8 Troubleshooting

Version 10.00.01, 2011-05-17

This chapter discusses how to troubleshoot TSTool problems. **Section 8.1 – Obsolete Commands** lists obsolete commands, which may no longer be supported by current software and should be phased out.

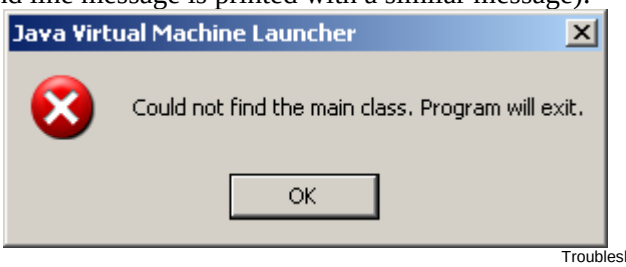
TSTool can run in interactive mode with a graphical user interface (GUI) and in batch mode. In both cases a log file contains messages from the program. By default, the log file is created in the logs directory under the main installation directory. It is recommended that the `StartLog()` command be inserted as the first command in each command file, using the name of the commands file in its name. The log file can then be viewed using the ***Tools…Diagnostics*** features (see **Chapter 3 – Getting Started**). However, in most cases, the log file should only be used for major troubleshooting because it contains technical details that may not be understandable by the user. The error-handling features of the GUI provide a status for each command. Often, an error in an early command leads to additional errors in other commands and therefore fixing the first error can resolve multiple problems.

The most common problems are program configuration (see the **Installation and Configuration Appendix**), user input error (see the commands reference for command syntax), and data errors for various input types (see below and see also the input type appendices). Other problems should be reported to the TSTool developers (use the ***Help…About*** menu to list support contacts). You may need to email the log file to support to help determine the nature of a problem.

When running the TSTool GUI, major problems will be indicated with an icon next to the offending command (see **Chapter 3 – Getting Started** for a summary of command error handling features). When running in batch mode, warnings are only printed to the log file. In either case, the log file viewer can be used to pinpoint the source of problems. If the run has been successful the GUI will show no problem indicators and the log file will contain primarily status messages, which provide useful information about data processing.

The following table summarizes common errors and their fixes. If an error is occurring in batch mode, it is useful to run via the GUI to utilize error feedback features.

TSTool Errors and Possible Solutions

Error	Possible solutions
TSTool does not run on Windows (error at start-up).	1. If using the TSTool executable on Windows and the following is shown (or a command line message is printed with a similar message):  Troubleshooting_LaunchError This error may be shown if software files have been manually moved. Reinstall using the installation program. 2. If TSTool is run on Windows using the batch file… The batch file (in the *bin* directory under the main install point) uses a command shell window that may be running out of environment space (in this case you should see a message in the command shell window to that effect). To correct, change the command shell window properties so that the initial memory is `4096` or greater. This may not take effect unless the command shell window was started from the ***Start*** menu. Additionally, to help diagnose errors, try running the *TSTool.exe* or *TSTool.bat* batch file from a command shell rather than from a desktop shortcut or Windows Explorer. Doing so may print useful messages to the command shell window.
Time series (of any data type) are not returned from the HydroBase database.	1. Verify that the database includes the water districts of interest using the ***File…Properties…HydroBase*** Information menu. HydroBase is distributed for the State of Colorado and water divisions and the version being used may be for a division. 2. Verify that the structure/station identifier is valid. For example, a USGS gage identifier may have changed. To verify, try querying by its name rather than the identifier. Also, use the CDSS StateView application to view HydroBase data details for the station/structure (it may have been renamed). 3. If a zero-filled time series file cannot be found, check its path or use the `TS alias = NewTimeSeries()` command.
A specific time series is not returned from the HydroBase database.	Time series in HydroBase are tagged with the data source (e.g., `USGS`). These data source abbreviations or their handling by software may have changed over time and a data source in a time series identifier may not be valid. Current software requires the data source for HydroBase time series, if a data source is used with the data type in HydroBase. Try interactively querying the time series to see if the data source has changed.
A data type combination is not available for queries.	TSTool has been implemented to support various input types as much as possible. However, it may not have features to view all time series in an input type. Refer to the input type appendix for limitations on data handling.
Time series (of any	TSTool queries time series by allocating memory for the requested period and

