TEAMCENTER

Teamcenter Environment Variables

Teamcenter 2312

SIEMENS

About Siemens Digital Industries Software

Siemens Digital Industries Software is a global leader in the growing field of product lifecycle management (PLM), manufacturing operations management (MOM), and electronic design automation (EDA) software, hardware, and services. Siemens works with more than 100,000 customers, leading the digitalization of their planning and manufacturing processes. At Siemens Digital Industries Software, we blur the boundaries between industry domains by integrating the virtual and physical, hardware and software, design and manufacturing worlds. With the rapid pace of innovation, digitalization is no longer tomorrow’s idea. We take what the future promises tomorrow and make it real for our customers today. Where today meets tomorrow. Our culture encourages creativity, welcomes fresh thinking and focuses on growth, so our people, our business, and our customers can achieve their full potential.

Support Center: **support.sw.siemens.com**

Send Feedback on Documentation: **support.sw.siemens.com/doc_feedback_form**

Contents

1. Getting started

Introduction to environment variables

This manual is intended for system administrators, data administrators, and Java developers who configure the Teamcenter functionality using environment settings. A basic understanding of object-oriented programming (OOP) and Teamcenter concepts is required.

Use environment variables and script files to configure Teamcenter. Environment settings allow you to configure and control many aspects of the Teamcenter software. By setting these variables for each site, Teamcenter can be quickly adapted and configured to meet any site's needs. Many of these variables point to specific network nodes and/or directories so that Teamcenter can be used in large and small homogeneous and heterogeneous environments.

What are environment variables?

Much of Teamcenter behavior and display can be configured with environment variables.

Environment variables are settings stored in the **tc_profilevars** properties file that allow you to control many aspects of Teamcenter behavior. This script is run directly by Bourne and Korn shells. C shells must use the **tc_cshvars** script.

Environment variables are run in script files to configure and control many aspects of Teamcenter. By setting these variables for each site, Teamcenter can be quickly adapted and configured to meet any site's needs. Many of these variables point to specific network nodes and/or directories so that Teamcenter can be used in large and small homogeneous and heterogeneous environments.

Manually configure the Teamcenter environment

Before you deploy Teamcenter, you should fully complete the configuration process.

Some configuration tasks, such as defining log files, are difficult to redefine once Teamcenter is deployed.

Teamcenter administrators typically configure site workstations and computers so that users can log on without manually setting the environment. If this has not been done, you must manually set the Teamcenter environment before you can run Teamcenter clients or utilities.

Only the **TC_ROOT** and **TC_DATA** environment variables must be set to run Teamcenter Foundation. These variables can be set automatically at logon. However, some stand-alone utilities such as **install** and **clearlocks** require that the entire Teamcenter environment be set.

Linux systems

To manually configure the Teamcenter environment on Linux systems, source the **tc_profilevars** and **tc_cshvars** scripts, inserting the appropriate paths to *TC_DATA* for your environment:

- **Bourne/Korn shell:**

  ```
  . TC_DATA/tc_profilevars
  ```

- **C shell:**

  ```
  source TC_DATA/tc_cshvars
  ```

These scripts create a subshell in which Teamcenter environment variables are set.

Windows systems

Configure your environment for Teamcenter by one of the following methods:

- Open a Teamcenter command prompt by choosing **Start→All programs→Teamcenter 14→***Tc-config-name* **Command Prompt**.
- Run the **tc_profilevars.bat** script:

  1. Open a Windows command prompt.
  2. Type the following command:

     ```
     call TC_DATA\tc_profilevars
     ```

     For example:

     call c:\Siemens\Teamcenter\tcdata\tc_profilevars

  3. If you use Teamcenter Integration for NX, enter the following additional commands:

     ```
     set UGII_BASE_DIR=NX-installation-path
     set UGII_ROOT_DIR=%UGII_BASE_DIR%\ugii
     set PATH=%UGII_ROOT_DIR%;%PATH%
     ```

> Note:
>
> These procedures use Linux command syntax except where otherwise specified. On Windows systems, modify command syntax as follows:

- Path syntax: Replace forward slashes (**/**) with backslashes (****).
- Environment variable names: Replace **$** prefix characters with leading and trailing **%** characters:

Linux	Windows
$*variable-name*	**%***variable-name***%**

Syntax definitions

This manual uses a set of conventions to define the syntax of Teamcenter environment variables. Following is a sample syntax format:

PUBLISH_PDF_ERROR_LOG_LOC

This environment variable is used with Visualization Illustration. Defines a location for the Convert/Print error log file different from the default location (**c:\Documents and Setting***user-name***\Local Settings\Temp\VisProd***location***\VVCP_error.log**).

This log file records all errors occurring during the PDF conversion process. Set this environment variable with the full path name and file name. For example:

```
c:\MyFiles\Error.log
```

The conventions are:

Bold	Bold text represents words and symbols you must enter exactly as shown. In the preceding example, you enter **c:\Documents and Setting**, **\Local Settings\Temp\VisProd** and **\VVCP_error.log** exactly as shown.
Italic	Italic text represents values that you supply. In the preceding example, you supply values for *user-name* and *location*.
`Courier font`	This font, indented, indicates an example of a setting or command you can input. In the preceding example, you could enter: `c:\MyFiles\Error.log`

2. Working with environment variables

Environment variable script files

Teamcenter environment variables are stored in the **tc_profilevars** script file, stored in the *TC_DATA* directory. This script can be run directly by Bourne and Korn shells. C shells must use the **tc_cshvars** script. When the **tc_cshvars** script is sourced from a C shell, environment variables are read from the **tc_profilevars** script file. This is done so that a single file (**tc_profilevars**) can be used to store environment variable settings for an entire site.

System environment variables are stored as defined by your operating system.

Setting environment variables for logging

Teamcenter saves information in various log files. View log files to identify problems by showing the state of an application at any given point.

For more information about log files, see *System Administration*.

You can modify Teamcenter logging behavior by setting environment variables. Set the following environment variables to configure logging information sent to system log and journal files.

Environment variable name	Use
API_JOURNAL	Enables Teamcenter Integration for NX journaling when set to **FULL**.
TC_JOURNAL	Accepts the following values: **OFF** Creates an empty journal file. **SUMMARY** Creates a summary report listing the total time spent on each journal routine. **FULL** Creates a full report listing each call. **MODULE** Uses the **TC_Journal_Modules** preference to select which modules should be journaled. **TC_Journal_Modules** accepts multiple strings as values. Each value must be a valid Teamcenter application module. Set to **FULL** to optimize logging information. Set to **MODULE** to fine tune logging information.
TC_JOURNAL_LINE_LIMIT	Limits the number lines in the journal file; the default is 30,000 lines. Set to **0** for no limit.

