



digital  
vision  
group

# Environment Configuration Setup

TRIPsystem  
Product Documentation

## End User License Agreement

All rights to this software, its documentation and logotypes of the TRIP product family and software (altogether "Software") supplied by DVG Operations GmbH (DVG) are exclusively owned by DVG.

The transfer of this Software, solutions or parts thereof requires the prior written agreement of DVG. Furthermore, the customer has the right to use licensed Software and / or process solutions supplied by DVG to the extent specified in his contract with DVG.

The free-to-use non-commercial version doesn't require a prior written agreement with DVG but such customers, organizations and/or third parties agree by using the software and / or solution of DVG to be strongly obliged to keep all rights to this software, documentation and logotypes of the TRIP product family absolutely uninfringed and protected.



# Table of Contents

<b>Introduction .....</b>	<b>8</b>
About this Document.....	8
The Configuration File tdbb.conf .....	9
Location of tdbb.conf .....	9
Configuration File Lookup on Windows.....	9
Configuration File Lookup on UNIX.....	9
Installation Procedure Considerations .....	9
Batch Setup.....	10
Printer Queues and Printer Control Files .....	10
Specifying Non-Printable Characters .....	11
More About Translation Tables .....	11
Logical Names .....	13
UNIX .....	13
Windows .....	13
<b>TRIPsystem Logical Names Reference (TDBS_).....</b>	<b>14</b>
ACCDIR .....	14
ACCESS_TOKEN_EXPIRATION .....	15
ACCFLG.....	16
APILOG .....	17
AUDIT .....	18
ASELIBS .....	19
AUTH_PROVIDER.....	20
AUTO_SAVE.....	21
BAFFIT_SECURITY.....	22
BAFFRE_TIMEOUT .....	23
BASES .....	24
BLOCK_SIZE .....	25
BOLD_COLOR (Windows only) .....	26
BUT_LOCATION.....	27
CHARS.....	28
CHIVOC .....	29
CLS .....	30

CODEPAGE (Windows TRIPclassic only) .....	31
COM .....	32
CONFLATOR_LANG .....	33
CONFLATORS .....	34
CTL .....	35
DEFATTR (UNIX only) .....	36
DELETE_MAX .....	37
DEMO .....	38
DISALLOW_GUEST .....	39
DISPLAY_FREQ .....	40
DISPLAY_ORIG .....	41
EDIT (TRIPclassic only) .....	42
ENTER_OS (TRIPclassic only) .....	43
ERRMAILST (UNIX only) .....	44
EXE .....	45
EXEC_SCRIPT .....	46
FIND_TIMEOUT .....	47
FIRST_WEEKDAY .....	48
FLOCK_TIMEOUT .....	49
GLBUPD_OPEN_DB_ONLY .....	50
GRAPH_LOG .....	51
GRAPH_RENUM .....	52
HIGHLIGHT .....	53
HOME .....	54
IMPERSONATE_GROUP (UNIX only) .....	55
LANG .....	56
LDAP_ANONYMOUS .....	57
LDAP_AUTOUSER .....	58
LDAP_AUTOUSER_GROUP .....	59
LDAP_AUTOUSER_PWSYNC .....	60
LDAP_AUTOUSER_UM .....	61
LDAP_BASE .....	62
LDAP_CACERT (UNIX only) .....	63
LDAP_CERT_DIR (UNIX only) .....	64
LDAP_CERT_TRUSTALL .....	65

LDAP_ERRORLOG .....	66
LDAP_FALLBACK .....	67
LDAP_MATCH .....	68
LDAP_MECHANISM .....	69
LDAP_PASSWORD .....	70
LDAP_SEARCH .....	71
LDAP_SERVER .....	72
LDAP_TIMEOUT .....	73
LDAP_USERNAME .....	74
LOAD_ANALYSE_TEXT .....	75
LOG .....	76
LOGIN_TICKETS .....	77
LOGPRUNE_INTERVAL .....	78
LOGPRUNE_MAX_LOGS .....	79
LOGPRUNE_MAXAGE .....	80
LOGPRUNE_MAXAGE_BATCH .....	81
LONG_PHRASE .....	82
MAX_ALLO_MEM .....	83
MAX_THREADS .....	84
MIGRATION_ERRORS_DIR .....	85
MONITOR_LIB .....	86
MONITOR_QPERF .....	87
MONLOG_FLUSH .....	88
MONLOG_MONOLITHIC .....	89
MONLOG_MONOLITHIC_PERIOD .....	90
MONLOG_TSTAMP .....	91
NO_GLBUPD_INDEX .....	92
OVFBUSZ .....	93
PRC (UNIX only) .....	94
PRINT .....	95
PRINT_EMPTY_FILE .....	96
PRINT_EMPTY_TFORM .....	97
PRINTUSER (Windows only) .....	98
RANGE_CHECK .....	99
REFRESH_TOKEN_EXPIRATION .....	100

REQUIRE_APIKEY .....	101
RESTART .....	102
RO_USER_PROCEDURES .....	103
SCRATCH .....	104
SIF .....	105
SORT .....	106
SPAWN .....	107
STO_LOCATION .....	108
STOP_WORDS .....	109
STRICT_UNICODE .....	110
SUPERMAN .....	111
SYS .....	112
TERMINAL (UNIX only) .....	113
TERMLM .....	114
THESAURUS_DEFAULT .....	115
TICKET_TIMEOUT .....	116
TOKEN_KEY .....	117
TRM (UNIX only) .....	118
TRMBUFSZ .....	119
UNKNOWN_FIELDS .....	120
XPI_LOGLEVEL .....	121
<b>TRIPserver Logical Names (TBS_) .....</b>	<b>122</b>
ASE .....	122
AT_CCL .....	123
CCL_FILE_DIR .....	124
CCL_NO_RELATIVE .....	125
COMFORTER .....	126
DIR (UNIX only) .....	127
HOSTINI .....	128
LOG .....	129
MAP (UNIX only) .....	130
MAP_DIR (UNIX only) .....	131
SCRATCH (UNIX only) .....	132
TIMEOUT .....	133
TripDaemonHost (Windows only) .....	134

TripDaemonPort (Windows only) ..... 135

TripNetPort (Windows only) ..... 136

UNIXLOGIN (UNIX only)..... 137



## Introduction

### About this Document

This document describes the TRIPsystem environment with respect to the supported configuration settings.





## The Configuration File `tdbs.conf`

### Location of `tdbs.conf`

The system wide configuration file, `tdbs.conf`, is located in the `conf` directory of the TRIPsystem installation. On UNIX, a link to the `conf` directory of the current TRIPsystem installation is created as:

```
/usr/local/trip/sys/conf
```

The `tdbs.conf` file was named `TRIPrcs` in earlier versions of TRIP and located in the root directory of the file system (in the root of the C:\ drive on Windows). The older configuration file name and its location are no longer used by TRIP.

### Configuration File Lookup on Windows

TRIPsystem will under Windows load its configuration from the file `tdbs.conf` located in the `conf` directory under the TRIPsystem installation.

Server-side add-on products (e.g. `TRIPcof`) will locate the TRIPsystem configuration by looking up its installation directory from the registry. The registry key

```
HEKY_LOCAL_MACHINE\SOFTWARE\Smaser\TRIPsystem
```

contains a value named `TDBS_HOME` that is automatically set to the current TRIPsystem installation directory. This value is present in two copies; one in the 64-bit location of the registry and one in the 32-bit location. This means that an application or TRIP module on Windows will be able to locate the TRIPsystem installation directory by looking up this registry value whether or not the application or module is 32-bit or 64-bit.

### Configuration File Lookup on UNIX

TRIPsystem will under UNIX use the `/usr/local/trip/sys/conf/tdbs.conf` file to load its configuration from.

Note that the `/usr/local/trip/sys/conf` directory is a link to the `conf` directory of the current TRIPsystem installation. Customizations of the installation procedure must not create this link as a directory instead, as it will cause instabilities in TRIP.

### Installation Procedure Considerations

If a custom installation procedure has been or will be implemented, the location of the configuration file must also be taken into account.

Notes:

- *When performing a custom installation of TRIPsystem, writing a pre-7.0 `TRIPrcs` file instead of the required `tdbs.conf` file will result in undefined behaviour when attempting to run it. Failure to correctly install server-side add-on products is also likely to occur.*
- *On UNIX, the directory `/usr/local/trip/sys/conf` is actually a symbolic link to the installation directory. If you are writing a custom installer, you must create this link yourself.*

## Batch Setup

### Printer Queues and Printer Control Files

*Notes:*

- *In Windows, only one printer definition is permitted in the `tdbs.conf` file, therefore in a TRIP installation on a Windows system, there are no printer queue, definition or control files.*
- *If you do wish to install a printer on a TRIP for Windows server, consult the TRIPsystem Installation guide, section entitled, "Configuring a Printer for Windows"*

For UNIX, the printer definition files are used as detailed in this chapter.

Print output in UNIX is, by default, sent to the printer queue indicated by the UNIX logical name `TDBS_PRINT`. However, output may be redirected to other printer queues using CCL orders such as:

```
Define PRINTER=PTR1  
Print PRINTER=PTR2
```

The first order sends the output of later Print commands (without a destination modifier e.g. `File` or `TForm`) to the printer queue specified in the file '`PTR1.PRN`', and the second command sends its output to the queue specified in the file '`PTR2.PRN`', regardless of the prior `Define` instruction.

To use a printer in such a manner, first create a printer control file called '`printername.PRN`' in the directory specified by the logical name `TDBS_PRC`. Unless these printer control files have been defined, all print will be sent to the queue designated by `TDBS_PRINT` by default.

TRIP uses a printer control file called `TDBS_PRC/PTR1.PRN` to execute the first order, and another printer control file, `TDBS_PRC/PTR2.PRN` for the second. Each line of text within these control files consists of a keyword, a colon `[:]` and a value, as shown below:

Keyword	Legal Values	Function
CHAR SET*	ENGLISH, GERman, LATIn 1, LATIn 2, MULTInational, NORwegian, ROMan, SWEdish	Any CHAR SET choice will override whatever character set was previously specified in TRIP.  (default is LATIn 1)
HIGHLIGHT OFF	any printable character or set of characters	Whatever is specified here will be printed after each hit term, for example, using an escape sequence to deactivate a print attribute such as bolding. Literal text strings or characters must be enclosed within single quotes.
HIGHLIGHT ON	any printable character or set of characters	Whatever is specified here will be printed before each hit term, for example, using an escape sequence to activate a print attribute such as bolding. Highlighting will work for an order such as Print Hhlight, where the printer control file in use provides highlighting. Literal text strings or characters must be enclosed within single quotes.
INIT	any printable character or set of characters	A printer initialization sequence is always printed or executed first, for example, literal text, or escape sequences such as changing from portrait to landscape printing. Literal text strings or characters must be enclosed within single quotes.
PAGE SIZE	rows, columns	This keyword overrides any previous format specification up to the maximum printer page size.
QUEUE	UNIX print queue name	Specifies the batch queue to be used for print preparation.
TRANS TAB	the filename portion of a file called filename.PRC. Filename.PRC is located in TDBS_PRC.	Contains a translation table specifying how characters being output will be presented to the printer.

Table 0–1 Keywords for printer control files

For example:

```

QUEUE: LP1
CHAR SET: ROM
PAGE SIZE: 60, 80

```

## Specifying Non-Printable Characters

To specify an escape sequence (ASCII character 27) in a control file, type 'esc', followed by the escape sequence surrounded by quotes. To include a control character, enter the caret [^] followed by the single-letter acronym of the desired control sequence, for example, ^P. See the printer control files given previously for an illustration.

## More About Translation Tables

A translation table consists of 16 lines with 16 hexadecimal codes. Each code position (two characters) read from left to right, downwards, represents the number of a character before its translation, and the code in that position is the number of the character it will be translated to.