Error	Possible solutions
data type) have `-999` or other missing values.	then filling in values from the database. The output period (or maximum if not specified) may be such that time series values were not found in the database and were set to the missing data value of `-999`. Use fill commands to fill the missing data within the requested period.
TSTool fails on large queries or displays out of memory error.	TSTool may run out of memory on queries (hundreds or thousands of time series, depending on machine memory). More time series may be able to be handled if run in batch mode because GUI resources are minimized. To increase the amount of memory that TSTool will use: 1. If running on Windows using the TSTool.exe program (the default configuration), increase the value of the `-XmxNNNm` option in the *bin\TSTool.l4j.ini* file under the software installation folder. 2. If running on Windows using the *TSTool.bat* file, change the `-XmxNNNm` option after the JRE program name to tell Java to allow more memory (increase the number of MB `NNN` as appropriate for the amount of memory available on the machine – use a high number to force using hard disk swap space if desired). 3. If running on Linux or Unix using the *tstool* script, change the `-XmxNNNm` option after the JRE program name to tell Java to allow more memory (increase the number of MB `NNN` as appropriate for the amount of memory available on the machine – use a high number to force using hard disk swap space if desired).
Unexpected failure.	If there was an error in input that was serious, TSTool may quit processing input. See the log file for details. If the log file does not offer insight, contact support. Specific causes of failure may include: 1. TSTool has been developed using a version of Java that is indicated in the metadata for software files. Trying to use an older Java version may cause unexpected errors. This will only be a problem in custom installations where the default Java distributed with TSTool is not used. To determine the Java version that is being used to run TSTool and that was used to create the software, use ***Tools…Diagnostics*** and select the ***Allow debug*** checkbox. Then use the ***Help…About TSTool*** menu item and press ***Show Software/System Details*** to display information that includes the Java version that is being used to run TSTool. 2. An unforeseen issue may be occurring. Contact support. You may need to provide the data and command file being used, which will allow troubleshooting and also allow developers to add additional tests to be run before software is released.
Unable to find files correctly.	The working directory is assumed to be the same as the location of the most recently opened or saved command file. The current working directory is generally displayed by editor dialogs that use a path and can also be displayed using ***File…Properties…TSTool Session***. If files are not being found, verify that the path to the file is correct, whether specified as an absolute path or relative to the command file.
When used with the HydroBase input type for a Microsoft Access database, an error occurs	1. TSTool tries to list the ODBC DSN that are available for HydroBase databases. It does so by running the *shellcon.exe* program, typically installed in *\cdss\bin* if TSTool is used with CDSS. If this directory is not in the `PATH` environment variable, the program will not be found and an error will occur. The `PATH` normally will include the directory after

Error	Possible solutions
selecting the HydroBase database. Microsoft Access versions of HydroBase have not been used for years so this problem is unlikely.	installation, in order to allow TSTool to be run from directories other than the installation directory. The `PATH` can be checked by opening a command shell and typing **`path`** at the command prompt. If the `PATH` is not properly set, edit it as follows: • For Windows NT/2000/XP machines, add *\CDSS\bin* (or *\Program Files\RTi\RiverTrak\bin*) to the `PATH` using the ***Settings...Control Panel...System...Advanced...Environment Variables*** settings. You may need to have the administrator perform this step. • For Windows 95/98 machines, add *\CDSS\bin* (or *\Program Files\RTi\RiverTrak\bin*) to the `PATH` in the *autoexec.bat* file. Reboot to apply for all subsequent windows. You may have done this previously. A work-around is to manually type in the ODBC DSN in the entry field. 2. Only user ODBC DSNs are listed when selecting HydroBase. If the DSN was defined as a system DSN, it will not be listed. Redefine the DSN as a user DSN.
TSTool is unable to load HEC-DSS DLL files and therefore HEC-DSS features are unavailable.	The HEC-DSS library files are distributed in the *TSTool-Version/bin* folder and by default this is where TSTool looks for the files. If the start folder for TSTool is changed from this folder, the files will not be found. Therefore, do not reconfigure TSTool to start in other than the *bin* folder.