Environment variable name	Use
TC_JOURNAL_PERFORMANCE_ONLY	Collects performance information only if set to **TRUE** (any value). Also set **TC_JOURNAL=FULL**, **TC_JOURNAL_LINE_LIMIT=0**. When set on the server manager, set to **2** to enable performance journaling for the entire pool. This setting has no effect on the running environment unless journaling is triggered.
TC_KEEP_SYSTEM_LOG	Accepts the following values: **Y** The system log is not deleted, even if there are no errors. **N** The system log is deleted if there are no errors. Set to **Y** to retain the system log.
TC_POM_JOURNALLING	Determines whether the POM module journals nested calls. Set to **N** for full journaling.
TC_SLOW_SQL	Writes explain plans to the system log file, if SQL statements take longer to run than the time specified. Set to a number of seconds. Defaults to **10** seconds. If set to zero, no explain plans are written. If set to a negative value, explain plans are written if run time exceeds 1000 seconds.
TC_SQL_DEBUG	Accepts the following values: **J** Copies all SQL statements to the journal file as well as the system log file. **P** Includes profiling data; reports the frequency of use of each distinct statement in a summary at the end of log file. **T** Includes timing data; reports the time of each SQL statement. **B** Binds variables; expands SQL bind variables (this is the default setting). **b** Reports bind variables in :1 format, rather than expanding them. Set to **JPT** to optimize logging information.
TC_TRACEBACK	Accepts the following values: **ON** Writes tracebacks to the system log file.

Environment variable name	Use
	OFF Does not write tracebacks to the system log file. When errors occur, the system writes tracebacks to the system log file by default. Use this environment variable to override the default behavior.

Environment variables listing

The following list of Teamcenter environment variables includes a brief description of each variable and how the variable is set (for example, through the **tc_profilevars** property file).

AdaptiveApplicationDisplay

Determines whether the adaptive application display functionality is enabled. When enabled, icons of applications that have not been used in a certain period of time do not display in the navigation pane, thus rarely used applications are hidden. Users can click the **More...** button at the bottom of the navigation pane, which displays the rest of the application icons. The period of disuse required to trigger this functionality is set in the AdaptiveApplicationDisplayDuration variable in the **client_specific.properties** file.

AdaptiveApplicationDisplayDuration

Determines the period of time, in days, an application must not be used to trigger the adaptive application display functionality. You must set the AdaptiveApplicationDisplay variable for this environment variable to be implemented. Set this variable in the **client_specific.properties** file.

AM_BYPASS

Bypasses Access Manager when logging on to Teamcenter enabling a system administrator to reload the AM rule tree. Use this environment variable if the rule tree needs repair.

> Example:
>
> Perhaps an unintended consequence of modifying a rule results in no longer being able to see your home folder, effectively preventing you from logging on. In this case, set this environment variable to any value, which prompts the system to bypass the AM rule tree when logging on.

AM_DEBUG

Logs the AM rule tree applicable to an object in the **syslog** file. Any non-NULL value enables this environment variable.

AM_MODE

Evaluates the user's group value based on a union of all groups to which the user belongs. This includes groups defined by predefined conditions and by access control lists (ACLs). When this environment variable is not set, a user's group is determined by the group value with which the user logged on. Enable this functionality by setting this environment variable with any value.

AM_PERFORMANCE_STATISTICS

Collects and displays Access Manager performance statistics for each call to a rule or accessor function, which are useful for debugging performance issues with rule and accessor implementations or the rule tree. The statistics are logged to the **syslog** file at server shutdown.

Enable this functionality by setting this environment variable with any value.

> Note:
>
> Because there is a significant performance impact to collect the statistics, the feature is disabled by default.

For more information about the statistics collected, see *Access Management Using Rules and ACLs*.

API_ALWAYS_REFRESH

Determines whether the **Refresh Object** caching method is enabled. This method caches an array of refreshed objects during each system operation, such as loading an assembly. Setting this environment variable to any value disables this caching method.

API_DEFAULT_DIR

Determines the download location for files associated with a part. The default NX startup directory is used if this environment variable is not set.

API_JOURNAL

Enables journaling of API files. Set to **FULL** to log all API modules.

API_SET_BYPASS

Enables DBA bypass of the ACL tree when using import utilities, such as **import_file**, when set to **true**.

AUTHORIZATION_MODE

Evaluates all of a user's group and role-in-group settings to determine the effective authorization rules for the user. If left unset, the system evaluates only the current logon settings.

AUX_PATH

(Optional) Provides auxiliary path information required by applications launched from the rich client. Add the **AUX_PATH** variable in the value set for the PATH variable. Using the **AUX_PATH** variable reduces the size of the overall **PATH** value by excluding the existing system values from the final path values used for rich client startup. This can improve startup performance.

Set the **AUX_PATH** environment variable prior to starting the rich client. For example:

- Windows systems

```
set AUX_PATH=C:\new\path;%AUX_PATH%
```

- Linux systems (using **ksh**)

```
export AUX_PATH=/new/path:$AUX_PATH
```

BMF_BYPASS_ALL_EXTENSION_RULES

Determines whether to bypass processing all the extension rules defined through the Business Modeler IDE when a particular operation is executed. Use this environment variable only for installations and upgrades. When set to **ON**, the system bypasses processing of all defined extension rules. When set to **OFF**, the system does not bypass processing of defined extension rules. The default value is **OFF**.

> Example:
>
> During the upgrade process, Teamcenter Environment Manager (TEM) sets this environment variable to **ON** to bypass processing of extension rules because processing the rules might cause an upgrade failure.
>
> You can also use this environment variable to set this behavior.

> Note:
>
> System administration privileges are required to implement this setting. If the user is not logged on with system administration privileges, extension rules will not be bypassed, regardless of the environment variable setting.

BMIDE_SCRIPT_ARGS

Sets the memory allocated to the Java heap sizing. For example, set its value to **-Xmx1024M** to allocate 1 GB of RAM to the Java heap space on the server (assuming you have that much memory space available). Create this environment on a corporate server to ensure there is enough memory to install large templates. You can also set this value on a machine running a Business Modeler IDE client to ensure there is enough memory to load templates containing a large amount of data model.

For more information, see *BMIDE for Data Model Design*.

BYPASS_RULES

Determines whether users logged on as a member of the system administrator group may bypass processing all business rules, including naming rules. Use this environment variable only for installations and upgrades. When set to **ON**, processing of all business rules is bypassed. When set to **OFF**, all business rules are processed. The default value is **OFF**.

> Example:
>
> During the upgrade process, the Teamcenter Environment Manager (TEM) can set this environment variable to **ON** to bypass processing of business rules. A naming rule validation could cause failure if TEM is looking for a specified object name but cannot locate it because a naming rule changes the object name.

