attributes Table for a Layer**

The attributes are displayed in the order and format determined from the input data. Attribute names in ESRI shapefiles are limited to ten characters. Information in the table can be selected (use ***Ctrl-A*** to select all) and can be copied to other software.

## Using GeoView with a Software Application

Software developers integrate the GeoView components with software applications and typically the software user does not need to know how GeoView works with the application. However, this section describes a few important concepts that will help facilitate setting up data for use by an application.

Basic GeoView implementations involve defining a GeoView Project (see the **GeoView Configuration – the GeoView Project File** section below for details on project file properties) and then interacting with the GeoView interface when the map is displayed. In a basic application, a GeoView can be added to show maps for reference purposes only. For example, the application may be an interface to a database containing location information. If a GeoView project file is defined with only base layers, then the zooming and features will allow a user to zoom into a region of interest, but there will be no interaction between the GeoView and the application.

In a more advanced application, layers in the GeoView Project file are assigned an `AppLayerType` property, which is recognized by the application. For example, a layer may be assigned an application type of "Streamflow" to indicate a streamflow gage. Additionally, the `AppJoinField` property can be defined to allow the application to join its data to the geographic data. This assignment in and of itself causes no effect in the GeoView. However, the application can now interact with the GeoView by asking for the "Streamflow" layer. This allows features in the GeoView to be selected from the application (e.g., in a database query screen) and allows the GeoView to provide information about the layer to the

application. For this type of implementation, it is important that the application layer types, feature (shape) type, and the required join fields are documented; consequently, new data layers can be used with the application with only a few configuration changes.

Some applications may automatically update the map interface by zooming to selected areas, selecting features, etc. Standard GeoView features are typically still available, as previously described.

## Limitations

The GeoView components have been developed not to serve as full-featured GIS components, but to support many common GIS activities like selection, zooming, and symbolization. The components have been developed to integrate with existing applications and use other tool sets, including time series viewing tools. Basic features have been implemented to address important needs for applications; however, additional features are implemented as requirements change. The GeoView components are not envisioned as a replacement for pure GIS tools like ESRI's ArcGIS products. In many cases, ESRI or other tools can be used to develop the data for use with GeoView.

Currently, properties that are changed interactively cannot be saved to the GeoView Project file.

GeoView software currently does not examine projection files optionally distributed with ESRI shapefiles. Projections must be defined in the project file (or, if omitted, the projection is assumed to be consistent for data layers). Only a few projections are recognized, as needed by specific GeoView software implementations.

## GeoView Configuration – the GeoView Project File

A GeoView display is configured mainly by using a GeoView Project (*gvp*) file, which is either read at software startup or when selected by the user. The purpose of the file is to persistently store the configuration of a map display so that it can be loaded again without redefining the configuration. The file format is simple text properties and can be read by applications implemented in various technologies running in various environments. An example of the file is shown below.

```
# Main properties global to the GeoView
# The format is:
# [Prefix]
# Prop=value
[GeoView]
GeoDataHome = .

# Properties for each GeoLayerView (data source and
# symbols)...
[GeoLayerView 1]
GeoLayer = xxx.shp

[GeoLayerView 2]
GeoLayer = xxx.shp
```

The GeoLayerViews listed first in the project file are drawn first and are therefore behind other layers on the map. For all properties, the comma is used as an internal delimiter and the semi-colon is used as a second layer of separation, as appropriate. Most properties will default to appropriate values if not specified (see tables below). The most important properties, as shown in the example above, are the `GeoDataHome`, which defines where data can be found, and `GeoLayer`, which defines where the data file is for each layer. Recognized layer file formats are listed in the following table and are described further in separate appendices. Support for additional layer types can be added as necessary.

| Layer Type | Description |
|---|---|
| ESRI shapefile | ESRI shapefiles are commonly used with ESRI software such as ArcView, ArcMap, and ArcExplorer. GeoView determines the type by looking for the *.shp* file extension and checking the internal file format. No projection is assumed but the `Projection` property for the GeoView and individual layers can be used to indicate the projection. |
| NWSRFS GeoData files | The National Weather Service River Forecast System (NWSRFS) uses ASCII and binary files defining various geographic layers. This format is detected by checking the file names, which are predefined for NWSRFS. The `Projection` property is defined as `Geographic` if ASCII data and HRAP if binary data. |
| NWS XMRG Radar Files | XMRG files are gridded radar files produced by the National Weather Service. GeoView treats these files as grid files. This format is detected by looking for an "xmrg" string in the filename. The `Projection` property is defined as HRAP. |

