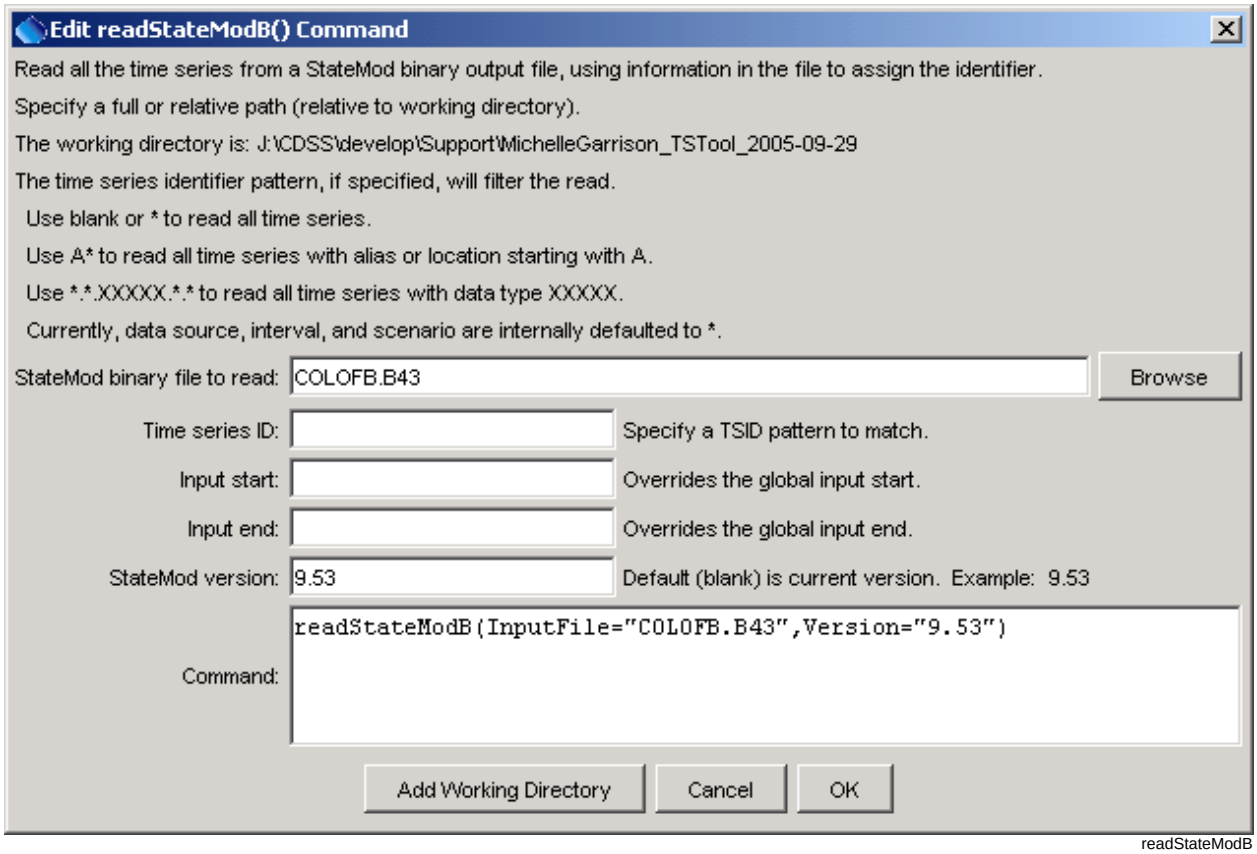
Command Reference: readStateModB()

Read time series from a StateMod binary output time series file

Version 06.10.09, 2005-09-30, Color, Acrobat Distiller

The `readStateModB()` command reads time series from a StateMod binary output time series file into memory (see the **StateModB Input Type Appendix**). The actual reading occurs as the commands are being processed. Therefore, if any other commands reference the StateMod time series, the time series identifiers must be specified manually or use wildcards in identifiers (identifiers are not available to list in dialogs like other time series specified with identifiers).

The following dialog is used to edit the command and illustrates the syntax for the command.



readStateModB

readStateModB() Command Editor

The command syntax is as follows:

```
readStateModB(param=value,…)
```

Command Parameters

Parameter	Description	Default
InputFile	The name of the StateMod binary time series file to read, surrounded by double quotes. The path to the file can be absolute or relative to the working directory. The ***Browse*** button can be used to select the file to read (if a relative path is desired, remove the leading path after the select).	None – must be specified.
TSID	Time series identifier pattern to filter the read.	Read all time series.
Version	StateMod version number corresponding to the file. The version is not included in the file and must be specified as a parameter to for older files. Changes in format occurred with version 9.01 and 9.69, mainly to add new data types.	Assume that the file format matches the current version.

The following example commands file illustrates how to read all Available_Flow time series for identifiers starting with 44 (e.g., to extract all such time series for a water district):

```
readStateModB(InputFile="..\StateMod\ym2002b.b43",TSID="44*.*.Available_Flow.*")
```

The following example illustrates how to read all time series from a binary file that was created with StateMod version 9.53 (e.g., to convert to a different format). As shown in the example, debug can be turned on for the log file to evaluate issues with the file format.

```
startLog(LogFile="commands.TSTool.log")
setDebugLevel(0,1)
readStateModB(InputFile="COLOFB.B43",Version="9.53")
```