CLASSPATH

Defines the directory for storing the rich client object log (**.log**) files when the **java.io.tmpdir** key is defined in the Virtual Machine **CLASSPATH** variable, as follows:

`-Djavaio.tmpdir=`*path-to-temp-directory*

This environment variable may contain multiple paths, delimited by semicolons (**;**).

> Caution:
>
> If a path in the **CLASSPATH** value contains whitespace characters, those paths must be enclosed in double quotes (**"**). For example:
>
> **"C:\Program Files\Microsoft\Web Platform Installer";D:\TcSE\apache-ant-1.9.4\bin**

DEPLOY_ARCHIVE_ACTIVE

Archives deployment files.

By default, this environment variable is not defined and considered turned on. Use this environment variable to turn off deployment archiving by setting the value to **false** (or **off**, **no**, or **0**). By default, the **deploy_archive** utility is automatically run when you deploy from Teamcenter Environment Manager (TEM) and the Business Modeler IDE.

For more information, see *BMIDE for Data Model Design*.

FCC_PROXYPIPENAME

Defines the base name of the set of FIFOs (pipes) used to communicate with the **FCCClientProxy**, for example:

set FCC_PROXYPIPENAME=\\.\pipe\FMSClientPipe|/tmp/FMSClientPipe

The value to the left of **|** is used for Windows hosts, and the value to the right of **|** is used for Linux hosts.

The value must exactly match the value of the **FCC_ProxyPipeName** FCC configuration parameter in the **fcc.xml** file.

For more information, see *System Administration*.

> Note:
>
> Siemens Digital Industries Software recommends you do not set this parameter except under certain circumstances where it may be required.
>
> For more information, contact your Siemens Digital Industries Software representative.

FMS_FCCSTARTUPLOG

Enables logging of FCC data. Set to the location (full path and file name) in which you want the data stored.

FMS_HOME

Specifies the FMS home directory. For server side, **FMS_HOME** specifies the home directory for FSC files, and on the client machine, **FMS_HOME** specifies the home directory for FCC files.

FMS_WINDOWS_MULTIUSER

Allows multiple users of the same Windows client machine to run separate and independent instances of the FMS client cache (FCC). Set the environment variable to **true** to enable multiple users to use the same FMS installation or set to **false** to run in legacy single-user mode. If set to **true**, each user launches their own instance of FCC and has their own FCC cache directory.

For more information, see *System Administration*.

FSC_HOME

Specifies the path where an FMS server cache (FSC) service can find the configuration files. Typically, this is set to *TC_ROOT***fsc**. Set this environment variable through the *TC_DATA*/**tc_profilevars** property file.

IMAN_SDL_MAGIC

Dumps a property table into the output page with any error generated while accessing an object property. This table shows the name type and value of each valid property of the relevant object.

> Note:
>
> Do not use this facility in production.

IMAN_SYS2_UID_DIR

Defines the directory containing the default uncustomized Teamcenter Integration for NX display libraries.

JAVA_HOME

Specifies the path to the Java location. Typically, this environment variable is set through the *TC_DATA*/**tc_profilevars** property file. In a run-time environment, this environment variable points to the same location as the JDK_HOME environment variable. In a development environment, set this environment variable to point to the same location as the JDK_HOME environment variable.

JDK_HOME

Specifies the path to the Java Development Kit (JDK) location. JDK is required for Teamcenter customization.

JRE_HOME

Specifies the path to the Java Runtime Environment (JRE) location. Typically, this environment variable is set through the *TC_DATA*/**tc_profilevars** property file.

> Note:
>
> You must download and install the JRE prior to using any of these Teamcenter components. This applies to all the platforms for Teamcenter: Windows and Linux.

JRE64_HOME

Specifies the path to the 64-bit version of the Java Runtime Environment (JRE). This environment variable is required if you have a 64-bit system and you run 64-bit applications that require a JRE.

> Note:
>
> You must download and install the JRE prior to using either of these Teamcenter components. This applies to all the platforms for Teamcenter: Windows and Linux.

LANG

Defines non-English locales on Linux systems. Specify the system locale by setting this environment variable and the LC_ALL environment variable using the Common Desktop Environment (CDE). These two variables must be identical to function properly.

Use with the MB_CHARS environment variable to support localizing Teamcenter by modifying string and date processing behavior to better conform to local usage. Set this environment variable to a valid locale. Display the list of locales by typing **locale -a** in a shell.

For more information about country code values, see *Teamcenter Installation on Linux Using TEM*.

LC_ALL

Defines non-English locales on Linux systems. Specify the system locale by setting this environment variable and the LANG environment variable using the Common Desktop Environment (CDE). These two variables must be identical to function properly.

For more information about country code values, see *Teamcenter Installation on Linux Using TEM*.

LD_LIBRARY_PATH

Defines the shared display library path for Oracle Solaris platforms. The *TC_DATA***/tc_profilevars** script searches for shared display libraries, constructs a path statement, and exports **LD_LIBRARY_PATH**.

LIBPATH

Defines the shared display library path for the IBM platform only. The *TC_DATA***/tc_profilevars** script searches for shared display libraries, constructs a path statement, and exports **LIBPATH**.

LOG_CONFIG_LOCATION

Specifies the custom **log4j** file to use for TCCS logging. When not using a custom **log4j** file, the *FMS_HOME***\log4j2.xml** file is used (as specified by the **LogConfigLocation** parameter in the *FMS_HOME***\log.properties** file).

For more information, see *System Administration*.

LOG_VOLUME_LOCATION

Specifies the location of the TCCS log volume as defined in the **starttccs.bat** file. By default, it is set to the *USERPROFILE***Siemens\logs\tccs** directory.

For more information, see *System Administration*.

MB_CHARS

Sets localized dataset naming for languages using character sets other than US ASCII. This environment variable is normally not set (enabling a single-byte environment); if set to **1**, a multibyte environment is enabled. When a multibyte environment is enabled, Teamcenter supports dataset names containing Katakana, Kanji, Hiragana, and extended ASCII characters. This is used for internationalizing Teamcenter.

MGC_HOME

Used with the Mentor Graphics Integration only. Defines the directory where Siemens EDA is installed.

MGC_LOCATION_MAP

Used with the Mentor Graphics Integration only. Defines the full path (directory and file name) of the Siemens EDA location map file.

MGC_TC_CONFIG_FILE

Used with the Mentor Graphics Integration only. Defines the full path (directory and file name) of the Mentor Graphics Integration configuration file.

MGLS_LICENSE_FILE

Used with the Mentor Graphics Integration only. Defines the full path (directory and file name) of the Siemens EDA license file.