The following example table is supposed to be stored in the file TDBS\_PRC:DECSWE.PRC and maps multinational characters onto the Swedish 7-bit ASCII code:

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
1	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
2	20	21	22	23	24	25	26	27	28	29	2A	2B	2C	2D	2E	2F
3	30	31	32	33	34	35	36	37	38	39	3A	3B	3C	3D	3E	3F
4	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F
5	50	51	52	53	54	55	56	57	58	59	5A	5B	5C	5D	5E	5F
6	60	61	62	63	64	65	66	67	68	69	6A	6B	6C	6D	6E	6F
7	70	71	72	73	74	75	76	77	78	79	7A	7B	7C	7D	7E	7F
8	80	81	82	83	84	85	86	87	88	89	8A	8B	8C	8D	8E	8F
9	90	91	92	93	94	95	96	97	98	99	9A	9B	9C	9D	9E	9F
A	A0	A1	A2	A3	A4	A5	A6	A7	A8	A9	AA	AB	AC	AD	AE	AF
B	B0	B1	B2	B3	B4	B5	B6	B7	B8	B9	BA	BB	BC	BD	BE	BF
C	41	41	41	41	5B	5D	5B	43	45	40	45	45	49	49	49	49
D	20	4E	4F	4F	4F	4F	5C	5C	5C	55	55	55	5E	59	20	53
E	61	61	61	61	7B	7D	7B	63	65	60	65	65	69	69	69	69
F	20	6E	6F	6F	6F	6F	7C	7C	7C	75	75	75	7E	79	20	20

Table 0-2 Sample translation table

When a translation table is used, a table character is substituted for a multinational character, and the numeric value of the multinational character acts as an index to its translation in the table. For example, in the table above the code 'C4' is being translated into '5B'.

Translation tables may also be referred to by TRIP orders such as:

```
DEFINE PCODE=decswe
DEFINE LPCODE=decswe
```

Both the orders specify that printed output is to be processed with the translation table in TDBS\_PRC/DECSWE.PRC. The first order affects normal Print output (without the destination modifiers Ffile or TForm), and the latter affects Print Local output.

*Note:*

*This function has been largely replaced by the CHAR SET keyword. It is currently maintained in support of printers not compatible with TRIP's character sets.*

## Logical Names

The logical names supported by TRIP provide mechanisms for database administrators to customize their users' environments. Those names or variables prefixed by 'TDBS' are used by the TRIP engine, whereas those prefixed by 'TRIP' are used by the TRIPclassic user interface.

Many of TRIP's functions can be influenced by setting variables in the user's environment, such as the location of the CONTROL file, the amount of accounting which is performed, the language with which the system will communicate with the user, the character set that the system is expecting data to be presented with, etc.

These variables are set differently for the different operating environments in which TRIP is present.

### UNIX

TRIP searches for logical names in the system wide TRIP configuration file, `tdbs.conf`, then in the [NonPrivileged] section of a user's local copy of `tdbs.conf` and finally for environment variables in the user's own environment.

In most cases, a setting in the user's environment will override any setting in the configuration file; however, TRIP will search for certain logical names in the `tdbs.conf` file first to prevent users disabling secure system functions, such as accounting.

### Windows

As in UNIX, logical names are searched for in the system wide configuration file, `tdbs.conf` and then in the [NonPrivileged] section of a user's local copy of `tdbs.conf`; however user and system environment variables have no effect.

The priority of logical names set in the user's local `tdbs.conf` [NonPrivileged] section, over those of logical names set in the system wide `tdbs.conf` file [NonPrivileged] section, follows that of UNIX.

## TRIPsystem Logical Names Reference (TDBS\_)

Most of these logical names are defined with defaults when TRIP is installed. Special notice is given where defaults are not defined.

### ACCDIR

<b>Function</b>	Specifies a directory to hold accounting files.
<b>Usage</b>	TDBS_ACCDIR
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	Fully specified directory (path) name
<b>Examples</b>	TDBS_ACCDIR=/usr/users/mydir

Depending on the setting of the variable ACCFLG, TRIP will attempt to create accounting files in the directory specified by ACCDIR. If this directory does not exist, users will be unable to login to TRIP. Normal users cannot override the setting of ACCDIR, thus stopping the redirection of user accounting logs. Log files are named according to the setting of ACCFLG.

## ACCESS\_TOKEN\_EXPIRATION

<b>Function</b>	Defines the expiration time in seconds for issued access tokens.
<b>Usage</b>	TDBS_ACCESS_TOKEN_EXPIRATION
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	3600
<b>Valid values</b>	A positive integer
<b>Examples</b>	TDBS_ACCESS_TOKEN_EXPIRATION=3600
<b>See Also</b>	TDBS_REFRESH_TOKEN_EXPIRATION

This variable defines the life span in seconds for access tokens. Set this to a low value and use refresh tokens for token renewal.

## ACCFLG

<b>Function</b>	Specify various options concerned with the system accounting function.
<b>Usage</b>	TDBS_ACCFLG
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	A positive integer between 1 and 511
<b>Examples</b>	TDBS_ACCFLG=217

This variable defines both how much accounting is to be performed by the system, and where the results of that accounting should be reported. By default, the system logging file is located in the directory pointed to by the SYS variable, and is called DEBIT.LOG.

The value specified by the ACCFLG variable is a bitmask, where each of the bits 0 through 7 have a defined meaning. The different bits can be defined by simply adding the bit values together. For example, to set bits 0, 3, and 5, the value of ACCFLG would be 41 (1 + 8 + 32).

A full definition of the various meanings of this variable is given in Chapter Four the TRIPsystem administration guide, 'System Logging Functions'.

Bit	Value	Meaning
0	1	Set the name of the log file to the TRIP username with extension '.LOG'. If neither this bit, nor bit 1 is set, the log file used is the system default DEBIT.LOG.
1	2	Set the name of the log file to the filename portion of the SIF variable with the extension '.LOG'. If neither this bit, nor bit 0 is set, the log file used is the system default DEBIT.LOG.
2	4	Use the filename portion of the SIF variable as the user identifier within the log file, rather than the TRIP username.
3	8	Log find, frequency, measure orders and opening clusters, in addition to the defaults.
4	16	Do not accumulate database statistics until logout, but write statistics every time the open database changes name.
5	32	Write output statistics every time a new show order begins, rather than waiting until the database changes, or until logout.
6	64	Log show focus orders as well as normal show orders.
7	128	Prevents output of records from searches performed against databases which are no longer open.
8	256	Use the system default name DEBIT.LOG also when storing the file in the directory indicated by TDBS_ACCDIR. Mutually exclusive with bits 0,1 and 2.

Table 12–3 Bits flags for accounting



## APILOG

<b>Function</b>	Enables logging of TRIPtoolkit API calls.
<b>Usage</b>	TDBS_APILOG
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	A fully qualified directory path
<b>Examples</b>	TDBS_APILOG=/opt/trip/v800/log
<b>See also</b>	TDBS_XPI_LOGLEVEL

If this logical name is defined, a process-specific log file will be created and written to. For every TRIPtoolkit API call that is made during the course of a session, the following items will be logged:

- Name of function
- Input parameter values
- Output parameter values
- Return code

Having this logical name enabled causes will significantly reduce TRIP's performance and is intended for troubleshooting purposes only. For this reason, TBS\_APILOG is to be set only when it is necessary to produce a log of the API calls made by an application, usually at the request of TRIP support.

## AUDIT

<b>Function</b>	Specifies an optional location for database audit files produced by TRIP. Database audit file creation cannot be activated with TRIPclassic, only with TRIPmanager or by a database import file.
<b>Usage</b>	TDBS_AUDIT
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	A fully qualified directory path
<b>Examples</b>	TDBS_AUDIT=/opt/trip/v800/log

Refer to the TRIPmanager administration guide for more information about this property.

## ASELIBS

<b>Function</b>	Specifies a list of shareable libraries to be searched when calling ASE functions from within TRIPsystem.
<b>Usage</b>	TDBS_ASELIBS
<b>Looked for in</b>	Non-privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	A comma separated list of library file names.
<b>Examples</b>	<pre>TDBS_ASELIBS=mylib1,mylib2,mylib3</pre> <p>Where mylib1, mylib2, etc. may be other logical names mapping to each individual ASE i.e.:</p> <pre>mylib1=c:\mylibs\mylib1.dll, mylib2=c:\mylibs\mylib2.dll, etc.</pre>

## AUTH\_PROVIDER

<b>Function</b>	To establish LDAP as the authentication provider, set this variable to LDAP. The default behaviour of the system in the absence of such a setting is to fallback to using CONTROL for all authentication requests.
<b>Usage</b>	TDBS_AUTH_PROVIDER
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	Use CONTROL for all authentication requests
<b>Valid values</b>	LDAP
<b>Examples</b>	TDBS_AUTH_PROVIDER=LDAP

## AUTO\_SAVE

<b>Function</b>	Specify that the current record should be stored in the BAF file whenever a part record is modified or inserted.
<b>Usage</b>	TDBS_AUTO_SAVE
<b>Looked for in</b>	Non-privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	Y (yes) or N (no)
<b>Examples</b>	TDBS_AUTO_SAVE=Y



## BAFFIT\_SECURITY

<b>Function</b>	Prevents the loading of records from outside TRIP.
<b>Usage</b>	TDBS_BAFFIT_SECURITY
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	Y (yes) or N (no)
<b>Examples</b>	TDBS_BAFFIT_SECURITY=Y

If BAFFIT\_SECURITY is set to Y the loading of TFORM files into TRIP databases must be done from inside TRIP. (The name of the logical name reflects the fact that all such loading of TFORM files always involves the running of the TRIP utility BAFFIT.)

This is to say that the standard method of executing cannot be used directly for the loading of TRIP records. If BAFFIT\_SECURITY is not set at all or set to something other than Y then BAFFIT may, just as was always possible in earlier TRIP versions, be run directly or executed from a script file.

If a custom-built script file is presently used for loading of TFORM files, then it may be adapted to do so even if BAFFIT\_SECURITY is set to Y. The part of it which presently causes the loading must be replaced by a sequence running TRIP (this requires the script file to have access to a TRIP user / password combination, of course) - in which sequence the CCL command LOad is given.

## BAFFRE\_TIMEOUT

<b>Function</b>	Specifies the time the TRIP system index utility will wait to get an exclusive lock on the BAF file when releasing old records after an index job.
<b>Usage</b>	<code>TDBS_BAFFRE_TIMEOUT</code>
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	300 seconds (5 minutes)
<b>Valid values</b>	Any positive integer value
<b>Examples</b>	<code>TDBS_BAFFRE_TIMEOUT=600</code> Sets the timeout period for BAF record release to 10 minutes.

## BASES

<b>Function</b>	The default location for the BAF, BIF and VIF files for new databases.
<b>Usage</b>	TDBS_BASES
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	UNIX:  /usr/local/trip/data  Windows:  C:\Program Files\Smaser\TRIP Databases
<b>Valid values</b>	A fully qualified directory path.
<b>Examples</b>	TDBS_BASES=/var/trip/bases

When a new database is created, its files are assigned TDBS\_BASES as storage location if the logical name is defined. This can be modified when the database design is being created (via TRIPmanager, TRIPclassic or API).

The TDBS\_BASES logical name is created by the TRIPsystem installer and the directory identified by the default value is created. If an upgrade installation is being performed, TDBS\_BASES will be created in a similar fashion if not previously defined in tdbb.conf.



## BLOCK\_SIZE

<b>Function</b>	Determines the file block size to use for BAF, BIF, VIF and SIF files.
<b>Usage</b>	TDBS_BLOCK_SIZE
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	2048
<b>Valid values</b>	2048 or 4096
<b>Examples</b>	TDBS_BLOCK_SIZE=4096

The storage block size in the database files (BAF, BIF and VIF) and in the session index file (SIF) is by default 2048 bytes. This is the value that has been used in all previous versions of TRIP.

On contemporary operating systems having a block size of 2048 bytes results in more disk I/O than necessary. To reduce this type of overhead and boost I/O performance (e.g. with indexing), the TDBS\_BLOCK\_SIZE can be set to a higher value (currently only 2048 and 4096 are supported).

