# 8 Troubleshooting

Version 07.00.00, 2006-11-22, Color, Acrobat Distiller

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.

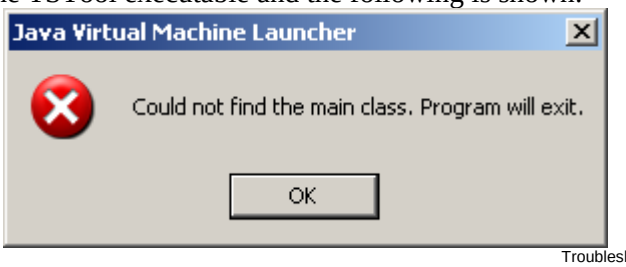
TSTool runs in both graphical user interface (GUI) and batch mode. In both cases a log file contains messages from the program. When the GUI is used, the log file is created in the logs directory under the main installation directory (e.g., *C:\CDSS\logs\tstool.log* for a CDSS installation or *C:\Program Files\RTi\RiverTrak\logs\TSTool.log* for an RTi RiverTrak® system installation). 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 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 (see **Chapter 1 - Acknowledgements** for support contacts). Code has been implemented to detect common errors, but you may need to refer to the log file to determine the nature of a problem.

In general, when running the TSTool GUI, you will be warned about major problems using pop-up dialogs. When running in batch mode, warnings are only printed to the log file. In either case, the log file can be viewed. The log file viewer can be used to pinpoint the source of warnings. If the run has been successful, you will only see status messages in the log file. Status messages provide useful information (such as regression results).

The following table summarizes common errors and their fixes. **Because database designs change over time, if an error is occurring in a batch run, you may want to interactively query the time series because the interface enforces conventions that are compatible with current database designs and it may be easier to diagnose errors interactively**.

**TSTool Errors and Possible Solutions**

| Error | Possible solutions |
|---|---|
| TSTool does not run on Windows (error at start-up). | 1. If using the TSTool executable and the following is shown:<br>  <br><br>Troubleshooting_LaunchError<br>This error may be shown if running the software from the command line after an installation. Reboot the computer so that installation settings are fully recognized.<br>2. If TSTool is run 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.<br><br>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. |
| When used with the HydroBase input type for a Microsoft Access database, an error occurs selecting the HydroBase database. | 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 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:<br><br>• 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.<br>• 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.<br><br>A work-around is to manually type in the ODBC DSN in the entry field.<br>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 Errors and Possible Solutions (continued)**

| Error | Possible solutions |
|---|---|
| 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.<br><br>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.<br><br>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 re-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. For HydroBase, the CDSS StateView application has additional displays. |
| Time series (of any data type) have `-999` values. | TSTool queries time series by allocating memory for the requested period and 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`. You can use fill options to fill the missing data within the requested period. |
| TSTool fails on large queries. | TSTool may run out of memory on queries (hundreds or thousands of time series, depending on machine memory). More time series can be handled if run in batch mode. In the *tstool.bat* file, change the `-XmxNNm` option after the JRE program name to tell Java to allow more memory (increase the number of MB `NN` 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:<br>1. TSTool has been developed using Java 1.4.2, which is referenced in the startup batch file or command line. Trying to use a different Java version may cause unexpected errors. To determine the Java version that is being used with TSTool, 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. |
| Unable to find files correctly. | TSTool uses Java technology, which at this time does not allow the working directory to be changed internally. The working directory is assumed to be the same as the location of the commands file. Verify that the working directory is as expected using ***File…Properties…TSTool Session***. |

## 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 was adopted:

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

However, 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. Recent enhancements have added new commands and converted some 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. Old commands will be transitioned to the new syntax until all commands have been updated. Support for the older notation is provided where possible. In most cases, editing an old command with the command editor dialogs will convert from old to new syntax automatically.

The following table lists obsolete commands. Although TSTool tries to run old commands as much as possible, some obsolete commands are not recognized. Old commands should be updated to the new commands, if possible. TSTool will print warnings for commands that are not recognized or are out of date.

The `TSID` abbreviation, when inside parentheses for a command, is interchangeable with the time series alias.