MOZ_PLUGIN_PATH

Set when using Security Services on a Linux platform. To log into a Security Services-enabled environment on a Linux platform, set this environment variable to a JRE 1.5 (or later) plug-in path. For example:

/admin/java/Solaris_JRE_1.5.0_11/plugin/sparc/ns7

NLS_SORT

Specifies how data coming back from an Oracle server is sorted. This environment variable is set through the *TC_DATA***/tc_profilevars** property file.

NR_BYPASS

Determines whether users logged in as a member of the system administrator group may bypass processing of naming rules. Use this environment variable only for installations and upgrades. When set to **ON**, processing of naming rules is bypassed. When set to **OFF**, all naming rules are processed. The default value is **OFF**.

Example:

During the upgrade process, the Teamcenter Environment Manager (TEM) can set this environment variable to **ON** to bypass processing of naming rules. A naming rule validation could cause failure if TEM is looking for a specified object name but cannot locate it because a naming rule changes the object name.

NXMGR_ALLOW_3_TIER_INTEROP

Determines whether Teamcenter Integration for NX integrates with Teamcenter in three-tier mode. Setting this variable allows Teamcenter Integration for NX to work in three-tier mode.

ORACLE_HOME

Specifies the directory where the Oracle software is installed.

ORACLE_SERVER

Defines the Oracle server network node. This protected variable is set during Teamcenter installation and added to the *TC_DATA***/tc_profilevars** directory.

ORACLE_SID

Defines the unique name of the Oracle database instance associated with *TC_DATA*. This protected variable is set during Teamcenter installation and added to the *TC_DATA***/tc_profilevars** file.

PATH

Defines the *TC_DATA***/tc_profilevars** script and appends the operating system **PATH** environment variable with the full path of the *TC_ROOT***/bin** directory.

You can use this environment variable with the AUX_PATH environment variable.

PLUGIN_LOGFILE

Points to a plug-in message log file for Acrobat Reader.

- Error: `Not an ARF file.`

  Used by **File Open**: unrecognizable input file format.

- Error: `Not Tool XML file.`

  Used by **File Open**: unrecognizable input file format.

- Error: `PDF file` *filename* `not found`

  Used by **File Open**: cannot find based PDF file.

- Error: `FDF file` *filename* `not found.`

  Used by **File Open**: cannot find based markup FDF file.

- Error: `Unable to open file` *filename*.

  Used by **File Open**: cannot access to the displayed file.

- Warning: `The selected file(s) have attached Markups. But cannot be displayed in the Reader because the based PDF is not comment-right enabled.`

  Used by **File Open by Reader**: The plug-in on the Reader requires the based PDF to have a **comment-right enable** token embedded in the PDF file for user to view and perform markup using Reader.

For more information about logging, see *System Administration*.

POM_PURGE_AGE_LIMIT

Specifies the period in seconds for which unreferenced or unlabeled historical revisions are protected from purging. Older historical revisions are considered for purging. Enter a negative value to disable purging, for example, **-1**.

POM_READ_EXPR_DB_REGEN_INTERVAL

Specifies (in minutes) when the database statistics are regenerated during daemon execution.

The default value is **30**.

POM_SCHEMA

Defines the **.ull** file specification (directory path and file name) of the POM schema file. Protected variable.

By default, the POM schema file is located in the *TC_DATA* directory. The file name is very long and conforms to the following format:

`pom_schema_`*ORACLE_SERVER_ORACLE_SID*

where *ORACLE_SERVER* is the Oracle server network node, and *ORACLE_SID* is the unique name of the Oracle database instance.

POM_TRANSMIT_DIR

Specifies the directory path and file name of the POM transmit schema file. This is a protected variable and requires a separate POM transmit schema file for 64-bit and 32-bit platforms.

By default, the POM transmit schema file is located in the *TC_ROOT*/**pom_transmit** directory. The file name is a concatenation of the site ID and schema version UID.

portalCGI

Specifies the path to the rich client CGI executable. For example:

portalCGI=tc/launchapp

This environment variable is set in the **site_specific.properties** file.

portalWebServer

Specifies the web server on which the rich client CGI executable resides. For example:

portalWebServer=http://*Web-server***/**

This environment variable is set in the **site_specific.properties** file.

PUBLISH_PDF_ERROR_LOG_LOC

Used with Visualization Illustration. Defines a location for the Convert/Print error log file different from the default location (**c:\Documents and Setting***user-name***\Local Settings\Temp\VisProd***xxx\123***\VVCP_error.log**).

This log file records all errors occurring during the PDF conversion process. Set this environment variable with the full path name and file name. For example:

c:\MyFiles\Error.log

PUBLISH_PDF_ERRORS

Used with Visualization Illustration. Determines whether the **Convert** and **Print** dialog boxes close when an error occurs. The default value (**yes**) closes the dialog boxes when an error occurs. Set to **no** for the dialog boxes to remain open after an error occurs.

PUBLISH_PDF_INTERVAL

Used with Visualization Illustration. Determines the amount of time (in milliseconds) before the system issues each subsequent PDF conversion. Use this setting to prevent conflicts between simultaneous PDF jobs. This situation occurs when publishing a portfolio to PDF using the **multi PDF** option. The default setting is **1000**.

PUBLISH_PDF_MAX_JOBS

Used with Visualization Illustration. Determines the maximum number of outstanding PDF conversion jobs that can be run simultaneously. This situation occurs when publishing a portfolio to PDF using the **multi PDF** option. The default setting is **10**.

PUBLISH_PDF_TIMEOUT

Used with Visualization Illustration. Determines the amount of time (in milliseconds) allotted for each PDF conversion to complete. The default setting is **INFINITE**.

PUBLISH_PDF_VVCP_DEBUG

Used with Visualization Illustration. Determines whether a Convert/Print log is generated during PDF conversion. This setting is used for debugging. Because enabling logging greatly increases the PDF conversion time, enable this only while debugging. Set to **on** to enable the logging. The default setting is **off**.

Note:

This environment variable works in conjunction with the **PUBLISH_PDF_VVCP_DEBUG_LOC** environment variable. You must set both environment variables to enable logging.

PUBLISH_PDF_VVCP_DEBUG_LOC

Used with Visualization Illustration. Determines the location of the Convert/Print log, which is used for debugging PDF conversion. Because enabling logging greatly increases the PDF conversion time, enable this only while debugging. Set this environment variable with the full path and file name. For example:

c:\Myfiles\VVCP.log

Note:

This environment variable works in conjunction with the **PUBLISH_PDF_VVCP_DEBUG** environment variable. You must set both environment variables to enable logging.