**NOTE:** Any BAF, BIF, VIF or SIF file created with a block size set to any other value than 2048 will not work on older versions of TRIP than 8.1-0. Attempting to use a file with a different block size will either crash the TRIP session or fail to open. A crash will happen with 7.1 and older as well as on 7.2 versions older than 7.2-6:3 and on 8.0 versions older than 8.0-10:1.

## **BOLD\_COLOR (Windows only)**

<b>Function</b>	When using TRIPclassic from a Windows command prompt, the window retains the background and text colors setup for it.
<b>Usage</b>	TRIP_BOLD_COLOR
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	Supported values are:  B = Blue C = Cyan G = Green M= Magenta R = Red Y = Yellow
<b>Examples</b>	<pre>TRIP_BOLD_COLOR=Y</pre> <p>Sets the Bold color in a TRIPclassic for Windows session to yellow.</p>

## BUT\_LOCATION

<b>Function</b>	Specifies the location (name and path) of the temporary BUT file used by the indexing program. (Used only when indexing is done via the index script).
<b>Usage</b>	TDBS_BUT_LOCATION
<b>Looked for in</b>	Non-privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	Full path and name of file
<b>Examples</b>	UNIX/Linux: <code>TDBS_BUT_LOCATION=/trip/tmp/db2.but</code> Windows: <code>TDBS_BUT_LOCATION=C:\trip\tmp\db2.but</code>

## CHARS

<b>Function</b>	Specifies the default character set to be used by TRIP.
<b>Usage</b>	TDBS_CHARS
<b>Looked for in</b>	Non-privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	LA1
<b>Valid values</b>	See table below.
<b>Examples</b>	TDBS_CHARS=LA1

Value	Meaning	Bit Width
LA1	ISO LAtin 1	8 bit
LA2	ISO LAtin 2	8 bit
LA3	ISO LAtin 3	8 bit
CHI	GB-2312-80 CHInese	16 bit
GBK	Superset of GB 2312-1980 Chinese	16 bit

The value of the CHARS variable is used for initializing the translation tables, which TRIP uses to map characters between different character sets. The value specified here is the default, but can be overridden by explicitly declaring the character set, for example in a TForm file.

### Notes:

- Unicode enabling (using UTF-8) of existing databases can only be done with TRIPmanager or via the API.
- New databases are Unicode enabled by default if created in a Unicode session (e.g. from TRIPmanager or via the TRIPnxp and TRIPjxp APIs).
- TRIP can handle a mixture of Unicode and non-Unicode databases provided there is only one type of encoding for the non-Unicode databases.
- TRIP will continue to be backward compatible with the current methods of text handling, i.e. anyone wishing to stay with a LATIN-1 database can continue to do so without the need to convert to Unicode.
- TRIPclassic will remain Latin-n and GBK enabled only

## CHIVOC

<b>Function</b>	Specifies file containing data for Chinese word segmentation.
<b>Usage</b>	TDBS_CHIVOC
<b>Looked for in</b>	Non-privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	Fully specified file name (including path)
<b>Examples</b>	UNIX/Linux:  TDBS_CHIVOC=/disk3/trip/chinese/chivoc.dat  Windows:  TDBS_CHIVOC=C:\trip\chinese\chivoc.dat

The specified file should contain data for use with the algorithm for Chinese word segmentation.

## CLS

<b>Function</b>	Specifies a path to a directory where classification scheme files will be located.
<b>Usage</b>	TDBS_CLS
<b>Looked for in</b>	Non-privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	Fully qualified directory path name
<b>Examples</b>	UNIX/Linux: TDBS_CLS=/disk3/trip  Windows: TDBS_CLS=C:\trip\

## CODEPAGE (Windows TRIPclassic only)

<b>Function</b>	Specifies what code page to use in the TRIPclassic user interface.
<b>Usage</b>	TRIP_CODEPAGE
<b>Looked for in</b>	Non-privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	A code page number valid on the current operating system. For example, 437, 850, 858, 865 or 1252
<b>Examples</b>	Windows:  <code>TRIP_CODEPAGE=858</code>

The code page that TRIPclassic will use for its display of text and window borders is the system's default codepage. A typical codepage on western Windows systems is 850.

The TRIPsystem installer will set the TRIP\_CODEPAGE variable to 858. Other code pages suitable for this are 437, 850, 865 and 1252, all of which except 1252 supports semi graphics.

For more details on this setting, please refer to the TRIP for Windows Installation Guide.

## COM

<b>Function</b>	Specifies a directory containing user written command scripts.
<b>Usage</b>	TDBS_COM
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	A fully specified directory (path) name
<b>Examples</b>	UNIX:  TDBS_COM=/opt/trip/scripts  Windows:  TDBS_COM=C:\TRIP\scripts
<b>See also</b>	TDBS_SPAWN, TBS_AT_CCL

Whenever a user invokes an external command script using the CCL command '@', the system will look for the named script in the directory pointed to by the TDBS\_COM logical name.

If TDBS\_COM is not defined AND the name of the script does not include a path definition, the system will attempt to locate the named script in the user's current working directory, first by name alone and then with the extensions ".com" and ".cmd" (in that order).

If, however, the CCL command contains a path definition, the system will simply attempt to execute the script using that path.

For example:

```
CCL:      @myscript
```

will look in COM and, if such is not defined, in the user's current working directory, while

```
CCL:      /opt/trip/scripts/myscript.com
```

will only look in '/opt/trip/scripts' for 'myscript.com'.



## CONFLATOR\_LANG

**Function** Specifies the language TRIP will use when stemming for classification and non-Boolean search.

**Usage** TDBS\_CONFLATOR\_LANG

**Looked for in** Non-Privileged section

**Defined by default?** Yes

**Default value** ENG

**Valid values** Any three-letter code taken from the table below:

Code	Meaning
ENG	ENGLISH
FIN	FINnish
GER	GERman
NOR	NORwegian
SWE	SWEdish

**Examples** TDBS\_CONFLATOR\_LANG=GER

TRIP uses the value of the CONFLATOR\_LANG logical name when indexing and searching fields that are marked as included in non-Boolean calculations as well as when training the classifier and when classifying records.

## CONFLATORS

<b>Function</b>	Specifies one or more files containing functions called when stemming for classification and non-Boolean search.
<b>Usage</b>	TDBS_CONFLATORS
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	UNIX:  <installation-dir>/<version>/<lib>libtripstem.so  Windows:  <installation-dir>/<version>/<bin>tripstem.dll
<b>Valid values</b>	Fully specified name (including path) of file containing callable functions.
<b>Examples</b>	UNIX:  TDBS_CONFLATORS=/opt/trip/v820/lib/libtripstem.so  Windows:  TDBS_CONFLATORS=C:\TRIP\v820\bin\tripstem.dll

The file installed with TRIP currently contains Porter stemming routines for the languages accepted by the TDBS\_CONFLATOR\_LANG logical name.

If you want to replace any of the default functions with your own, add the path to the file with your functions before the default file, separated by a “,” , e.g.:

```
TDBS_CONFLATORS=/home/mystem.so,/opt/trip/v820/lib/libtripstem.so
```

## CTL

<b>Function</b>	Specifies the location of the system schema dictionary, CONTROL.
<b>Usage</b>	TDBS_CTL
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	The SYS directory in the TRIP tree
<b>Valid values</b>	A fully specified directory (path) name.
<b>Examples</b>	UNIX:  TDBS_CTL=/opt/trip/v800/sys  Windows:  TDBS_CTL=C:\TRIP\v800\sys

TRIP locates the schema dictionary, CONTROL, using the variable CTL. This allows application developers to maintain parallel environments simply by redefining the CTL variable for their process. This has many advantages, primarily integrity and security.

Take care, however, when making copies of the CONTROL database, as this database must be upgraded by the TRIP installation procedure when the TRIP version changes. Before creating such an environment, consult your local TRIP representative about the MODCON procedure.

## DEFATTR (UNIX only)

<b>Function</b>	Specifies a BOLD alternative for VT-terminal clones that doesn't include a definition for a BOLD variant.
<b>Usage</b>	TDBS_DEFATTR
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	B (bold), U (underline) or R (reverse)
<b>Examples</b>	TDBS_DEFATTR=R



## DELETE\_MAX

<b>Function</b>	Specifies the max limit for deleting records via the CCL DELETE command.
<b>Usage</b>	TRIP_DELETE_MAX
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	Any non-negative integer, N or NOLIMIT
<b>Examples</b>	TDBS_DELETE_MAX=20

Setting the value of TDBS\_DELETE\_MAX to a string which is not starting with a digit gives results in a delete max of MAXINT. Setting the value to 0 gives the result that deleting records is now allowed.

## DEMO

<b>Function</b>	Specifies the location of the demonstration databases.
<b>Usage</b>	TRIP_DEMO
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	The 'demo' directory in the TRIP tree
<b>Valid values</b>	A fully specified directory (path) name.
<b>Examples</b>	UNIX:  TDBS_DEMO=/opt/trip/v800/demo  Windows:  TDBS_DEMO=C:\TRIP\v800\demo

Included with every TRIP system are a number of demonstration databases:

Database	Contents
ALICE	Contents of Alice in Wonderland and Through the Looking Glass
CARROLL	Same content as ALICE, but arranged using head and part records
CORR	Correspondence to and from Paralog staff members (Paralog was the company that created TRIP)
THESALI	Thesaurus for use with ALICE and CARROLL

All of these databases are located using the DEMO variable.

## DISALLOW\_GUEST

<b>Function</b>	By default, if a user using an external authentication provider, such as LDAP, provides a valid set of credentials for that authentication provider, but the user is unknown to TRIP, the user will be logged into TRIP as a guest user (under the BUILTIN_GUEST account). To disable this functionality set this variable to True.
<b>Usage</b>	TDBS_DISALLOW_GUEST
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	BUILTIN_GUEST account allowed
<b>Valid values</b>	True
<b>Examples</b>	TDBS_DISALLOW_GUEST=True

## DISPLAY\_FREQ

<b>Function</b>	Enable/disable merge across databases of terms in frequency restricted term lists.
<b>Usage</b>	TDBS_DISPLAY_FREQ
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	M
<b>Valid values</b>	<b>M</b> (merge) or <b>N</b> (no merge)
<b>Examples</b>	TDBS_DISPLAY_FREQ=N

Using the value M (merge) with large database clusters might cause long execution time and increased use of memory. In such scenarios, the TDBS\_DISPLAY\_FREQ logical name should be set to N (no merge).



## DISPLAY\_ORIG

<b>Function</b>	Specifies that displayed data should be fetched from the BAF file instead of the BIF file.
<b>Usage</b>	TDBS_DISPLAY_ORIG
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	N
<b>Valid values</b>	Y (yes) or N (No)
<b>Examples</b>	TDBS_DISPLAY_ORIG=Y



## EDIT (TRIPclassic only)

<b>Function</b>	Specifies which editor is to be used during creation/modification of formats.
<b>Usage</b>	TDBS_EDIT
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	UNIX: Value of the UNIX EDITOR environment variable; normally "vi"  Windows: Windows notepad
<b>Valid values</b>	Any valid and installed editor command name.
<b>Examples</b>	UNIX:  TDBS_EDIT=emacs  Windows:  TDBS_EDIT=write

In TRIPclassic only, whenever a user attempts to create or modify a report or a procedure/macro, or use an external editor for data entry with <Gold><E>, the TRIP kernel will invoke one of the system editors for the user.

The EDIT variable allows the user to specify which editor is to be used.

## ENTER\_OS (TRIPclassic only)

<b>Function</b>	Specifies whether it is possible to exit to the calling shell from a TRIP application.
<b>Usage</b>	TRIP_ENTER_OS
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	Y
<b>Valid values</b>	Y (yes) or N (no)
<b>Examples</b>	TRIP_ENTER_OS=N
<b>See also</b>	TDBS_SPAWN

In TRIPclassic only, if TDBS\_SPAWN is set to Y, the TRIP\_ENTER\_OS logical name is used to determine if external execution also will include launching an operating system shell.