The following main GeoView properties can be defined in the project file. Graphical user interfaces to allow interactive editing of all properties are being implemented.

| Main GeoView Property | Description | Default Value |
|---|---|---|
| `Color` | Background color for map. See the discussion after the properties tables for a discussion of how to define colors. | `White`. |
| `FontName` | Name of font to use for GeoView components (e.g., buttons). This property currently can only be set internally with software. | System-specific. |
| `FontSize` | Size of font, in points, to use for GeoView components (e.g., buttons). This property currently can only be set internally with software. | System-specific. |
| `FontStyle` | Font style to use for GeoView components (e.g., buttons). This property currently can only be set internally with software. | System-specific. |
| `GeoDataHome` | Directory where the GIS data exist. This directory will be prepended to layer files if they are not absolute paths already. | If not specified or if specified as ".", the directory will be set as the home of the GeoView Project file. |
| `InitialExtent` | Initial extent of the map display, in data coordinates. The coordinates should be specified as "`XMIN,YMIN XMAX, YMAX`", where the first pair is the lower-left corner of the extents and the second pair is the upper right. **This property has not been implemented. See the MaximumExtent property.** | No default. The initial extent will be the maximum data extent. |
| `MaximumExtent` | Maximum extent of the map display when zoomed out, in data coordinates. The coordinates should be specified as "`XMIN,YMIN XMAX,YMAX`", where the first pair is the lower-left corner of the extents and the second pair is the upper right. | No default. The maximum extent will be the maximum data extent. |
| `Projection` | Projection for the GeoView. The projection definition varies depending on the projection (some projections require more parameters). The following projections are currently supported:<br><br>`Geographic` - no projection (decimal degrees)<br><br>`HRAP` - used by National Weather Service<br><br>`UTM,Zone[,Datum,FalseEasting] [,FalseNorthing][,CentralLongitude] [,OriginLatitude][,Scale]` - Universal Transverse Mercator. The `Zone` is required (e.g., `13` for Colorado). `Datum` defaults to `NAD83`. The `FalseEasting` defaults to `500000`. The `FalseNorthing` defaults to `0`. The `CentralLongitude` is computed from the `Zone`. The `OriginLatitude` defaults to zero. The Scale defaults to `.9996`. | No default. All data are assumed to be the same projection. |
| `ProjectAtRead` | Indicates whether layer features are projected at read-time to the GeoView projection. This slows down the application initially but increases performance later during map refreshes. | `false` (it is usually best to project all data to a common projection rather than relying on GeoView to do projections) |
| `SelectColor` | Color to use for selected features. See the discussion after this table for examples of how to specify colors. | `Yellow`<br><br>A more unique magenta/pink color with RGB 255,120,255 is being considered. |

The following GeoLayerView properties can be defined, corresponding to each data layer/file:

| GeoLayer View Property | Description | Default Value |
|---|---|---|
| `AppJoinField` | Specify the field(s) that should be used by an application to join the layer data to application data. If multiple fields are necessary, separate the field names by commas (e.g., `"wd,id"`). | None. |
| `AppLayerType` | Indicate a layer type to be handled by an application. For example, a layer may be tagged as "Streamflow". The application can then use this information to treat the layer differently (e.g., to know how to join the data to application data). Valid `AppLayerType` values must be defined and understood by the application. | None. |
| `Color` | Color for features when the `SymbolClassification` is `SingleSymbol`. If point data, this is the main color for the symbol. If line data, this is the line color. If polygon data, this is the fill color. See the discussion after this table for examples of how to specify colors. | Random. |
| `ColorTable` | Used when the `SymbolClassification` property is `ClassBreaks` or `UniqueValues` and requires more than a single color. The number of colors should be one more than the number of class break values if `SymbolClassBreaks` is used and equal the number of class values if `UniqueValues` is used. Color tables can be defined in three ways:<br><br>1. `ColorTableName;NumColors`<br>Predefined tables include `Gray`, `BlueToCyan`, `BlueToMagenta`, `BlueToRed`, `CyanToYellow`, `MagentaToCyan`, `MagentaToRed`, `YellowToMagenta`, and `YellowToRed`. These named tables choose primary colors where necessary to provide clean color breaks.<br><br>2. `Ramp;NumColors;Color1;Color2`<br><br>3. `Custom;NumColors;Color1;...;ColorN`<br><br>**Only the first option is currently enabled**. See the discussion after this table for examples of how to specify colors. | Named color table using `Gray`. |
| `EventLayerView` | Indicates if the layer view is an event layer (ESRI Map Object notation). This property is not currently used in the Java tools. | `false` |
| `GeoLayer` | Name of file for the data layer, typically with the file extension. If an ESRI shapefile, specify the *.shp* file. If a relative path, the `GeoView.GeoDataHome` property will be prepended to the file name. This property is used to detect a break in the GeoLayerView numbering, indicating the end of layer views. | No default. Should always be specified. |
| `IgnoreDataOutside` | Indicate a range of values that should be considererd. Currently this applies only to grid layer types. Specifying a range can be used, for example, to draw only cells with positive data values. The range should be specified as two numbers separated by a comma (e.g., `.00001,100.0`). | Not specified. All cells are considered. |
| `LabelField` | Specify one or more fields to be used for the label, separated by commas. If a `LabelFormat` property is specified, use it to format the label; otherwise, format each field according to the field specifications from the attribute data source. | No default. Specify one or more fields for the label. |

| GeoLayer View Property | Description | Default Value |
|---|---|---|
| LabelFontName | Font to use for labels. **This property has not been enabled.** | Helvetica |
| LabelFontSize | Label font height, points. **This property has not been enabled.** | 10 |
| LabelFormat | Specify a C-style format string to format the fields. The format specifications must agree with the data types being formatted. For example, if two floating-point fields are specified with the LabelField property, the corresponding format may be "%10.1f, %5.2f". | If not specified, the label will be formatted using the field width and precision determined from the data table, with values separated by commas. |
| LabelPosition | Label position. If point data, the position is relative to the point coordinates. If line or polygon data, the position is relative to the centroid coordinates. The position of the text will be offset to not overwrite a symbol and can be UpperRight, Right, LowerRight, Below, LowerLeft, Left, UpperLeft, or Above. | Right |
| Name | The layer view name that should be displayed in the legend. | No default. If not specified, the file name will be used in the legend. |
| OutlineColor | Outline color for point or polygon symbols. See the discussion after this table for examples of how to specify colors. | Default to the same as main color). |
| Projection | Projection for the layer's data. See the main GeoView Projection property for available values. | No default. It is assumed that all data in a project have a consistent projection. |
| ProjectAtRead | Indicates whether features are projected at read-time to the GeoView projection. This slows down the application initially but increases performance later during map refreshes. This property can be set once in the GeoView main properties. | false (it is usually best to project all data to a common projection rather than relying on GeoView to do projections) |
| ReadAttributes | Indicates whether attributes should be read when the data are read. If possible, based on the layer data format, attributes will be read on the fly as needed. Reading the attributes (true) takes more memory but will result in faster performance. | false |
| ReferenceLayer | Indicates whether the layer should be drawn in the reference GeoView. Indicate as true or false. Typically only the most general boundary information should be used in the reference layer. | false |
| SelectColor | Specify the color to be used when drawing selected features. This property is useful if the default select color does is not easily viewed. | Use the GeoView. SelectColor property. |
| SkipLayerView | Can be set to true to skip the layer altogether when reading the project file (useful for commenting out layers during development). If this property is used, the number sequence for the layer views can be kept the same. | false (the layer view will be displayed) |
| SymbolClassBreaks | Class breaks that correspond to the ClassBreaks SymbolClassification property. The number of values should be one less than the number of values in the ColorTable property for the classification. | No default, although an application may suggest values. |
| SymbolClassField | Attribute data field that is used when the SymbolClassification property is ClassBreaks or UniqueValues. | No default, although an application may suggest a value based on the available attributes. |