PUBLISH_PDF_VVCP_TIMEOUT

Used with Visualization Illustration. Determines the amount of time (in seconds) of the time-out for convert and print jobs. The default setting (**600**) is typically adequate for PDF conversions. Set the value higher if, for example, a particular large, single PDF conversion job is failing due to a completion time-out.

ROSE

Defines the directory containing the STEP Translator run-time application files. Set this environment variable in the *TC_DATA***/tc_profilevars** property file; the path must always point to the *TC_DATA* directory.

Note:

The STEP system fails if the path pointing to the *TC_DATA* directory contains drive letters.

ROSE_DB

Defines the directory containing the STEP Translator run-time data files. This environment variable is set through the *TC_DATA***/tc_profilevars** property file; it must always point to the *TC_DATA* directory.

Note:

The STEP system fails if the path pointing to the *TC_DATA* directory contains drive letters.

SIEMENS_LOGGING_ROOT

Specifies a custom base log file location for logs from business logic servers, the server manager, and the web tier.

SITCONS_AUTH_KEY

Specifies a Site Consolidation license key value. You must set the key to authorize the desired Site Consolidation utilities.

For more information, see *Teamcenter Site Consolidation*.

> Note:
>
> The license key value is available from Support Center:
>
> **https://support.sw.siemens.com/**.

SPLM_LICENSE_SERVER

Specifies the location of the default local license server.

The default setting is:

port@host

Replace *port* with the port number and *host* with the host name of the license server, for example, **28000@tchost**. The *port* and *host* values must match those you specified in the **SERVER** line of the Teamcenter license file.

> Note:
>
> During Teamcenter upgrade, this value is designated as the default local license server.

TAO_ROOT

Defines the directory in which the TAO object request broker (ORB) is installed.

TC_ALLOW_INCOMPATIBLE_TYPES

Multi-Site checks for compatible schema during an import from a site that is using a different version of Teamcenter from the importing site. An import fails if incompatible types are found, even if the export file does not contain instances of incompatible objects.

Set this environment variable at the importing site to any value to suppress incompatible error messages, write a warning to the **syslog** file, and continue the import process. Delete this variable after all databases are upgraded to same release.

> Note:
>
> Although any value suppresses the error and continues the import, for clarity, you should set the value to **true**.

TC_BIN

Defines the directory containing Teamcenter binary files. This protected variable is set during Teamcenter installation and added to the *TC_DATA***/tc_profilevars** property file.

TC_BYPASS_CANNED_METHODS

[Deprecated as of Teamcenter 13.1] Bypasses the defined canned method. For example, various methods may need to be bypassed during a legacy import.

When a session is started with this environment variable set, the methods defined are bypassed for all types. Valid values are:

All

Bypass all methods for all types.

method_1

Bypass **method_1** for all types that have **method_1** configured.

method_1, method_2

Bypass **method_1** and **method_2** for all types that have **method_1** and **method_2** configured.

TCCS_CONFIG

Sets the name of a directory (under the **TCCS_CONFIG_HOME** location) containing TCCS configuration files. This environment variable is required only when a custom TCCS configuration directory is used.

If not defined, the default value is **Teamcenter**.

For more information, see *System Administration*.

TCCS_CONFIG_HOME

Sets the path to the directory containing TCCS configuration directories. This environment variable is required only when a custom TCCS configuration directory is used.

If not defined, the default values for TCCS_CONFIG_HOME are as follows:

For this OS	If not defined, the default value for TCCS_CONFIG_HOME is this
Windows	For a private installation: **%USERPROFILE%\Siemens\cfg\tccs** For a shared installation: **%ALLUSERSPROFILE%\Siemens\cfg\tccs**
Linux	For a private installation: **$HOME/Siemens/cfg/tccs** For a shared installation: **/etc/Siemens/cfg/tccs**

For information, see *System Administration*.

TCCS_JAVA

Defines the JRE to use for TCCS. Typically it is set in the **starttccs.bat** file to point to the JAVA_HOME or JRE_HOME environment variables.

For information, see *System Administration*.

TC_DATA

Defines the directory containing Teamcenter data files. This protected variable is set during Teamcenter installation and added to the *TC_DATA***/tc_profilevars** property file.

TC_DATA_MODEL

Specifies the directory where data model templates are installed. By default, this is set to *TC_DATA***/model**. This environment variable is set through the *TC_DATA***/tc_profilevars** property file.

TC_DB_CONNECT

Accesses Oracle database tables by logging in to the Oracle database as user **infodba** using a partially encrypted connection string passed to the Oracle database by Teamcenter. The corresponding password is encrypted for security reasons. This protected variable is set during Teamcenter installation and added to the *TC_DATA***/tc_profilevars** property file.

> Warning:
>
> Because this environment variable is partially encrypted, it cannot be set manually. If it becomes corrupted or you forget the password, it must be regenerated using the Teamcenter Environment Manager (TEM).

TC_FS_SERVICE

Associated with File Management System. This environment variable is set through the *TC_DATA***/tc_profilevars** property file.

TC_HANDLERS_DEBUG

Determines the amount of workflow handler information sent to the system log file. Set to **ALL** to send debug information on all workflow handlers to the system log file. Set to one or more specific workflow handler names to send debug information on the specified workflow handlers to the system log file. If left unset, no workflow handler debug information is logged.

TC_HELP_FILE

Defines the directory containing Teamcenter online help files. This variable is set during Teamcenter installation and added to the *TC_DATA***/tc_profilevars** property file.

TC_HTDOCS

Specifies the directory to hold dynamic files. By default, it is set to *TC_ROOT***\web\htdoc**. This environment variable is set through the *TC_DATA***/tc_profilevars** property file.

TC_INCLUDE

Defines the directory containing custom ITK header files. Set this protected environment variable to access custom ITK programs. This environment variable is set through the *TC_DATA***/tc_profilevars** property file.

TC_Installation_Logging

Enables or suppresses installation logging for the entire site. This log file, which contains a record of activities performed by the Teamcenter Environment Manager (TEM) installation program, is in the install directory under the application root directory. The date-time stamp represents the date and time TEM was run. For example, **install0522241627.log** indicates that TEM was run at 4:27 on February 24, 2005.

If set to **ON**, administration logging is enabled. The default value is **OFF**.

TC_INSTALL_DIR

Specifies the directory for Teamcenter installation files. This environment variable is set through the *TC_DATA*/**tc_profilevars** property file.

TC_JOURNAL

Accepts the following values:

OFF	Creates an empty journal file.
SUMMARY	Creates a summary report listing the total time spent on each journal routine.
FULL	Creates a full report listing each call.
MODULE	Uses the **TC_Journal_Modules** preference to select which modules should be journaled. **TC_Journal_Modules** accepts multiple strings as values. Each value must be a valid Teamcenter application module.

Set to **FULL** to optimize logging information.