**TSTool Command Summary – Obsolete Commands**

| Command | Description |
|---|---|
| `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. |
| `-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. |
| `-averageperiod MM/YYYY MM/YYYY` | Specify the period to be used to compute averages when the `-fillhistave` option is specified. **This command is obsolete and should be converted to `setAveragePeriod()`.** |
| `-batch` | Indicates to run in batch mode. This is automatically set if `-commands` is specified. This command is no longer needed. |
| `-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. This command is no longer needed. |
| `-cy` | Output in calendar year format. This command is obsolete and should be replaced with `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. This command is obsolete and should be replaced with `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. This command is obsolete. Use the `createFromList()` command. |
| `-datasource ODBCDataSourceName` | Specify an ODBC Data Source Name to use for the HydroBase database. In commands files, this option is obsolete and should be replaced with `setDataSource()`. |
| `-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. Use the `createFromList()` command. |

**TSTool Command Summary – Obsolete Commands (continued)**

| Command | Description |
|---|---|
| `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. This command has been replaced with the `TS X = newEndOfMonthTSFromDayTS()` command, optionally followed by a `fillInterpolate()` command. |
| `-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. See also `-archive_dbhost`. In command files, this command is obsolete and should be replaced with `setDatabaseHost()`. |
| `-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. This command is obsolete and should be replaced with `setOutputDetailedHeaders()`. |
| `fillconst (TSID, Value)` | Fill the time series with a constant value. This command is obsolete and has been replaced by `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. This command is obsolete and should be replaced by `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. This command is obsolete and should be replaced by `fillHistMonthAverage()` and `fillHistYearAverage()`. |
| `Graph g = newGraph(GraphType, Visibility, TimeSeriesToGraph)` | Create a new graph window. This command is no longer supported. |
| `-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. |
| `-ignorelezero` | Treat data values <= `0` as missing when computing averages but do not replace when filling. This command is obsolete and should be replaced with `setIgnoreLEZero()`. |
| `-include_missing_ts` | If a time series cannot be found, include an empty time series. This command is obsolete and should be replaced with `setIncludeMissingTS()`. |

**TSTool Command Summary – Obsolete Commands (continued)**

| Command | Description |
|---|---|
| `-informix` | Use `setDatabaseEngine()`. |
| `-missing Value` | Use the specified value for missing data values (StateMod only). The default is `-999.0`. This command is obsolete and should be replaced by `setMissingDataValue()`. |
| `-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. This command is obsolete and should be replaced with `setUseDiversionComments()`. |
| `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. This command is obsolete and should be replaced by `setOutputPeriod()`. |
| `-o outputfile` | Specify output file name. This is used in conjunction with other `-o` options. This option is obsolete and should be replaced with `writeStateMod()` or other output commands. |
| `-odatevalue` | Output a DateValue format file. This option is obsolete and should be replaced with `writeDateValue()`. |
| `-ostatemod` | Output a StateMod format file. This option is obsolete and should be replaced with `writeStateMod()` or other output commands. |
| `-osummary` | Output a time series summary. This command is obsolete. There is currently not a replacement other than using the GUI. |
| `-osummarynostats` | Output a time series summary without statistics (this is used with the data extension procedure developed by Ayres for CDSS). This command is obsolete. There is currently not a replacement other than using the GUI. |
| `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. This command is obsolete and should be replaced with `fillRegression()`. |
| `regress12(TSID1, TSID2)` `regressMonthly( TSID1,TSID2)` | Same as `regress()` except 12 separate monthly regressions values are calculated. This command is obsolete and should be replaced with `fillRegression()`. |
| `regresslog(TSID1, TSID2)` | Same as `regress()` except regressions values are calculated logarithmically. This command is obsolete and should be replaced with `fillRegression()`. |
| `regresslog12(TSID1, TSID2)` `regressMonthlyLog( TSID1,TSID2)` | Same as `regresslog()` except 12 monthly regressions values are calculated. This command is obsolete and should be replaced with `fillRegression()`. |

**TSTool Command Summary – Obsolete Commands (continued)**

| Command | Description |
|---|---|
| `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. This command is obsolete and should be replaced with `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`). This command is obsolete and should be replaced with `setConstantBefore()`. |
| `setQueryPeriod( Start,End)` | Set the global period to query databases and read from files. This command has been replaced by the `setInputPeriod()` command. |
| `-sqlserver` | This command is obsolete and should be replaced by `setDatabaseEngine()`. |
| `-slist File` | Create time series from a list file. This command is obsolete and should be replaced by `createFromList()`. |
| `-units value` | Output using the specified units (default is to use database units). This command is not active. |
| `-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. This command is obsolete and should be replaced with `setWarningLevel()`. |
| `-wy` | Output in water year format. This command is obsolete and should be replaced with `setOutputYearType()`. |