| GeoLayer View Property | Description | Default Value |
|---|---|---|
| `SymbolClassification` | Indicates how data are to be classified when displaying the shape symbols. Values can be `SingleSymbol` (e.g., single point symbol, line style, or polygon fill color), `UniqueValues` (display a unique symbol style for each value, **currently not implemented**), or `ClassBreaks` (display a unique symbol style for groupings of values - requires specification of the `SymbolClassField` and `SymbolClassBreaks` properties). | `SingleSymbol` |
| `SymbolSize` | For point data, specify the symbol size in pixels. For line data specify the line width in pixels. Not used for polygon data. This property may need to be expanded to properly handle printed output (might need to use points rather than pixels or allow the units of measure to be set in the property). **This property is currently not enabled.** | 6 pixels for points, 1 pixel for lines. |
| `SymbolStyle` | Indicates the symbol style. If point symbols, the style is the symbol identifier (e.g., `CircleFilled`). If line data, the symbol style is the line style (currently only `Solid` is supported). If polygon data, the symbol style is the fill patter (currently only `FillSolid` is supported). See below for a full discussion of symbol styles. | `None` for points, `Solid` for lines, `FillSolid` for polygons. |

#### Color Specification

Colors are specified for a number of different properties, including the feature color and outline color. In order to allow flexibility in specifying colors, a number of formats are supported:

- Named color. Available colors are: `None` (transparent), `Black`, `Blue`, `Cyan`, `DarkGray`, `Gray`, `Green`, `LightGray`, `Magenta`, `Orange`, `Pink`, `Red`, `White`, `Yellow`
- Comma-separated Color Triplets as 0-255 (e.g., `255,0,0`) or 0.0 -1.0 (e.g., `1.0,0.0,0.0`).
- Hexadecimal: 0xRRGGBB (e.g., `0xFF0000` for red)

#### Color Tables

Color tables are simply a list of colors. Typically the symbol maintains a color table if the classification is other than `SingleSymbol`. The symbol will also keep track of unique values or class breaks and use this information to determine a color to display for a shape. A number of predefined color tables are supported but and user-defined tables is supported in the property format.

#### Symbol Style - Point Data

Symbol styles for point data are the same as for time series viewing tools. The following styles are available:

- `None`
- `Arrow-Down`, `Arrow-Left`
- `Asterisk`
- `Circle-Hollow`, `Circle-Filled`
- `Diamond-Hollow`, `Diamond-Filled`
- `Plus`, `Plus-Square`
- `Square-Hollow`, `Square-Filled`

- `Triangle-Down-Hollow`, `Triangle-Down-Filled`, `Triangle-Left-Hollow`, `Triangle-Left-Filled`, `Triangle-Right-Hollow`, `Triangle-Right-Filled`, `Triangle-Up-Hollow`, `Triangle-Up-Filled`
- `X-Cap`, `X-Diamond`, `X-Edge`, `X-Square`

### Classification

Classification is used to symbolize data. The following classifications are supported:

| Classification Type | Description |
|---|---|
| `SingleSymbol` | This is the default for all layers if not specified. For point data, a single symbol is used, centered on the . For line data, a single line width and color is used. For polygon data, single fill and outline colors are used. |
| `UniqueValues` | The data values for the field specified with the `SymbolClassField` property is sorted and checked for unique values. Each value is then assigned a color in the color table. |
| `ClassBreaks` | The number of class breaks should be one less than number of colors in the color table for the symbol. Breaks are defined by using a groupings of features based on the values of the field specified with the `SymbolClassField` property:<br><br>< first value<br>>= first value < second value<br>...<br>> last value |

## GeoView Project File Examples

This section provides several examples, extracted from GeoView Project files.