Set to **MODULE** to fine tune logging information.

TC_JOURNAL_LINE_LIMIT

Limits the number lines in the journal file; the default is 30,000 lines. Set to **0** for no limit.

TC_JOURNAL_PERFORMANCE_ONLY

Enables performance journaling for the entire pool when set to **2** on the pool server manager. This setting has no effect on the running environment unless journaling is triggered.

Setting this environment variable generates **.pjl** placeholder files in the pool server temp directories. These placeholder files are of negligible size unless journaling for a process has been triggered.

TC_KEEP_SYSTEM_LOG

Controls the deletion of the system log file.

Accepts the following values:

Y	The system log is not deleted, even if there are no errors.

N The system log is deleted if there are no errors.

Set to **Y** to retain the system log.

TC_language_default

Defines the default language for the Teamcenter server process.

Teamcenter allows users to select the locale on their clients, regardless of the locale used by the Teamcenter server pool manager. The only constraints are that requested locales are properly installed on the server side (which may not be true for customized locales) and that the server-side system can cope with the locale encoding.

Use this environment variable to define the default locale in the event that no locale is passed; for example, no locale is passed from services-oriented architecture (SOA). The value defines the internal Teamcenter representation associated with a given language.

Valid values are a single string containing one of the following country codes indicating the desired locale or the name of the directory where your Teamcenter server customized localization files are located.

Accepts the following values:

cs_CZ Czech

de_DE German

en_US English (default value)

es_ES Spanish

fr_FR French

it_IT Italian

ja_JP Japanese

ko_KR Korean

pl_PL Polish

pt_BR Portuguese (Brazilian)

ru_RU Russian

zh_CN Chinese (as spoken in China)

zh_TW Chinese (as spoken in Taiwan)

TC_LIBRARY

Sets the platform-specific shared display library path. This environment variable is set to the LD_LIBRARY_PATH, LIB_PATH environment variable.

TC_LOG

Defines the directory containing system log files created during Teamcenter sessions. This directory is created during Teamcenter installation and must not be removed. If this directory is removed, system log files cannot be created. This environment variable is set through the *TC_DATA*/**tc_profilevars** property file.

The default setting is:

TC_DATA/log_*ORACLE_SERVER*_*ORACLE_SID*

where *ORACLE_SERVER* is the Oracle server network node, and *ORACLE_SID* is the unique name of the Oracle database instance.

TC_LOG_DIR

Specifies the directory into which error files, log files, and output files are placed.

TC_LOG_VOLUME_DIR

Specifies the directory into which error files, log files, and output files are placed.

TC_LOGGER_CONFIGURATION

Specifies the directory containing the **logger.properties** file. Default loggers are created with the log levels defined in this properties file.

Specify the directory path or the whole file path to the properties file.

For information, see *System Administration*.

TC_LOG_VOLUME_DIR

Specifies the location of the log volume where the logs are transferred. This environment variable is set through the *TC_DATA*/**tc_profilevars** property file.

TC_LOG_VOLUME_NAME

Specifies the name of the log volume where the logs are transferred. This environment variable is set through the *TC_DATA*/**tc_profilevars** property file.

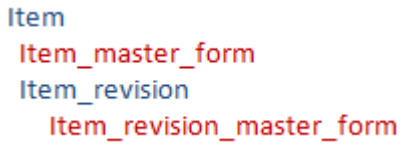
TC_LOV_SHARED_MEMORY_SAFETY_FACTOR

Specifies the safety factor to control memory allocation for the LOV shared memory feature. The default value is **3**.

TC_MASTERFORM_DELEGATE

Sets how the Master forms inherit their access privileges.

When an item is created in Teamcenter, the following Master form objects are created:



- If this environment variable is not set (which is the default), the Item Master form (**Item_master_form**) inherits its privileges from its parent item (**Item**), and the Item Revision Master form (**Item_revision_master_form**) inherits its access privileges from its parent item revision (**Item_revision**).

- If this environment variable is set, the Item Master form and the Item Revision Master form derive their privileges through Access Manager rules evaluation independent of their parent item or item revision.

> Note:
>
> By default, this environment variable is *not* set. Setting it to any nonnull value turns off the Master form access privileges delegation. This environment variable is set through the *TC_DATA***/tc_profilevars** property file.

TC_MFK_INDEX_KEY_SIZE

Specifies the byte size limit of the index key size when creating the key in **POM_KEY** table. If the actual index key byte size exceeds the limit, the application returns a failure. You can set it before evaluating or deploying multifield key definitions based on the database platform limitation. The default value is 900.

For more information, see *BMIDE for Data Model Design*.

TC_MSG_ROOT

Specifies the path to where utilities can locate localized error messages. Typically, this is set to *TC_ROOT***/lang/textserver/**. This environment variable is set through the *TC_DATA***/tc_profilevars** property file.

TC_no_obsolete_message

Setting this environment variable to any value suppresses ITK warning messages.

TC_NO_TEXTSRV_SHARED_MEMORY

Disables shared memory functionality and reverts system behavior to in-process text storage when the variable is set to **TRUE**. Use this environment variable when the TC_SHARED_MEMORY_DIR environment variable is already set.

For more information, see *System Administration*.

TC_NX_DEFAULT_PART

Defines the default NX part file. This environment variable is set through the *TC_DATA***/tc_profilevars** property file.

TC_ONLINE_HELP

Specifies whether online help is installed. This environment variable is set through the *TC_DATA*/**tc_profilevars** property file.

TC_PERFORMANCE_MONITOR

Automatically displays the **Performance Monitor** dialog box in the rich client after logon and enables SQL tracking statistics.

TC_POM_JOURNALLING

Determines whether the POM module journals nested calls.

Set to **N** for full journaling.

TC_POM_SPONSORED_USER

Specifies the sponsored user. Set to the user ID of the sponsored user.

TC_PRINTER

Defines the default Teamcenter printer. Protected variable. To access other printers, use the **Workspace System→Printer** command.

TC_ROOT

Top (root) directory of the Teamcenter directory structure. This environment variable must be set before running the application or command line utilities.

For more information, see Teamcenter Utilities.

TC_SHARED_MEMORY_DIR

Specifies the directory where the memory backing store files are stored. If you do not set this environment variable, the **TEMP** environment variable (Windows) or **/tmp** (Linux) directory is used. On Windows, if the **TEMP** environment variable is not available, the **C:\temp** directory is used.

For more information about this environment variable, see *System Administration*.

TC_SKIP_CLIENT_CACHE

Causes the rich client to run in legacy mode and avoid downloading or using a local client meta cache. Set this environment to any value to activate it, for example, **TC_SKIP_CLIENT_CACHE=1**. To unset the variable, leave the value blank, for example, **TC_SKIP_CLIENT_CACHE=**.