## ERRMAILST (UNIX only)

<b>Function</b>	Specify to whom mail should be sent in case of error.
<b>Usage</b>	TDBS_ERRMAILST
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	Any username, or list of comma-separated usernames
<b>Examples</b>	UNIX:  TDBS_EDIT=emacs  Windows:  TDBS_EDIT=write

Whenever an error occurs during a batch job (such as INDEX), the TRIP system generates a mail message to send to either the user who submitted the job or to all of the people listed in the ERRMAILST variable.

Here is an example of a mail message:

```
****      Error when indexing database ALICE [during SCIFFIT]
****      Please consult the log file :-
****              /TRIP/LOGS/INDEX_ALICE.LOG
****      for more detail
```

## EXE

<b>Function</b>	Specifies the location of the TRIP executables and scripts.
<b>Usage</b>	TDBS_EXE
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	The 'bin' directory in the TRIP tree structure
<b>Valid values</b>	A fully specified directory (path) name
<b>Examples</b>	UNIX:  TDBS_EXE=/opt/trip/v800/bin  Windows:  TDBS_EXE=C:\TRIP\v800\bin

When a command script is running, such as INDEX or LOAD, the executables (programs) that it attempts to invoke are found using the EXE variable. It can be very useful to maintain two separate environments, particularly when developing ASE routines, by reassigning the EXE variable for your programmers.

## EXEC\_SCRIPT

<b>Function</b>	Enable/disable the possibility to execute scripts in CCL.
<b>Usage</b>	TDBS_EXEC_SCRIPT
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	Y
<b>Valid values</b>	Y (yes) or N (no)
<b>Examples</b>	TDBS_EXEC_SCRIPT=N
<b>See also</b>	TDBS_SPAWN

**NOTE:** TDBS\_SPAWN is the main switch, setting it to N disables the setting of this one.

## FIND\_TIMEOUT

<b>Function</b>	Specifies the time, in seconds, before retrying a search when a record is found to be unsearchable due to indexing.
<b>Usage</b>	TDBS_FIND_TIMEOUT
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	One second
<b>Valid values</b>	Any integer
<b>Examples</b>	TDBS_FIND_TIMEOUT=5

Searching a database at the same time as the database is being indexed can result in a message stating that the database is not available for searching at that moment.

This can happen when a search is made for a term that is currently being modified in the indices and the data blocks involved have not yet been completely flushed to the index files.

Before issuing the error message, TRIP makes another try to re-execute the search after one second, but sometimes this time delay is not sufficient

This time-out option allows the TRIP administrator to set to a value that is acceptable to the users.

## FIRST\_WEEKDAY

<b>Function</b>	Specifies the first day of the week, for use in the date calculation performed by the CCL function TODAY().
<b>Usage</b>	TDBS_FIRST_WEEKDAY
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	MON
<b>Valid values</b>	English three-letter abbreviation of the week day name; MON, TUE, WED, THU, FRI, SAT, and SUN
<b>Examples</b>	TDBS_FIRST_WEEKDAY=SUN

TRIP uses Monday as the default first day of the week, as is specified by the ISO8601 standard. Use this configuration property to change it to another week day.

This configuration property only impacts the behavior of the CCL function TODAY().



## FLOCK\_TIMEOUT

<b>Function</b>	Specifies the write access timeout when writing to a BAF or BIF file.
<b>Usage</b>	TDBS_FLOCK_TIMEOUT
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	5
<b>Valid values</b>	Whole number of seconds expressed as an integer
<b>Examples</b>	TDBS_FLOCK_TIMEOUT=20

The overall request timeout in a networked application is typically 60 seconds. In order not to cause network timeout to happen during a record write operation, make sure the TDBS\_FLOCK\_TIMEOUT value is less than 60 seconds.

## GLBUPD\_OPEN\_DB\_ONLY

<b>Function</b>	GLBUPD will normally act upon all databases that have been opened during the current TRIP session. Setting this variable to 'True' will make GLBUPD act only upon those databases opened by the last BASE command.
<b>Usage</b>	TDBS_GLBUPD_OPEN_DB_ONLY
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	All databases opened in the current session will be acted upon by GLBUPD
<b>Valid values</b>	True
<b>Examples</b>	TDBS_GLBUPD_OPEN_DB_ONLY=True

## GRAPH\_LOG

<b>Function</b>	Enables logging from the graph database functionality. Log files named “graphprocessor_PID.log”, where PID is the process ID of the TRIP process. The files are created in the TDBS_LOG directory.
<b>Usage</b>	TDBS_GRAPH_LOG
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	Y(es) or N(o)
<b>Examples</b>	TDBS_GRAPH_LOG=Y

As is the case with all logging, graph logging has a negative performance impact, meaning it should only be enabled for short periods and primarily for troubleshooting purposes.

## GRAPH\_RENUM

<b>Function</b>	Instructs the graph database processor to removes intermediary search sets and renumber the result search set.
<b>Usage</b>	TDBS_GRAPH_RENUM
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	N
<b>Valid values</b>	Y(es) or N(o)
<b>Examples</b>	TDBS_GRAPH_RENUM=Y

The graph database processor typically creates several intermediary search sets when it executes a graph operation. To remove those and only keep the end result, the TDBS\_GRAPH\_RENUM logical name should be set to Y.

## HIGHLIGHT

<b>Function</b>	Enable/disable the search hit highlight in formatted output and with related API functions.	
<b>Usage</b>	TDBS_HIGHLIGHT	
<b>Looked for in</b>	Non-Privileged section	
<b>Defined by default?</b>	Yes	
<b>Default value</b>	M	
<b>Valid values</b>	<b>Y</b>	Highlight hit terms but only the right most ones after an AND or proximity operation.
	<b>N</b>	No highlight
	<b>A</b>	Highlight all hits
	<b>M</b>	Highlight all hits but exclude all non-relevant posting values, e.g. numerical values.
<b>Examples</b>	TDBS_HIGHLIGHT=M	

## HOME

<b>Function</b>	This property is automatically defined by TRIP itself to the fully qualified path to the TRIPsystem installation directory. This value is defined prior to reading the configuration files, so it can be used as part of the values of other logical names to avoid specifying hard coded paths.
<b>Usage</b>	TDBS_HOME
<b>Looked for in</b>	Automatically defined
<b>Defined by default?</b>	Yes
<b>Default value</b>	The fully qualified path to the installation directory.
<b>Valid values</b>	N/A
<b>Examples</b>	UNIX:  TDBS_HOME=/opt/trip/v800  Windows:  TDBS_HOME=C:\TRIP\v800

This property must not be set explicitly in the environment or in any of the configuration files.

## IMPERSONATE\_GROUP (UNIX only)

<b>Function</b>	Allows TRIP processes to assume the group membership of the group assigned to the file permissions of the TRIP databases.
<b>Usage</b>	TDBS_IMPERSONATE_GROUP
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	FALSE
<b>Valid values</b>	TRUE or FALSE
<b>Examples</b>	TDBS_IMPERSONATE_GROUP=TRUE

This property, if set to TRUE, allows TRIP to impersonate the operating system group memberships that the TRIP database file permissions specify. This used to be standard behavior until version 8.1-3.

The OS group impersonation behavior is not compatible with how TRIP normally operates for clustered search and display operations. Therefore, this property should only be considered if the operating system owner and group for the TRIP database files cannot be set uniformly so that the operating system user that the TRIP processes run as have complete access to them.

If this property is set, the TDBS\_MAX\_TREADS property should be set to 0 (zero). While this will have a slight negative performance impact, it also eliminates any problems with incorrect search results that otherwise may occur.

## LANG

<b>Function</b>	Specifies the language with which TRIP will communicate with the user.
<b>Usage</b>	TDBS_LANG
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	UNIX: Specified during installation. Windows: ENG
<b>Valid values</b>	Any three-letter code taken from the table below.
<b>Examples</b>	TDBS_LANG=GER

The valid values for LANG are:

Code	Meaning
CHI	CHInese
ENG	ENGLISH
FIN	FINnish
GER	GERman
NOR	NORwegian
SWE	SWEdish

TRIP uses the value of the LANG variable to determine which language to use when reporting errors, defining a CCL dialect or while giving help. If the LANG variable is defined to a language not specified in the above list, an English error will result which will stop entry to the TRIP system. If the LANG variable defines a legal language code, but the message to be output does not exist for that language, the English message will be output by default.



## LDAP\_ANONYMOUS

<b>Function</b>	In order to find users, TRIP needs to be able to browse the LDAP repository. If the repository supports anonymous access for browsing, set this variable to True, otherwise set it to False.
<b>Usage</b>	TDBS_LDAP_ANONYMOUS
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	True or False
<b>Examples</b>	TDBS_LDAP_ANONYMOUS=False

This logical name is only considered if the value of AUTH\_PROVIDER is set to LDAP.

## LDAP\_AUTOUSER

<b>Function</b>	Registers a TRIP user profile for the authenticated LDAP user if the user does not already exist in TRIP. The owner of the user is determined by the TDBS_LDAP_AUTOUSER_UM logical name.
<b>Usage</b>	TDBS_LDAP_AUTOUSER
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	False
<b>Valid values</b>	True or False
<b>Examples</b>	TDBS_LDAP_AUTOUSER=True

This logical name is only considered if the value of AUTH\_PROVIDER is set to LDAP.

## LDAP\_AUTOUSER\_GROUP

<b>Function</b>	Identifies the name of a group that will be assigned to users created automatically if the TDBS_LDAP_AUTOUSER logical name is set to True.  The group must be owned by the user identified by the logical name TDBS_LDAP_AUTOUSER_UM, or by SYSTEM if that logical name is not set.
<b>Usage</b>	TDBS_LDAP_AUTOUSER_GROUP
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	The name of a TRIP group.
<b>Examples</b>	TDBS_LDAP_AUTOUSER_GROUP=LDAPUSERS

This logical name is only considered if the value of AUTH\_PROVIDER is set to LDAP, and LDAP\_AUTOUSER is set to True.

## LDAP\_AUTOUSER\_PWSYNC

<b>Function</b>	When the value of LDAP_AUTOUSER is set to True, this logical name causes TRIP to set the password for the TRIP profile to that of the authenticated LDAP user if it differs.
<b>Usage</b>	TDBS_LDAP_AUTOUSER_PWSYNC
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	False
<b>Valid values</b>	True or False
<b>Examples</b>	TDBS_LDAP_AUTOUSER_PWSYNC=True

This logical name is only considered if the value of AUTH\_PROVIDER is set to LDAP, and LDAP\_AUTOUSER is set to True.

## LDAP\_AUTOUSER\_UM

<b>Function</b>	Identifies the user manager (UM) that will be assigned as owner to TRIP users created automatically if the logical name TDBS_LDAP_AUTOUSER is set to True.  If this logical name is not set, the UM user will default to SYSTEM.
<b>Usage</b>	TDBS_LDAP_AUTOUSER_UM
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	SYSTEM
<b>Valid values</b>	The name of a TRIP user with user manager (UM) privileges.
<b>Examples</b>	TDBS_LDAP_AUTOUSER_UM=TRIPDBA

This logical name is only considered if the value of AUTH\_PROVIDER is set to LDAP, and LDAP\_AUTOUSER is set to True.

## LDAP\_BASE

<b>Function</b>	When attempting to authenticate a user, the user's identity will typically be provided as an RDN (Relative Distinguished Name) rather than a fully specified DN (Distinguished Name). In order to turn that RDN into a DN for authentication, you must provide this variable as a base for the authentication by an RDN.
<b>Usage</b>	TDBS_LDAP_BASE
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	Base part of a DN in a LDAP repository
<b>Examples</b>	TDBS_LDAP_BASE=ou=tox,o=pharma,c=us

This logical name is only considered if the value of AUTH\_PROVIDER is set to LDAP.

## LDAP\_CACERT (UNIX only)

<b>Function</b>	Fully qualified path to the CA certificate to use for validation.
<b>Usage</b>	TDBS_LDAP_CACERT
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	A fully qualified path to a PEM-formatted certificate file.
<b>Examples</b>	<p>This example informs TRIP of the name of the CA certificate to use:</p> <pre>TDBS_LDAP_CACERT=/etc/pki/CA/example.pem</pre>

This logical name is only considered if the value of AUTH\_PROVIDER is set to LDAP and the value of LDAP\_MECHANISM is set to TLS or SSL.