8.1 Obsolete Commands

TSTool and the commands that it supports have evolved over time. In early versions, many commands used syntax similar to the following:

```
-SomeCommand parameter
```

Later, the function notation with fixed parameter list was adopted:

```
someCommand(parameter1,parameter2)
```

Parameters for such commands were required to be in a specific order and enhancements were difficult to implement because the parameter order needed to be maintained. Subsequent enhancements added new commands and converted older commands to a new free-format "named parameter" notation:

```
SomeCommand(Param1=Value1,Param2=Value2)
```

The new notation allows parameters to be omitted when using a default value, and allows new parameters to be added to commands, as necessary, to enhance existing functionality. The above syntax is now standard throughout TSTool. Support for the older notation is provided where possible.

Prior to TSTool version 10, some commands used the syntax:

```
TS Alias = Command(…)
```

In version 10, this syntax has been made similar to all other commands:

```
Command(Alias="….",…)
```

In most cases, loading an old command file will automatically convert from old to new syntax. TSTool provides warnings for commands that are not recognized or are out of date and cannot automatically be updated –the command editor can be used to correct errors.

The following table lists obsolete commands. The `TSID` abbreviation, when inside parentheses for a command, is interchangeable with the time series alias.

TSTool Command Summary – Obsolete Commands

Command	Description	Replacement
`add(TSID,TSID1, TSID2,...)`	Add the 2nd+ time series to the first time series, retaining the original identifier. This form of the command is obsolete and should be updated to use the new form described that includes a flag for handling missing data.	Current `Add()`.
`-archive_dbhost HostName`	This option is normally set during installation and is typically not specified in command files. Specify the Internet host name for the remote HydroBase database server. This is configured at installation time and will be either "localpc" (for a local Microsoft Access HydroBase database, indicating that no remote server is used) or a machine name for the Informix database server. To change the defaults from those in the *tstool.bat* file, specify this option again on the command line or edit the batch file. See also `-dbhost`. This option is used in addition to the `-dbhost` information to allow a TSTool user to switch between the local PC and the main database server.	`OpenHydroBase()` and configuration information.
`-averageperiod MM/YYYY MM/YYYY`	Specify the period to be used to compute averages when the `-fillhistave` option is specified.	`SetAveragePeriod()`
`-batch`	Indicates to run in batch mode. This is automatically set if `-commands` is specified.	None – no longer used.
`-browser Path`	This option is normally set during installation and is typically not specified in command files. Specify the path to the web browser to use for on-line documentation.	None – no longer used.
`CreateTraces()`	Create an ensemble from a time series.	`CreateEnsemble()`
`-cy`	Output in calendar year format.	`SetOutputYearType()`
`-d#[,#]`	Set the debug level. The first number is the debug level for the screen. The second is for the log file. If one level is specified, it is applied to the screen and log file output.	`SetDebugLevel()`
`-data_interval Interval`	Indicate the data interval (e.g., MONTH, DAY) to use with all structures/stations indicated by the `-slist` option. See the appendices for a list of intervals for different input and data types. This option is only available in batch mode.	`CreateFromList()`
`-datasource ODBCDataSourceName`	Specify an ODBC Data Source Name to use for the HydroBase database.	`OpenHydroBase()` and configuration information.
`-data_type Type`	Indicate the data type (e.g., DivTotal, DQME) to use with all structures/stations indicated by the `-slist` option. This option is only available in batch mode. This command is obsolete.	`CreateFromList()`

TSTool Command Summary – Obsolete Commands (continued)