The following features provide additional client caching functionality:

- **generate_client_meta_cache** utility

  Caches metadata to improve client interaction with the server. Metadata is cached and refreshed automatically, but the cache can be regenerated using this utility. The utility is usually run during installation and upgrade.

  For more information, see *Teamcenter Utilities*.

Note:

Run this utility after a style sheet is edited to keep the style sheet feature in the client cache folder up-to-date. You can also use this utility to generate the full client cache folder if for some reason the client cache folder has been deleted.

For more information, see *Rich Client Customization*.

- **ClientCache** folder

  Stores cached metadata. If you are logged on to the rich client as a database administrator (DBA), you see the folder under the **Home** location.

  Warning:

  If the administrator moves, renames, or deletes this folder, all clients fall back to a nonclient-cached mode.

- **Generate Client Cache** check box

  Runs the **generate_client_meta_cache** utility. The check box is added to Teamcenter Environment Manager (TEM).

  For more information, see *Teamcenter Environment Manager — Usage*.

- AUX_PATH environment variable

  Adds auxiliary path information required by applications launched from the rich client.

TC_SKIP_REV_PARTITION

Turns off database partitioning, when set to **Y**. To allow database partitioning, set it to **N**.

POM-revisable tables (used by 4GD) use database partitioning for historical data automatically during upgrade or install based on the feature availability with the database. Partitioning helps query performance where large volume of historical data is present.

Even though database partitioning is available and working on Oracle Enterprise, it requires a license at extra cost. By default, database partitioning is switched on. If you want to switch it off, set this environment variable to **Y** before installation or upgrade.

TC_SLOW_SQL

Writes explain plans to the system log file for SQL statements that run longer than the specified time. The explain plans can be used to suggest indexes that can be added to speed up the query. Set the environment variable to a number of seconds. The specified value is interpreted as a floating point time in seconds. If it is set to zero (**0**), explain plans are not written. If set to a negative value, explain plans are written if the SQL statements exceed 1000 seconds. The time is real time, not CPU time. It only generates explain plans for the first 10 cases. Defaults to 10 seconds.

TC_SQL_DEBUG

Accepts the following values:

J	Copies all SQL statements to the journal file as well as the system log file.
P	Includes profiling data; reports the frequency of use of each distinct statement in a summary at the end of log file.
T	Includes timing data; reports the time of each SQL statement.
B	Binds variables; expands SQL bind variables (this is the default setting).
b	Reports bind variables in :1 format, rather than expanding them.

> Note:
>
> Set to **JPT** to optimize logging information.

TCSS_DEBUG_LEVEL

Enables debug output to *'%appdata%***\teamcenter\sso\debug.log** file for the .NET library. It accepts the following case insensitive values:

- **true**
- **false**
- **debug**
- **info**
- **warn**
- **error**
- **fatal**

If this variable is not defined, Teamcenter sets the level to **warn**. If the variable is defined with no value or and invalid value, Teamcenter sets the level to **debug**. This behavior is enforced by the **setDebug** method in Java and the **Debug** property in C#.

TC_SSO_APP_ID

Identifies which installation is authenticated. Use this environment variable when multiple Teamcenter sites are served by a single identity provider, or Security Services is configured to use an ID other than Teamcenter (this is the default when the **TC_SSO_APP_ID** environment variable is not set). When the multiple sites have different sets of users authorized to use the application for each installation, or different identifications in each installation, you can identify which installation is authenticated.

Set this environment variable in both the *TC_DATA***/tc_profilevars** file for the server and the client's **site_specific.properties** file.

> Note:
>
> Set this environment variable only if Security Services is enabled using the TC_SSO_SERVICE environment variable.

TC_SSO_CHANGE_PASSWORD_PAGE

Determines the URL to which the client redirects the change password page by setting the complete URL for the local change password page provided using the identity provider used with Security Services. This environment variable is optional, and when available, is set through the *TC_DATA*/**tc_profilevars** file.

> Note:
>
> Set this environment variable only if Security Services is enabled using the TC_SSO_SERVICE environment variable.

TCSSO_HARD_LOGIN_TIMEOUT_SECONDS

Specifies the amount of time (in seconds) to display the consent logon banner on which the user must click the **I Accept** button. The default setting is **150**.

TC_SSO_LOGIN_PAGE

Determines the URL to which the client redirects the logon by setting the complete URL for the Security Services logon page. This environment variable is set through the *TC_DATA*/**tc_profilevars** file.

> Note:
>
> Set this environment variable only if Security Services is enabled using the TC_SSO_SERVICE environment variable.

TCSSO_LOGIN_SERVICE_URL

Enables the WebSEAL feature within clients. Set this variable to the URL address of the Security Services Login Service as configured for the WebSEAL proxy server.

TC_SSO_LOGIN_URL

Directs the rich client to use the Security Services application client library to obtain an **appToken** from the **SSO** applet. If the applet is not already running, a browser is launched on the given URL to start a Security Services session. This environment variable is set in the rich client's properties file. For the two-tier rich client, the file is the **client_specific.properties** file. For the four-tier rich client, the file is the **site_specific.properties** file.

> Note:
>
> Set this environment variable only if Security Services is enabled using the TC_SSO_SERVICE environment variable.

TCSSO_SAM_AUTH_ZONE

Specifies an IdP zone that is registered with SAM Auth, which is used to authenticate users.

TC_SSO_SERVICE

Enables Security Services functionality by setting the complete URL the ITK server uses to communicate with Security Services. This environment variable is set through the *TC_DATA*/**tc_profilevars** file.

In the rich client and ITK, the ability to set or change a password in the Teamcenter database is disabled when this environment variable is set. The **Change Password** command remains available from the **Actions** menu. In the Organization application, the password text field and blank password check box are disabled.

TC_TMP_DIR

Specifies where the utilities should create temporary files. The value must be a full directory path on a local computer, not a network or UNC path. This value is typically **C:\Temp** on Windows and **/tmp** on Linux.

This environment variable is set through the *TC_DATA*/**tc_profilevars** property file.

TC_TRACEBACK

Controls the writing of tracebacks to the system log file.

Accepts the following values:

ON	Writes tracebacks to the system log file.
OFF	Does not write tracebacks to the system log file.

> Note:
>
> When errors occur, the system writes tracebacks to the system log file by default. Use this environment variable to override the default behavior.

TC_USE_DEFAULT_EXPORT_SITES

Specifies whether to use the **TC_default_export_sites** preference. Set to **YES** to enable the preference.

TC_USE_LOV_SHARED_MEMORY

Determines whether the Teamcenter server loads localized LOV display names into shared memory. Set this environment variable to either **TRUE** or **FALSE**. The default setting is **TRUE**.