This logical name replaces the older variable LDAP\_CERT\_TLS\_CACERT.

## LDAP\_CERT\_DIR (UNIX only)

<b>Function</b>	Fully qualified path to the certificate directory.
<b>Usage</b>	TDBS_LDAP_CERT_DIR
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	A fully qualified directory path.
<b>Examples</b>	This example informs TRIP of the name of a directory with Base64-encoded certificates in PEM format:  <code>TDBS_LDAP_CERT_DIR=/etc/pki/CA</code>

This logical name is only considered if the value of AUTH\_PROVIDER is set to LDAP and the value of LDAP\_MECHANISM is set to TLS or SSL.

This value, if defined, must point to a directory containing CA certificates in PEM format. The files each contain one CA certificate. The files are looked up via links named after the CA subject name hash value, which must therefore be available.

Refer to OpenLDAP documentation for details on how to set up a CA certificate directory.

This variable replaces the older variables LDAP\_CERT\_SSL\_CERTDB and LDAP\_CERT\_TLS\_CERTDIR.



## LDAP\_CERT\_TRUSTALL

<b>Function</b>	This variable determines if server-side certificates should be validated or if they should be accepted at face value.
<b>Usage</b>	TDBS_LDAP_CERT_TRUSTALL
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	False
<b>Valid values</b>	True or False
<b>Examples</b>	<p>This example configures LDAP SSL/TLS connections to accept any server certificate:</p> <pre>TDBS_LDAP_CERT_TRUSTALL=True</pre>

This logical name is only considered if the value of AUTH\_PROVIDER is set to LDAP and the value of LDAP\_MECHANISM is set to TLS.

This variable replaces the older variables LDAP\_CERT\_SSL\_TRUSTALL and LDAP\_CERT\_TLS\_TRUSTALL.

## LDAP\_ERRORLOG

<b>Function</b>	Enables the ldap_error.log file, used for analysis of LDAP login problems. If enabled, the log file is written to the directory defined by TDBS_LOG. Keep disabled for normal operation.
<b>Usage</b>	TDBS_LDAP_ERRORLOG
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	False
<b>Valid values</b>	True, False or Debug
<b>Examples</b>	<code>TDBS_LDAP_ERRORLOG=True</code>

This logical name is only considered if the value of AUTH\_PROVIDER is set to LDAP.

If set to True, the log file will only be written to if errors occur, and then only with information about the error itself.

If set to Debug, plenty of additional information will be written to the log file. This should primarily be used to analyze general LDAP connection problems. Do not use this value on production systems.

## LDAP\_FALLBACK

<b>Function</b>	Allows the login of users with TRIP user profiles that do not correspond to any LDAP user to log in even when LDAP authentication is in use. Note that SYSTEM will always be permitted to log in.
<b>Usage</b>	TDBS_LDAP_FALLBACK
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	False
<b>Valid values</b>	True or False
<b>Examples</b>	TDBS_LDAP_FALLBACK=True

This logical name is only considered if the value of AUTH\_PROVIDER is set to LDAP.

## LDAP\_MATCH

<b>Function</b>	Once the user has been found (i.e. their RDN has been dereferenced to a DN) its record must be turned into a TRIP username for use within the CONTROL database. The following variable is used to specify the field from the user record that will provide this mapping.
<b>Usage</b>	TDBS_LDAP_MATCH
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	A field name from the user record in the LDAP repository
<b>Examples</b>	<code>TDBS_LDAP_MATCH=uid</code> <code>TDBS_LDAP_MATCH=sAMAccountName</code>

This logical name is only considered if the value of AUTH\_PROVIDER is set to LDAP.

## LDAP\_MECHANISM

<b>Function</b>	Communication with the LDAP server(s) can take place in two different ways, either basic (the SIMPLE mechanism) or via an encrypted transmission (the TLS or SSL mechanism). Set this variable accordingly.
<b>Usage</b>	TDBS_LDAP_MECHANISM
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	SIMPLE, TLS or SSL
<b>Examples</b>	<code>TDBS_LDAP_MECHANISM=SIMPLE</code>

This logical name is only considered if the value of AUTH\_PROVIDER is set to LDAP.

## LDAP\_PASSWORD

<b>Function</b>	If anonymous browse access is not supported, you must provide the DN (username) and credentials (password) for the user that will be used to perform browse operations when searching for users to authenticate. This variable provides the password for this user.
<b>Usage</b>	TDBS_LDAP_PASSWORD
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	Any string valid as a password in this context
<b>Examples</b>	TDBS_LDAP_PASSWORD=Abcd1234

This logical name is only considered if the value of AUTH\_PROVIDER is set to LDAP.

## LDAP\_SEARCH

<b>Function</b>	To find a user by RDN, specify an LDAP search pattern using the %u% substitution string to stand for the user's provided RDN. Any occurrence of the "%u%" pattern within the string will be replaced with whatever "username" is provided to TRIP during the login process.
<b>Usage</b>	TDBS_LDAP_SEARCH
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	An LDAP search pattern (e.g. as specified in the example)
<b>Examples</b>	<code>TDBS_LDAP_SEARCH=(&amp;(objectclass=person)(uid=%u%))</code>

This logical name is only considered if the value of AUTH\_PROVIDER is set to LDAP.

The name of the objectclass is dependent on the organization of your LDAP repository, and the field searched by %u% is any field in the user records in this repository.

If the LDAP server is an Active Directory, the (uid=%u%) part of the search expression should normally be replaced by (sAMAccountName=%u%).

## LDAP\_SERVER

<b>Function</b>	The LDAP provider needs to know which server to use for authenticating. This variable definition the name of a single server, optionally including a port number. In the absence of a port number, the default port for LDAP (or LDAP over SSL) will be provided by the system.
<b>Usage</b>	TDBS_LDAP_SERVER
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	An LDAP server name or address, optionally including port number
<b>Examples</b>	<pre>TDBS_LDAP_SERVER=pluto TDBS_LDAP_SERVER=charon:3030</pre>

This logical name is only considered if the value of AUTH\_PROVIDER is set to LDAP.



## LDAP\_TIMEOUT

<b>Function</b>	This variable provides a maximum number of milliseconds that TRIP should wait for a response from the LDAP server(s).
<b>Usage</b>	TDBS_LDAP_TIMEOUT
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	A time in milliseconds
<b>Examples</b>	<p>This example sets up a maximum response time of 3 seconds:</p> <pre>TDBS_LDAP_TIMEOUT=3000</pre>

This logical name is only considered if the value of AUTH\_PROVIDER is set to LDAP.

## LDAP\_USERNAME

<b>Function</b>	If anonymous browse access is not supported, you must provide the DN (username) and credentials (password) for the user that will be used to perform browse operations when searching for users to authenticate. This variable provides the username for this user.
<b>Usage</b>	TDBS_LDAP_USERNAME
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	A DN specifying a user in the LDAP repository
<b>Examples</b>	<p>This example sets up a maximum response time of 3 seconds:</p> <pre>TDBS_LDAP_USERNAME= cn=Mg,dc=johnd,dc=com</pre>

This logical name is only considered if the value of AUTH\_PROVIDER is set to LDAP.

## LOAD\_ANALYSE\_TEXT

<b>Function</b>	Instructs the LOAD program how to analyse text values
<b>Usage</b>	TDBS_LOAD_ANALYSE_TEXT
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	ANALYSE
<b>Valid values</b>	ANALYSE or CUSTOM
<b>Examples</b>	TDBS_LOAD_ANALYSE_TEXT=ANALYSE

If unset or set to set to ANALYSE, the LOAD program will always analyse TEXT values in TFORM data for paragraphs and sentences. If set to CUSTOM, the LOAD program will only do such analysis if the TEXT value in the TFORM is not marked up in any way (no sentence or paragraph markers).

## LOG

<b>Function</b>	Specifies an optional location for all log files produced by TRIP.
<b>Usage</b>	TDBS_LOG
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	User's default directory
<b>Valid values</b>	A fully specified directory (path) name
<b>Examples</b>	UNIX:  TDBS_LOG=/opt/trip/v800/log  Windows:  TDBS_HOME=C:\TRIP\v800\log

NOTE: Not to be confused with TBS\_LOG.

Whenever a user submits a batch job (such as a print or index request), TRIP will create a batch log file either in the directory pointed to by LOG if defined, or in the user's default directory.

## LOGIN\_TICKETS

<b>Function</b>	Enables login ticket functionality.
<b>Usage</b>	TDBS_LOGIN_TICKETS
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	N
<b>Valid values</b>	Y(es) or N(o)
<b>Examples</b>	TDBS_LOGIN_TICKETS=Y
<b>See also</b>	TDBS_TICKET_TIMEOUT

A login ticket is a unique, generated, alternate temporary identity for a TRIP user. A login ticket can be requested via the APIs in TRIPjxp and TRIPnpx, and can, once obtained, be used instead of username and password to log in to TRIP as the user to which the ticket was assigned.

The purpose of this feature is to allow an SSO-like behavior in web applications that use TRIP user logins. The user must explicitly log in once, at which time the application requests a login ticket and saves it (e.g. by setting it as a cookie). At subsequent access of the application at a time when the application session has expired, the application can use the login ticket in the cookie to log in the user automatically.

To login using a login ticket, the application enters the login ticket as the password and leaves the username argument blank.

A login ticket is valid for a configurable time, which resets each time the user is logging in. This time is determined by the tdb.conf privileged section configuration symbol TDBS\_TICKET\_TIMEOUT, that (if not set) is 30 days by default.

All assigned login tickets are automatically revoked when the TRIP daemon (tripd) is restarted.

## LOGPRUNE\_INTERVAL

<b>Function</b>	Interval in minutes between scans by the logmain ttool for log files to remove.
<b>Usage</b>	TDBS_LOGPRUNE_INTERVAL
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	60
<b>Valid values</b>	A positive integer number
<b>Examples</b>	TDBS_LOGPRUNE_INTERVAL=60

The `logmaint` utility is automatically used when running TRIPsystem in a container based on the Docker image distribution. It can also be used interactively on regular non-Docker installations on Linux and Solaris. No Windows version is available.

## LOGPRUNE\_MAX\_LOGS

<b>Function</b>	The maximum number of log files of each type to retain unless they are older than max age (as indicated by LOGPRUNE_MAXAGE and LOGPRUNE_MAXAGE_BATCH).
<b>Usage</b>	TDBS_LOGPRUNE_MAX_LOGS
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	100
<b>Valid values</b>	A non-negative integer number
<b>Examples</b>	TDBS_LOGPRUNE_MAX_LOGS=100

The `logmaint` utility is automatically used when running TRIPsystem in a container based on the Docker image distribution. It can also be used interactively on regular non-Docker installations on Linux and Solaris. No Windows version is available.

## LOGPRUNE\_MAXAGE

<b>Function</b>	The maximum number of days to retain log files.
<b>Usage</b>	TDBS_LOGPRUNE_MAXAGE
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	30
<b>Valid values</b>	A positive integer number
<b>Examples</b>	TDBS_LOGPRUNE_MAXAGE=30

The `logmaint` utility is automatically used when running TRIPsystem in a container based on the Docker image distribution. It can also be used interactively on regular non-Docker installations on Linux and Solaris. No Windows version is available.



## LOGPRUNE\_MAXAGE\_BATCH

<b>Function</b>	The maximum number of days to retain logs for successful batch logs. The maximum age for batch job logs containing errors is managed using LOGPRUNE_MAXAGE.
<b>Usage</b>	TDBS_LOGPRUNE_MAXAGE_BATCH
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	7
<b>Valid values</b>	A positive integer number
<b>Examples</b>	TDBS_LOGPRUNE_MAXAGE_BATCH=7

The `logmaint` utility is automatically used when running TRIPsystem in a container based on the Docker image distribution. It can also be used interactively on regular non-Docker installations on Linux and Solaris. No Windows version is available.