Command	Description	Replacement
`day_to_month_reservoir (TSID,ndays,flag)`	Read a daily time series and convert to a monthly time series using the reservoir method. This is generally only applied to reservoir storage.	`NewEndOfMonthTS FromDayTS()` and `FillInterpolate()`
`-dbhost HostName`	This option is normally set during installation and is typically not specified in command files. Specify the Internet host name for the primary HydroBase database server. This is configured at installation time and will be either `localpc` (for a local Microsoft Access database) or a machine name for the Informix database server. To change the defaults from those in the tstool.bat file, specify this option again on the command line or edit the batch file.	`OpenHydroBase()` and configuration information.
`-detailedheader`	Insert time series creation information in output headers. This preserves information from the log file that may otherwise be lost. The default is not to generate detailed headers.	See output command parameters to control.
`fillCarryForward()`	Fill by repeating value.	`FillRepeat()`
`fillconst (TSID, Value)`	Fill the time series with a constant value.	`FillConstant()`
`-fillData File`	Specify a StateMod format fill pattern file to be used with the `fillpattern()` command. This command can be repeated for multiple pattern files.	`SetPatternFile()`
`-fillhistave`	Currently only enabled for frost dates and monthly data. Indicates that the time series should be filled with the historical average values from the output period where data are missing (after filling by other methods). See also the `-averageperiod` option.	`FillHistMonthAverage()` and `FillHistYearAverage()`
`Graph g = newGraph(GraphType, Visibility, TimeSeriesToGraph)`	Create a new graph window.	This command is no longer supported. See `ProcessTSProduct()`.
`-helpindex Path`	This option is normally set during installation and is typically not specified in command files. Specify the path to help index file for on-line documentation.	No longer used.
`-ignorelezero`	Treat data values <= `0` as missing when computing averages but do not replace when filling.	`SetIgnoreLEZero()`
`-include_missing_ts`	If a time series cannot be found, include an empty time series.	`SetIncludeMissingTS()`

TSTool Command Summary – Obsolete Commands (continued)

Command	Description	Replacement
`-informix`	Indicate that Informix is used for HydroBase.	Not used.
`-missing Value`	Use the specified value for missing data values (StateMod only). The default is `-999.0`.	See `WriteStateMod()`.
`-fillusingcomments`	This option only applies to diversion time series and causes the diversion comments to be evaluated. Comments that indicate no diversion in an irrigation year will result in missing data for that year being replaced with zeros.	`FillUseDiversionComments()`
`month1/year1 month2/year2`	Specifies beginning and ending months for period of record - calculations are still based on the entire period of record (i.e., regression values) but the final output is according to these values, if given. Month 1 is January. Years are 4-digit.	`SetOutputPeriod()`
`-o outputfile`	Specify output file name. This is used in conjunction with other `-o` options.	`Write*()` commands.
`-odatevalue`	Output a DateValue format file.	`WriteDateValue()`
`-ostatemod`	Output a StateMod format file.	`WriteStateMod()`
`-osummary`	Output a time series summary.	`WriteSummary()`
`-osummarynostats`	Output a time series summary without statistics (this is used with the data extension procedure developed by Ayres for CDSS).	No longer supported.
`regress(TSID1,TSID2)`	Performs a linear regression analysis between the two time series, filling missing data of the first time series. Regression information is printed to the log file.	`FillRegression()`
`regress12(TSID1, TSID2)` `regressMonthly( TSID1,TSID2)`	Same as `regress()` except 12 separate monthly regressions values are calculated.	`FillRegression()`
`regresslog(TSID1, TSID2)`	Same as `regress()` except regressions values are calculated logarithmically.	`FillRegression()`
`regresslog12(TSID1, TSID2)` `regressMonthlyLog( TSID1,TSID2)`	Same as `regresslog()` except 12 monthly regressions values are calculated.	`FillRegression()`
`setconst(TSID,Value)`	Set the time series to the given value for all data. If the time series is not in the database, created an empty time series and then set to a constant value.	`SetConstant()`
`setconstbefore(TSID, Value,Date)`	The time series to the given value for all data on and before the specified date (`YYYY-MM or MM/YYYY`).	`SetConstant()`
`setConstantBefore()`	Set a value constant before a date/time.	`SetConstant()`
`SetMissingDataValue()`	Set the missing data value used in a StateMod time series.	See `WriteStateMod()`.

TSTool Command Summary – Obsolete Commands (continued)

Command	Description	Replacement
`setQueryPeriod( Start,End)`	Set the global period to query databases and read from files.	`SetInputPeriod()`
`-sqlserver`	Specify that SQL Server is used for HydroBase.	`OpenHydroBase()` and configuration information. SQL Server is also now the default because Microsoft Access is no longer supported.
`-slist File`	Create time series from a list file.	`CreateFromList()`
`-units value`	Output using the specified units (default is to use database units).	No longer used. If necessary, units can be converted by a number of commands including `ConvertUnits()`.
`-w#[,#]`	Set the warning level. The first number is the warning level for the screen. The second is for the log file. If one level is specified, it is applied to the screen and log file output.	`SetWarningLevel()`
`-wy`	Output in water year format.	`SetOutputYearType()`

This page is intentionally blank.