If shared memory cannot be used, the system uses process memory. In rare instances when the system reports a problem with shared memory and cannot fall back to using process memory, set this environment variable to **FALSE** and restart the system.

Because localized LOV display can slow system performance, you can use the **Fnd0LOVDisplayAsEnabled** global constant to disable loading localized LOV display names.

TC_USE_METADATA_SHARED_MEMORY

Determines whether the Teamcenter server uses metadata shared with other Teamcenter servers or uses metadata local to its own process. The default setting is **TRUE**. Set the environment variable to **FALSE** to run the Teamcenter servers in local memory mode.

Using local metadata causes an increase in memory footprints for Teamcenter servers.

TC_USE_PREFS_SHARED_MEMORY

Determines whether the Teamcenter server loads preferences into shared memory. Set this environment variable to either **TRUE** or **FALSE**. The default setting is **FALSE**.

> Note:
>
> Siemens Digital Industries Software recommends using the default settings.

TC_USE_REV_PARTITIONING

Specifies database partitioning. Set this environment to **Y** to do database partitioning for minor revisable classes. Set it to **N** to skip database partitioning for minor revisable classes.

POM-revisable tables (used by 4GD) use database partitioning for historical data automatically during upgrade or install based on the feature availability with the database. Partitioning helps query performance where a large volume of historical data is present.

Even though database partitioning is available and working on Oracle Enterprise, it requires a license at extra cost. Therefore, by default this environment variable is set to **N**. To enable partitioning for 4GD data, you set this environment variable to **Y** before installation or upgrade, or execute the **install** utility with the **-revisioning PARTITION** argument.

TC_USER_LIB

Contains user-defined custom libraries. On Windows systems, the *TC_DATA***/tc_profilevars** file adds this directory to the **PATH** variable. On Linux systems, if this variable is set in the environment, the **tc_profilevars** file adds this directory to the **LD_LIBRARY_PATH** or **LIB_PATH** settings.

TC_USER_MSG_DIR

Points to the directory that contains custom XML text and error message files.

The directory to which this variable points must have a subfolder matching the value of the TC_language_default preference, for example:

TC_USER_MSG_DIR=*path-name/language_locale*

Replace *language_locale* with the language subdirectory name, for example:

TC_USER_MSG_DIR=C:\my_custom_errors\en_US

TC_USER_TOOLS

Contains user-defined software applications. Protected variable. This environment variable is set through the *TC_DATA***/tc_profilevars** property file.

TC_USE_TEST_ATTR_MAPPINGS

Determines whether Teamcenter references the test attribute mappings rather than the real mappings. Set this environment variable to any value. This causes Teamcenter to access the test mappings rather than the real mappings.

> Note:
>
> Remove **TC_USE_TEST_ATTR_MAPPINGS** from the Teamcenter environment when test attribute mapping should not be used.

TCVIS_ALSG_ENABLED

Disables support for the viewing of JtSimplification datasets in the Lifecycle Viewer and the stand-alone viewers, such as Mockup. By default, this variable is not set and **JtSimplification** is automatically enabled in properly licensed viewers. Set this variable to **False** to disable the functionality.

TC_WEB_NO_CACHE

Optimizes performance by caching TcScript parse trees if set to **FALSE**. Avoids the need to restart the **TcServer** process after changes if set to **TRUE**. Set to **TRUE** while modifying TcScript code and then set back to **FALSE** after you complete your changes.

TC_XML_ENCODING

Includes templates from other languages. On the server side, this environment variable controls the encoding environment for the XML file generated. The first line in the XML file must specify the encoding required.

TEAMCENTER_SSL_CERT_FILE

Specifies the path to the SSL certificate authority (CA) file for Teamcenter applications. Set this environment variable in the *TC_DATA***/tc_profilevars** property file, immediately after the TC_DATA variable setting, as follows:

```
TEAMCENTER_SSL_CERT_FILE=${TC_DATA}/pom_transmit/ca-bundle.crt;
export TEAMCENTER_SSL_CERT_FILE
```

In this example, the CA file (**ca-bundle.crt**) is located under the **pom_transmit** directory.

TEMP_FILESEARCHPATH

Sets the temporary path to use for Teamcenter file searches. This environment variable is set through the *TC_DATA***/tc_profilevars** property file.

TNS_ADMIN

Identifies the path where the **tnsnames.ora** file and **sqlora.net** files are located. These files are used when communicating with an Oracle server. This environment variable is set through the *TC_DATA***/tc_profilevars** property file.

Transient_Volume_Installation_Location

Specifies the node name of the transient volume. It is a location-based logical identifier and is generally set to the host name of the local machine by the *TC_DATA***/tc_profilevars** script. This environment variable overrides the preference of the same name. When a four-tier transient volume is mounted on all Teamcenter server hosts at a given location, use this setting to configure the distributed transient volume.

TRAVERSAL_THRESHOLD

Specifies the maximum number of objects to be traversed. If the maximum number of objects to be traversed is more than is specified in the variable, traversal may be incomplete and therefore the results are incomplete.

For more information about traversal, see *Tc XML and PLM XML Configuration for Data Import and Export*.

UGII_BASE_DIR

Defines the topmost directory of the NX directory structure, the NX equivalent of *TC_ROOT*. This environment variable is set by NX.

UGII_CHECKING_LEVEL

Enables server checking when it is set to **1**. (Set to **0** to disable.) When checking is enabled, the system uses the *TC_DATA***\logger.debug.properties** file for logging instead of the **logger.properties** file. The default settings of the debug file generate useful debugging messages.

For more information, see *System Administration*.

UGII_OPTION

Determines how to interpret and convert file specifications. This environment variable can drastically affect Teamcenter File Services and other Teamcenter behavior. For example, if you create Teamcenter volumes with mixed-case names and this preference is set to **LOWER**, Teamcenter File Services is not able to find your volumes. Therefore, ensure that this environment variable is set consistent with file specification naming conventions at your site.

This environment variable accepts the following values:

- **NONE**

File specifications are interpreted without converting case.

- **LOWER**

  Converts the entire file specification (directory and file name) to all lowercase.

- **UPPER**

  Converts the entire file specification (directory and file name) to all uppercase.

- **HINAME**

  Converts the file name to all uppercase but leaves the directory path name unchanged.

- **LONAME**

  Converts the file name to all lowercase but leaves the directory path name unchanged.

UIL_INCLUDE

Specifies a semicolon-separated list of directories where **.uih** files to be included are located.

XFILESEARCHPATH

Defines the directory containing X-Window System resource definition files. Protected variable. This environment variable is set through the *TC_DATA***/tc_profilevars** property file. If not set, X-Window System uses the **HOME** directory.