## LONG\_PHRASE

<b>Function</b>	Specifies that TRIP should accept entire phrases of any length at input to the BAF file when value is Y(es). When value is N(o) entire phrases longer than 255 characters will not be accepted.
<b>Usage</b>	TDBS_LONG_PHRASE
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	Y
<b>Valid values</b>	Y (yes) or N (no)
<b>Examples</b>	TDBS_LONG_PHRASE=N

NOTE: When the phrase is longer than 255 characters, only the first 255 normalized characters of the entire phrase are indexed. All words in the entire phrase are always indexed.

## MAX\_ALLO\_MEM

<b>Function</b>	Sets maximum allocated memory in Megabytes (Mb) during scan phase of indexing program.
<b>Usage</b>	TDBS_MAX_ALLO_MEM
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	1024 Mb
<b>Valid values</b>	Integer giving memory size in Mb
<b>Examples</b>	TDBS_MAX_ALLO_MEM=1024



## MAX\_THREADS

<b>Function</b>	Determines the number of threads that TRIP can use to parallelize search and display operations against database clusters. If not specified, TRIP will by default use 16 threads for such operations. Parallel execution can be disabled by assigning 0 (zero) to this logical name.
<b>Usage</b>	TDBS_MAX_THREADS
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	16
<b>Valid values</b>	An integer value 4 or larger, or 0 (zero) to disable use of threads.
<b>Examples</b>	TDBS_MAX_THREADS=32

## MIGRATION\_ERRORS\_DIR

<b>Function</b>	Defines a directory where copies for post-mortem analysis of the CONTROL database files are placed after a failure during upgrade install to migrate the CONTROL database to the new TRIP version.
<b>Usage</b>	TDBS_MIGRATION_ERRORS_DIR
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	Fully qualified path to a monitor library file
<b>Examples</b>	UNIX:  TDBS_MIGRATION_ERRORS_DIR=/var/lib/trip/migration_errors

If this variable is not set, TRIP will fall back on the following, in priority order:

1. Subdirectory "migration\_errors" under TBDS\_CTL, if TDBS\_CTL refers to a directory for which the current user has write permission.
2. Subdirectory "migration\_errors" under TBDS\_LOG, if TDBS\_LOG refers to a directory for which the current user has write permission.
3. UNIX Only: Temporary directory "/tmp/tripsystem\_migration\_errors"

In all cases, the files copied for post-mortem analysis will be in a subdirectory whose name is the timestamp for the time at which the error occurred.

## MONITOR\_LIB

<b>Function</b>	Defines a shared / dynamic library to use to handle TRIPsystem monitoring events, thereby enabling the monitoring subsystem which is disabled by default.
<b>Usage</b>	TDBS_MONITOR_LIB
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	Fully qualified path to a monitor library file
<b>Examples</b>	UNIX:  TDBS_MONITOR_LIB=/opt/trip/v800/bin/libmonlog.so  Windows:  TDBS_MONITOR_LIB=C:\TRIP\v800\bin\libmonlog.dll

Event monitoring is a feature that allows the TRIP systems administrator or DBA to output events from TRIP sessions.

Events in this context are:

- Errors in the current session
- Changes to users (e.g. created, deleted)
- Changes and actions on databases (e.g. created, deleted, opened, closed)
- Submitted batch jobs (index jobs, print jobs and global updates)
- Session changes (login, logout)

**NOTE:** Having monitoring enabled may have a small negative performance impact.

## MONITOR\_QPERF

<b>Function</b>	Instructs the monitoring subsystem to emit performance measurements on query executions.
<b>Usage</b>	TDBS_MONITOR_QPERF
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	N
<b>Valid values</b>	Y(es) or N(o)
<b>Examples</b>	TDBS_MONITOR_QPERF=Y



## MONLOG\_FLUSH

<b>Function</b>	Instructs the monitoring handler library (libmonlog) to always flush writes to the log, making them available as soon as they are occurring.
<b>Usage</b>	<code>TDBS_MONLOG_FLUSH</code>
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	N
<b>Valid values</b>	Y(es) or N(o)
<b>Examples</b>	<code>TDBS_MONLOG_FLUSH=Y</code>

Note that enabling this option will have a small negative performance impact.



## MONLOG\_MONOLITHIC

<b>Function</b>	Determines if the monitoring handler library (libmonlog) is to use a single log file for all TRIP sessions, or if session-specific log files are to be written.
<b>Usage</b>	TDBS_MONLOG_MONOLITHIC
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	Y
<b>Valid values</b>	Y(es) or N(o)
<b>Examples</b>	TDBS_MONLOG_MONOLITHIC=N
<b>See also</b>	TDBS_MONITOR_LIB TDBS_MONLOG_MONOLITHIC_PERIOD

## MONLOG\_MONOLITHIC\_PERIOD

<b>Function</b>	Determines if the time period for monolithic log files from the monitoring handler library (libmonlog). A new log file is created for each new period.
<b>Usage</b>	TDBS_MONLOG_MONOLITHIC_PERIOD
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	DAY
<b>Valid values</b>	<ul style="list-style-type: none"> <li>• HOUR</li> <li>• DAY</li> <li>• WEEK</li> <li>• MONTH</li> <li>• YEAR</li> </ul>
<b>Examples</b>	TDBS_MONLOG_MONOLITHIC_PERIOD=DAY
<b>See also</b>	TDBS_MONLOG_MONOLITHIC

The name of the monolithic log file is determined the chosen period:

```

HOUR   eventlog_<year><month><day><hour><min><sec>.log
DAY     eventlog_<year><month><day>.log
WEEK    eventlog_<year><dayofyear>.log
MONTH   eventlog_<year><month>.log
YEAR    eventlog_<year>.log
  
```

Year is always a 4-digit number. The values for month, day, hour, minute and second are all zero-padded 2-digit numbers. The value for 'dayofyear' used for WEEK time period is a zero-padded 3-digit number that denotes the day of the year for the Monday in the current week.

If monolithic logging is disabled, the log file format from the libmonlog library will be:

```
eventlog_<year><month><day><hour><min><sec>_<process_id>.log
```

which makes the log files named after TRIP process ID and the second at which the process started.

## MONLOG\_TSTAMP

<b>Function</b>	Determines if a year-to-second time stamp should be included log entries from the monitoring handler library (libmonlog).
<b>Usage</b>	TDBS_MONLOG_TSTAMP
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	N
<b>Valid values</b>	Y(es) or N(o)
<b>Examples</b>	TDBS_MONLOG_TSTAMP=Y
<b>See also</b>	TDBS_MONITOR_LIB

## NO\_GLBUPD\_INDEX

<b>Function</b>	When doing Global Update via the glbupd script, an automatic indexing is performed by default after the global update is done. To avoid this, set this logical name to Y.
<b>Usage</b>	TDBS_NO_GLBUPD_INDEX
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	N
<b>Valid values</b>	Y (yes) or N (no)
<b>Examples</b>	TDBS_NO_GLBUPD_INDEX=Y

## OVFBUSZ

<b>Function</b>	Index tuning logical name for specifying the size of the overflow file buffer (*.STO and *.TPO).
<b>Usage</b>	TDBS_OVFBUSZ
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	32 kilobytes
<b>Valid values</b>	From 2 to 32 kilobyte
<b>Examples</b>	TDBS_OVFBUSZ=10
<b>See also</b>	TDBS_TERMLM, TDBS_TRMBUSZ

The indexing process is tunable so that it may be biased towards systems with large amounts of memory.

Whilst the indexing process will continue to work well in memory-constrained environments, administrators of large systems will see significant performance improvements when tuning appropriately.

When indexing large data collections, e.g. new databases, large batch updates, etc., it can be extremely advantageous to tune these parameters generously. For best performance, set all parameters to their maximum values, although this requires significant memory resource in order not to fail.

A system equipped with more than 1GB of RAM, dedicated to the TRIP indexing task, is required in order for the maximum settings to be used successfully. Setting the parameters to their maximum values on a heavily loaded, or memory constrained system will be counter-productive as the index task will then execute far more slowly than if the parameters were left at their default values.

## PRC (UNIX only)

<b>Function</b>	Specifies the location of TRIP printer control files.
<b>Usage</b>	TDBS_PRC
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	The PRC directory in the TRIP tree structure
<b>Valid values</b>	A fully specified directory (path) name
<b>Examples</b>	TDBS_PRC=/opt/trip/v800/prc

Printer control files are used by TRIP to direct printed output to a correct printer, and to ensure that such things as the character set, the highlighting characters, the initialization sequence, etc. are correct for that printer. There are two types of printer control files:

File Extension	File Type
*.PRC	Printer character set control files
*.PRN	Printer name control files

The master is the 'PRN' file, which may name a 'PRC' file to be used for character translation during the printing process. For details, see the printer control section in this chapter.

## PRINT

<b>Function</b>	Specifies the printer queue to which hard copy output is to be spooled.
<b>Usage</b>	TDBS_PRINT
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	UNIX: Yes  Windows: No
<b>Default value</b>	UNIX: lp  Windows: None
<b>Valid values</b>	UNIX: Any valid printer name.  Windows: Any valid printer mountable by the TRIP Daemon service's owner.
<b>Examples</b>	UNIX:  TDBS_PRINT=lp0  Windows:  TDBS_PRINT=\\ServerName\PrinterName

Once print preparation has completed, the hard copy output from the print job is directed by TRIP to the queue, or device, named by the PRINT variable. For validity, you should be able to use the name that you specify for the PRINT variable in the following commands:

UNIX:        `lpr -P<xyz>`

Windows:   `Print /D:\\ServerName\PrinterName <xyz>`

where 'xyz' is the path and filename given to the PRINT variable.

**NOTE:** For specific details on setting up TRIP printing in a Windows environment, refer to the section entitled, "Configuring a printer for TRIP", in the file "TRIPsystem\_Installation\_Guide\_Win.pdf", which is included in the "doc" directory of the TRIPsystem installation.

## PRINT\_EMPTY\_FILE

<b>Function</b>	Specifies the result of printing files from empty result sets.
<b>Usage</b>	TDBS_PRINT_EMPTY_FILE
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	N
<b>Valid values</b>	Y(es) or N(o)
<b>Examples</b>	TDBS_PRINT_EMPTY_FILE=N

When this logical name is set to N (no), an empty result set will not generate any empty files, just an error message saying that the print result is empty. Specifying the value as Y will cause the generation of an empty file for empty result sets.



## PRINT\_EMPTY\_TFORM

<b>Function</b>	Specifies the result of printing TFORM files from empty result sets.
<b>Usage</b>	TDBS_PRINT_EMPTY_TFORM
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	N
<b>Valid values</b>	Y(es) or N(o)
<b>Examples</b>	TDBS_PRINT_EMPTY_TFORM=N

When this logical name is set to N (no), an empty result set will not generate any empty files, just an error message saying that the print result is empty. Specifying the value as Y will cause the generation of an empty file for empty result sets.

## PRINTUSER (Windows only)

<b>Function</b>	Specifies the user to be granted printer access to the printer defined by TDBS_PRINT.
<b>Usage</b>	TDBS_PRINTUSER
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	Any valid user name
<b>Examples</b>	TDBS_PRINTUSER=DomainName\UserName

**NOTE:** For specific details on setting up TRIP printing in a Windows environment, refer to the section entitled, "Configuring a printer for TRIP", in the file "TRIPsystem\_Installation\_Guide\_Win.pdf", which is included the "doc" directory of the TRIPsystem installation.

## RANGE\_CHECK

<b>Function</b>	If set to Y (yes, the default), index information about value ranges will be used when searching in numeric fields (Integer, Number, Date and Time).
<b>Usage</b>	TDBS_RANGE_CHECK
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	Y
<b>Valid values</b>	Y (yes) or N (no)
<b>Examples</b>	TDBS_RANGE_CHECK=Y

Setting TDBS\_RANGE\_CHECK to N (no) will disable the value range check for numeric fields. As a disabled value range check will have a negative performance impact for numeric searches, this should only be done if TRIP support so advises.