The following example illustrates how to configure base layers in a GeoView Project file:

```
# GeoView project file for Rio Grande basin.

# Main GeoView properties.

[GeoView]

# Main home for data
# If a directory is not specified, the directory will be determined when the
# GeoView project file is selected.
#GeoDataHome = "C:\cdss\statemod\data\rgtwday\gis"
# ArcView/ArcExplorer Default...
#SelectColor = Yellow
# Arc 8...
#SelectColor = Cyan
# All-purpose (magenta/pink)
SelectColor = "255,120,255"
MaximumExtent = "266400,4090475 503060,4260700"

# Now list the layer views.  A layer view consists of specifying a data layer
# (e.g., shapefile) and view information (e.g., symbol).  This is equivalent to
# the ESRI "theme" concept.  The layers specified first are drawn on the bottom.
# Start with number 1 and increase the layer number sequentially as layers are
# added on top.

[GeoLayerView 1]
GeoLayer = div3_districts.shp
Name = "Water Districts"
# RGB 153 204 50 - green-yellow
#Color = "0x99CC32"
# tan
Color = "255,240,190"
OutlineColor = black
ReferenceLayer = true
AppLayerType = "BaseLayer"

[GeoLayerView 2]
GeoLayer = div3_lakes.shp
#GeoLayer = div3_lakes.shp
Name = "Lakes"
# - blue
Color = "165,250,254"
OutlineColor = "0,130,254"
AppLayerType = "BaseLayer"

[GeoLayerView 3]
Name = "Rivers"
GeoLayer = div3_rivers.shp
#GeoLayer = div3_rivers.shp
# RGB - blue
Color = "0,188,253"
AppLayerType = "BaseLayer"

[GeoLayerView 4]
GeoLayer = div3_highways.shp
Name = "Roads and Highways"
Color = "255,0,0"
AppLayerType = "BaseLayer"

[GeoLayerView 5]
GeoLayer = div3_cities.shp
Name = "Cities and Towns"
SymbolStyle = "Square-Filled"
SymbolSize = 6
Color = "red"
LabelField = "Name"
LabelPosition = RightCenter
AppLayerType = "BaseLayer"
```

The following example illustrates how to display point data layers. These properties should be inserted at the appropriate location in a GeoView Project file.

```
[GeoLayerView 15]
#SkipLayerView = true
GeoLayer = div3_flowstations_2001-10-24.shp
Name = "Stream Gages"
# orange
Color = "254,167,0"
SymbolStyle = "Circle-Filled"
SymbolSize = 6
AppLayerType = "Streamflow"
AppJoinField = "STATION_ID"
#LabelField = "STATION_NA, STATION_NA"
#LabelFormat = "%s, %s"

[GeoLayerView 18]
#SkipLayerView = true
GeoLayer = div3_reservoirs_2001-10-24.shp
Name = "Reservoirs (WDID)"
# black
Color = "black"
SymbolStyle = "Triangle-Up-Filled"
SymbolSize = 6
AppLayerType = "Reservoir"
AppJoinField = "ID_LABEL_6"
```

# Appendix: Spatial Data Format – ESRI Shapefile

2004-05-24, Acrobat Distiller

## Overview

ESRI Shapefiles are a relatively simple format for spatial data, consisting of a file containing shape information (*.shp*), a file containing attribute data (*.dbf*), and an index file (*.shx*). The GeoView package currently supports the following shapefile shape types:

- Point (shape type 1)
- Multi-point (shape type 8)
- Arc/Line (shape type 3)
- Polygon type (shape type 5)
- Null shape (shape type 0)

Support for additional shape types may be added in the future, consistent with the shapefile specifications.

This page is intentionally blank.

## StateModGUI Binder Spine

This page, when printed, can be used for a spine in a binder.

# Colorado's Decision Support Systems (CDSS)

# StateMod Graphical User Interface (StateModGUI)

# Recommended Tabs

Version 06.03.02, 2006-03-04, Color, Acrobat Distiller

When printing the manual, it is recommended that tabbed dividers be inserted for the following document sections:

**Table of Contents**

**1. Acknowledgments**

**2. Introduction**

**3. Getting Started**

**4. Add/Delete Data**

**5. View/Edit Data**

**6. Running StateMod**

**7. Results**

**8. Using the Map**

**9. Tools**

**10. Troubleshooting**

**Installation/Configuration**

**Release Notes**

**TSView**

**GeoView**

This page is intentionally blank.