## REFRESH\_TOKEN\_EXPIRATION

<b>Function</b>	Defines the expiration time in seconds for issued refresh tokens.
<b>Usage</b>	TDBS_REFRESH_TOKEN_EXPIRATION
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	2592000
<b>Valid values</b>	A positive integer
<b>Examples</b>	TDBS_REFRESH_TOKEN_EXPIRATION=2592000
<b>See Also</b>	TDBS_ACCESS_TOKEN_EXPIRATION

This variable defines the life span in seconds for the single-use refresh tokens, which are used to renew access tokens.

## REQUIRE\_APIKEY

<b>Function</b>	Defines the extent to which TRIPsystem requires a valid API key for use of the token features.
<b>Usage</b>	TDBS_REQUIRE_APIKEY
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	ALWAYS
<b>Valid values</b>	ALWAYS, TOKENS, REFRESH, NO
<b>Examples</b>	TDBS_REQUIRE_APIKEY=ALWAYS

This privileged TRIPsystem environment variable is used to determine if and to which degree API Keys are used and required. The following values are valid for this variable:

<b>ALWAYS</b>	An API key is required for creating a new token pair, refreshing a token pair, and for login using an access token. An API key is also required for token administration (e.g. revocation) if requested via the network. This value is the default if the variable is not explicitly set.
<b>TOKENS</b>	An API key is required for creating a new token pair and for refreshing a token pair. Login using an access token does not require an API key.
<b>REFRESH</b>	A valid API key is only required for refreshing a token pair.
<b>NO</b>	API keys are not required by any token operation.

## RESTART

<b>Function</b>	Specifies whether TRIP should attempt to restart from saved SIF.
<b>Usage</b>	TDBS_RESTART
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	Yes/No
<b>Examples</b>	TDBS_RESTART=NO

When TRIP starts, it checks to see if there are any old sessions stored in SIF files. If so, by default it will open the SIF and restore the searches performed during that saved session. If, however, you do not wish old SIFs to be used, you can define the RESTART variable to be 'NO' and the old SIFs will be ignored by TRIP.

For SIF locations, see the definition of the SIF variable.

## RO\_USER\_PROCEDURES

<b>Function</b>	<p>If set to Y, this logical name allows private user procedures to be executable by other users. Other users will also be able to read procedure properties such as the comment when such information is requested via the API. Procedure lists such as can be seen in TRIPclassic and TRIPmanager are not affected by this.</p> <p>This does not apply to procedures owned by SYSTEM. Such procedures remain strictly private.</p>
<b>Usage</b>	TDBS_RO_USER_PROCEDURES
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	N
<b>Valid values</b>	Y(es) or N(o)
<b>Examples</b>	TDBS_RO_USER_PROCEDURES=Y

## SCRATCH

<b>Function</b>	Specifies a scratch directory for certain TRIP operations.
<b>Usage</b>	TDBS_SCRATCH
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	A fully specified directory (path) name
<b>Examples</b>	UNIX:  TDBS_SCRATCH=/opt/trip/v800/scratch  Windows:  TDBS_SCRATCH=C:\TRIP\v800\scratch

During the index procedure for a database, TRIP needs to be able to write certain temporary files that it uses for virtual memory management. These files are written to the directory specified by the SCRATCH variable. If it is not defined, the index procedure will temporarily define it to be the user's current working directory.

The SCRATCH variable must also be defined whenever you attempt to invoke the MODCON executable for upgrading the CONTROL database between different TRIP versions.

**NOTE:** All temporary files will be stored in the scratch area if this variable has been defined.



## SIF

<b>Function</b>	Specifies the optional location of all SIFs.
<b>Usage</b>	TDBS_SIF
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	Current working directory
<b>Valid values</b>	A fully specified directory (path) name plus an optional filename
<b>Examples</b>	UNIX:  TDBS_SIF=/opt/trip/v800/sif  Windows:  TDBS_SIF=C:\TRIP\v800\sif

During the startup of TRIP, the system creates a session index file or SIF for each user. This SIF is used to record both the searches which are performed and the results gained, so that a previous session can be restarted without having to rerun all of the searches involved.

Depending on the searches the user performs, these files can become very large, and so it can be useful to move them to a location where there is sufficient room for growth.

When the SIF variable is defined, TRIP will create the SIF using its definition as either the complete name of the file, or the directory into which to write the SIF with the name username.SIF, where username is the name of the TRIP user.

## SORT

<b>Function</b>	Specifies the sorting collation sequence to be used.
<b>Usage</b>	TDBS_SORT
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	Defined during installation (same as LANG)
<b>Valid values</b>	Any three-letter code from the table below.
<b>Examples</b>	TDBS_SORT=GER

The valid values for LANG are:

Code	Meaning
ENG	ENGLISH
FIN	FINnish
GER	GERman
NOR	NORwegian
SWE	SWEdish

The SORT variable defines the collation sequence which TRIP will use when sorting data, that is, the order in which diacritically-altered characters (ñ, î, å, ö, etc.) will appear. Certain languages expect diacritically-modified characters to sort differently than their Latin equivalent.

## SPAWN

<b>Function</b>	Specifies whether it is possible to execute external scripts or programs from within TRIP.
<b>Usage</b>	TDBS_SPAWN
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	Y
<b>Valid values</b>	Y (yes) or N (no)
<b>Examples</b>	TDBS_SPAWN=N
<b>See also</b>	TDBS_EXEC_SCRIPT, TDBS_COM, TRIP_ENTER_OS, TBS_AT_CCL

## STO\_LOCATION

<b>Function</b>	Specifies the location (name and path) of the temporary STO file used by the indexing program. (Used only when indexing is done via the index script).
<b>Usage</b>	TDBS_STO_LOCATION
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	Full path and name of file.
<b>Examples</b>	TDBS_STO_LOCATION=/tmp/db2.sto

## STOP\_WORDS

<b>Function</b>	Specifies limits for adaptive stop words used when searching with the Fuzz CCL command.
<b>Usage</b>	TDBS_STOP_WORDS
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None (no stop words)
<b>Valid values</b>	X, Y  Where X and Y are integer values between 0 and 100 indicating a percentage.
<b>Examples</b>	TDBS_STOP_WORDS=75,10

Here **X** represents the percentage of records in which a word must occur and **Y** represents the average number of occurrences, per record, of the same word, before it becomes a stop word. If the thresholds set by **X** and **Y** are exceeded, the word in question will be automatically defined as a stop word.

**NOTE:** Setting both **X** and **Y** to 100 effectively gives no stop words. The same result is also achieved by unsetting/undefining the TDBS\_STOP\_WORDS logical name.

## STRICT\_UNICODE

<b>Function</b>	Enable/disable non-Unicode sessions from data modification access to Unicode databases.
<b>Usage</b>	TDBS_STRICT_UNICODE
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	Y
<b>Valid values</b>	Y (yes) or N (no)
<b>Examples</b>	TDBS_STRICT_UNICODE=N

The default behavior is to disallow modification of Unicode databases from non-Unicode sessions. The reason for that is to protect from data corruption that otherwise can occur if the text being edited cannot be fully represented in the current session character set.

In scenarios where it is certain that the Unicode databases in use only contains fully compatible text (e.g. a Unicode database that only contains Latin-1 compatible data) this logical name may be set to N.

## SUPERMAN

<b>Function</b>	Gives complete access to all TRIP objects to the user SYSTEM.
<b>Usage</b>	TDBS_SUPERMAN
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	Y (yes) or N (no)
<b>Examples</b>	TDBS_SUPERMAN=Y



## SYS

<b>Function</b>	Specifies the location of the TRIP system files.
<b>Usage</b>	TDBS_SYS
<b>Looked for in</b>	Privileged and non-privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	The SYS directory in the TRIP tree structure
<b>Valid values</b>	A fully specified directory (path) name
<b>Examples</b>	UNIX:  TDBS_SYS=/opt/trip/v800/sys  Windows:  TDBS_SYS=C:\TRIP\v800\sys

Many of the functions of TRIP make use of system definition files, such as language-specific message codes, menu labels, etc. All of these files are located using the SYS variable. Be careful if you are considering reassigning this variable. Without certain of the files in the SYS directory, the TRIP system cannot start at all.



## TERMINAL (UNIX only)

<b>Function</b>	Specifies the terminal type.
<b>Usage</b>	TRIP_TERMINAL
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	Value of UNIX TERM environment variable.
<b>Valid values</b>	Any valid terminal identifier.
<b>Examples</b>	TDBS_TERMINAL=vt200

In TRIP for UNIX only, the TERMINAL variable points TRIP at a terminal driver file, located in the TRM directory in the TRIP tree structure called terminal.TRL. Terminal is the value defined for the TERMINAL variable.

There are several terminal definition files delivered with the TRIPclassic system, but the one most commonly used is vt200.

The trm files, also in the TRM directory in the TRIP tree structure, contain the definitions for the escape sequences, for every possible terminal “action”, for a given terminal type.

To define a new terminal type, e.g. abc123, make a copy of one of the already existing terminal definition files and modify the new copy to the requirements of your terminal type; then move it to the <tdbs\_trm> directory and run the utility program <tdbs\_exe>/trmmake with “abc123” as its only parameter. This will “compile” the terminal definition file to a binary format which TRIPclassic will use to handle your terminal type (see separate instructions on the structure of the terminal definition file).

*Note:*

- <tdbs\_exe> is the value of TDBS\_EXE in the tdbb.conf file
- <tdbs\_trm> is the value of TDBS\_TRM in the tdbb.conf file

If the TERMINAL variable defines a terminal type that TRIP does not recognise, the user will be prompted to provide a terminal identifier until an acceptable response is gained.

## TERMLM

<b>Function</b>	Index tuning logical name for specifying the number of terms held by the internal indexing structure before it is written to the STO file.
<b>Usage</b>	TDBS_TERMLM
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	131072 kterms
<b>Valid values</b>	From 64 to 131072 kterms
<b>Examples</b>	TDBS_TERMLM=1024
<b>See also</b>	TDBS_OVFBUFSZ, TDBS_TRMBUFSZ

The indexing process is tunable so that it may be biased towards systems with large amounts of memory.

Whilst the indexing process will continue to work well in memory-constrained environments, administrators of large systems will see significant performance improvements when tuning appropriately.

When indexing large data collections, e.g. new databases, large batch updates, etc., it can be extremely advantageous to tune these parameters generously. For best performance, set all parameters to their maximum values, although this requires significant memory resource in order not to fail.

A system equipped with more than 1GB of RAM, dedicated to the TRIP indexing task, is required in order for the maximum settings to be used successfully. Setting the parameters to their maximum values on a heavily loaded, or memory constrained system will be counter-productive as the index task will then execute far more slowly than if the parameters were left at their default values.

## THESAURUS\_DEFAULT

<b>Function</b>	Defines the default mapping for DEFINE THESAURUS command when used in its simplest form.
<b>Usage</b>	TDBS_THESAURUS_DEFAULT
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	PHRASE
<b>Valid values</b>	PHRASE, WORD
<b>Examples</b>	TDBS_THESAURUS_DEFAULT=PHRASE

This logical name controls TRIP's behavior when using the simplest form of DEFINE THESAURUS, e.g.:

```
DEFINE THESAURUS=THESALI
```

The default mapping is to use full phrase search, which corresponds to the value PHRASE of this logical name and for which the DEFINE THESAURUS expression looks like this fully written out:

```
DEFINE THESAURUS=PHRASE:THESALI.PHRASE:'PHRASE'
```

If full phrase mapping is not desirable as default, use the value WORD with this logical name. This makes the default of the DEFINE THESAURUS expression look like this fully written out:

```
DEFINE THESAURUS=PHRASE:THESALI.PHRASE:PHRASE
```

The phrase based mapping replaced the word based mapping as default for DEFINE THESAURUS in version 8.3-1.

## TICKET\_TIMEOUT

<b>Function</b>	Determines the time that can pass without a user logging in before their login ticket is automatically revoked.
<b>Usage</b>	TDBS_TICKET_TIMEOUT
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	30d
<b>Valid values</b>	<p>An integer indicating the time period followed by a letter suffix indicating the length of the period. Valid suffixes are:</p> <ul style="list-style-type: none"><li>• <b>d</b> – Indicates number of days (default)</li><li>• <b>h</b> – Indicates number of hours</li><li>• <b>m</b> – Indicates number of minutes</li><li>• <b>s</b> – Indicates number of seconds</li></ul>
<b>Examples</b>	TDBS_TICKET_TIMEOUT=48h
<b>See also</b>	TDBS_LOGIN_TICKETS

## TOKEN\_KEY

<b>Function</b>	Specifies the location of the token encryption key.
<b>Usage</b>	TDBS_TOKEN_KEY
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	A fully-qualified file name
<b>Examples</b>	TDBS_TOKEN_KEY=/var/lib/trip/.keys/tokenkey

This privileged TRIPsystem environment variable defines the fully qualified path to the server-side file that contains the 256-bit (32-byte) AES encryption key used to encrypt the access and refresh tokens as returned to the calling application.

This file must contain a cryptographically secure random number, and can be created using the following OpenSSL command:

```
$ openssl rand -out /path/to/my/token/key 32
```

Rotating this key immediately invalidates all issued access and refresh tokens.

The absence of a valid value for this variable means that token access is disabled.

## TRM (UNIX only)

<b>Function</b>	Specifies the location of the terminal driver files.
<b>Usage</b>	TDBS_TRM
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	The TRM directory in the TRIP tree structure
<b>Valid values</b>	A fully-specified directory (path) name
<b>Examples</b>	TDBS_TRM=/opt/trip/v800/trm

When TRIP is attempting to locate a terminal driver file, it does so by using the value of the TRM variable to specify the directory containing the drivers.

## TRMBUFSZ

<b>Function</b>	Index tuning logical name for specifying the .BUT file buffer size in kilobytes.
<b>Usage</b>	TDBS_TRMBUFSZ
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	8 kilobytes
<b>Valid values</b>	From 2 to 512 kilobytes
<b>Examples</b>	TDBS_TRMBUFSZ=500
<b>See also</b>	TDBS_OVFBUFSZ, TDBS_TERMLM

The indexing process is tunable so that it may be biased towards systems with large amounts of memory.

Whilst the indexing process will continue to work well in memory-constrained environments, administrators of large systems will see significant performance improvements when tuning appropriately.

When indexing large data collections, e.g. new databases, large batch updates, etc., it can be extremely advantageous to tune these parameters generously. For best performance, set all parameters to their maximum values, although this requires significant memory resource in order not to fail.

A system equipped with more than 1GB of RAM, dedicated to the TRIP indexing task, is required in order for the maximum settings to be used successfully. Setting the parameters to their maximum values on a heavily loaded, or memory constrained system will be counter-productive as the index task will then execute far more slowly than if the parameters were left at their default values.

## UNKNOWN\_FIELDS

<b>Function</b>	Defines how TRIP treats references to unknown fields in cluster search expressions.
<b>Usage</b>	TDBS_UNKNOWN_FIELDS
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	IGNORE, NOHITS
<b>Examples</b>	TDBS_UNKNOWN_FIELDS=IGNORE

This logical name will, if set, override the default behavior of failing a cluster search command when an unknown field is included in the search condition.

Setting TDBS\_UNKNOWN\_FIELDS to `IGNORE` causes TRIP to completely ignore non-existent fields in CCL search conditions against database clusters. The condition as executed will be as if the field was not specified to begin with.

Setting TDBS\_UNKNOWN\_FIELDS to `NOHITS`, causes TRIP to evaluate the part of the CCL search condition that references the field to an empty search result. Whether the whole search condition results in hits or not depends on the parts of the condition unaffected by this variable are expressed.



## XPI\_LOGLEVEL

<b>Function</b>	Enables logging from the XPI library, which implements the high-level API endpoints used by TRIPmanager, TRIPjxp and TRIPnpx.
<b>Usage</b>	TDBS_XPI_LOGLEVEL
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	DEBUG
<b>Examples</b>	TDBS_XPI_LOGLEVEL=DEBUG
<b>See also</b>	TDBS_APILOG, TDBS_LOG

This logical name will, if set, cause process-specific log files to be written to the directory indicated by the TDBS\_LOG logical name. The log file names will have “xpilog\_” as prefix.

Having this logical name enabled causes will significantly reduce TRIP’s performance and is intended for troubleshooting purposes only. For this reason, TDBS\_XPI\_LOGLEVEL is to be set only when it is necessary to produce a log of the calls made by an application, usually at the request of TRIP support.

## TRIPserver Logical Names (TBS\_)

### ASE

<b>Function</b>	Flag for automatic ASE calls at start/stop of tbserver.
<b>Usage</b>	TBS_ASE
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	0
<b>Valid values</b>	Bit 0 = 1: call server ASE (AseStartTRIPserver) after start of tbserver  Bit 1 = 1: call server ASE (AseStopTRIPserver) before stop of tbserver
<b>Examples</b>	TBS_ASE=2

## AT\_CCL

<b>Function</b>	Permission to execute the CCL command @... (i.e. to execute OS scripts).
<b>Usage</b>	TBS_AT_CCL
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	0 (no)
<b>Valid values</b>	1 (yes) or 0 (no)
<b>Examples</b>	TBS_AT_CCL=1
<b>See also</b>	TDBS_COM, TDBS_SPAWN

## CCL\_FILE\_DIR

**Function** If this symbol is set, CCL commands that read or write files on the TRIP server (EXPORT, IMPORT, LOAD and PRINT) will when called with an unqualified file name argument (file name only, with no directory path), be automatically adjusted so that the files are written to the directory that this symbol names.

If this variable is used, the contents of the directory it refers to should regularly be removed by a scheduled job (e.g., cron).

See also TBS\_CCL\_NO\_RELATIVE

**Usage** TBS\_CCL\_FILE\_DIR

**Looked for in** Non-privileged section

**Defined by default?** No

**Default value** None

**Valid values** A fully qualified path to a local directory on the TRIP server.

**Examples** UNIX/Linux:

```
TBS_CCL_FILE_DIR=/var/lib/trip/output
```

Windows:

```
TBS_CCL_FILE_DIR=C:\trip\output
```

## CCL\_NO\_RELATIVE

<b>Function</b>	<p>Specifies that all CCL commands that read or write files on the TRIP server must use fully qualified paths.</p> <p>If the TBS_CCL_FILE_DIR is also set, relative paths will be automatically adjusted so that specified files are read and written to that directory.</p> <p>If this symbol is set but TBS_CCL_FILE_DIR is not, the specification of relative paths in CCL commands will result in an error.</p>
<b>Usage</b>	TBS_CCL_NO_RELATIVE
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	TRUE or FALSE
<b>Examples</b>	TBS_CCL_NO_RELATIVE=TRUE

## COMFORTER

<b>Function</b>	TRIPserver comforter interval.
<b>Usage</b>	TBS_COMFORTER
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	5
<b>Valid values</b>	Time period (integer value in seconds) between calls to comforter; if set to 0 there will be no comforter calls at all.
<b>Examples</b>	TBS_COMFORTER=0



## DIR (UNIX only)

<b>Function</b>	Transfer directory.
<b>Usage</b>	TBS_DIR
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	Current directory
<b>Valid values</b>	Fully specified directory path
<b>Examples</b>	TBS_DIR=/opt/trip/tbs



## HOSTINI

<b>Function</b>	Path of host-ini-file
<b>Usage</b>	TBS_HOSTINI
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None (see also note below)
<b>Valid values</b>	Fully specified path of host-ini-file WITHOUT file name
<b>Examples</b>	UNIX:  TBS_HOSTINI=/opt/trip/server  Windows:  TDBS_SYS=C:\TRIP\server

**NOTE:** When using TBS\_HOSTINI, only the file name should be specified in the call to the start session routine. You can specify the full path and file name when calling the start session routine, and in this case, you **MUST NOT** specify TBS\_HOSTINI at all.

In the host-ini-file you can define values for environment variables. These values will be added to or replacing those already in the TRIP config (former TRIPrcs) file. You cannot set or override values in the Privileged section of the config file in this way.



## LOG

<b>Function</b>	Specifies the generation of an optional logfile for all TBserver transactions.
<b>Usage</b>	TBS_LOG
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	None
<b>Valid values</b>	0 to 3 in increasing levels of verbosity:  0 - no logging 1 - log call of TDB routines only 2 - log parameters too 3 - log communication too

**Examples** TBS\_LOG=3

### Notes:

- Due to the constant disk access required for updating the logs, TBserver logging causes a considerable reduction in TRIP's performance and is intended for troubleshooting purposes only; for this reason TBS\_LOG is to be set only when it is necessary to produce a log of the transactions taking place between the client and server parts of a TRIP client/server application, usually at the request of TRIP support.
- As soon as the required log files have been produced, either delete the TBS\_LOG logical name, or reset it back to a value of zero.
- Log files will be produced by all TBserver sessions that commence once the TBS\_LOG logical name has been defined. Once the TBS\_LOG logical name has been deleted, or set to zero, new TBserver sessions will run without creating logs.
- Any log files produced will be created in the directory specified by the logical name TDBS\_LOG and will have a name format similar to 'TBserver\_nnnn.log'; where nnnn is a unique numerical date/time stamp to avoid file name clashes.
- Care should be taken not to confuse TBS\_LOG with TDBS\_LOG

## MAP (UNIX only)

<b>Function</b>	Specifies logging of start/stop of tbserver in tbserver.map.		
<b>Usage</b>	TBS_MAP		
<b>Looked for in</b>	Non-Privileged section		
<b>Defined by default?</b>	Yes		
<b>Default value</b>	2		
<b>Valid values</b>	Logging level:		
	0	-	no logging
	1	-	log only start
	2	-	log stop too
<b>Examples</b>	TBS_MAP=1		

## MAP\_DIR (UNIX only)

<b>Function</b>	Specifies path to tbserver.map.
<b>Usage</b>	TBS_MAP_DIR
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	TDBS_LOG or TDBS_SCRATCH if defined; otherwise /tmp
<b>Valid values</b>	Path to directory
<b>Examples</b>	TBS_MAP_DIR=/disk1/log



## SCRATCH (UNIX only)

<b>Function</b>	Specifies working directory of tbserver.
<b>Usage</b>	TBS_SCRATCH
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	/tmp
<b>Valid values</b>	Path to directory
<b>Examples</b>	TBS_SCRATCH=/trip/tbserver/tmp



## TIMEOUT

<b>Function</b>	Sets TBserver timeout (in seconds)
<b>Usage</b>	TBS_TIMEOUT
<b>Looked for in</b>	Non-Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	No timeout
<b>Valid values</b>	Timeout integer value in seconds
<b>Examples</b>	TBS_TIMEOUT=10



## TripDaemonHost (Windows only)

<b>Function</b>	Identifies the machine on which the TRIP Daemon to use is installed.
<b>Usage</b>	<code>TripDaemonHost</code>
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	localhost
<b>Valid values</b>	Any machine name or IP-address that refer to the local machine.
<b>Examples</b>	<code>TripDaemonHost=localhost</code>



## TripDaemonPort (Windows only)

<b>Function</b>	Identifies the UDP port number at which the TRIP Daemon can be reached.
<b>Usage</b>	<code>TripDaemonPort</code>
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	4711
<b>Valid values</b>	Any valid port UDP number not in use by other software on the local system.
<b>Examples</b>	<code>TripDaemonPort=4712</code>



## TripNetPort (Windows only)

<b>Function</b>	Identifies the TCP port number to which the TRIPnet Daemon listens for connections at.
<b>Usage</b>	<code>TripNetPort</code>
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	No
<b>Default value</b>	23457
<b>Valid values</b>	Any valid TCP port number not in use by other software on the local system.
<b>Examples</b>	<code>TripNetPort=23457</code>

This variable (if defined) takes precedence over the definition of the pctdbs service in the services file.



## UNIXLOGIN (UNIX only)

<b>Function</b>	Change tbserver user to specified user (, password)
<b>Usage</b>	TBS_UNIXLOGIN
<b>Looked for in</b>	Privileged section
<b>Defined by default?</b>	Yes
<b>Default value</b>	Run as “root”
<b>Valid values</b>	UNIX user[,password]
<b>Examples</b>	TBS_UNIXLOGIN=user[,password]

