

TDAppTools v3.1

(see [Version history](#) for details on changes)

For a complete overview of features and how to integrate TDAppTools into your projects several videos have been created instead of a complete written manual.

Please watch the videos on YouTube TDWiki channel:

https://www.youtube.com/playlist?list=PL5Q_7YSEiYKA1UbLcZJWXm0wjQfwYxY5x

Quick overview of files

Each TD version has its own set of files to be used which are part of the TDAppTools package.

Pick the correct version based on the TD version you use for your application.

Each specific folder contains these files:

TDAppTools.apd / TDAppTools_x64.apd

Runtime file to deploy along with your application runtime.

It contains the TDAppTools features as a module (dynamlib).

For TD x64 versions, the file needed is TDAppTools_x64.apd.

TDAppTools.dll / TDAppTools_x64.dll

Runtime file to deploy along with your application runtime.

This file is a helper dll file which contains specific functionalities to make some of the TDAppTools features possible. This dll is needed by TDAppTools.apd and must be deployed with the dynamlib for TDAppTools to work.

The dll is compatible with all TD versions starting with TD 2.1 and up. TDAppTools reuses the same TDAppTools.dll in each TD version.

For TD x64 versions, the file needed is TDAppTools_x64.dll.

TDAppTools.apl / TDAppTools_x64.apl

TD Source library file to include in your application project source.

This is the interface to TDAppTools features. By calling the available global functions from this library in your project, the TDAppTools features can be started from within your application.

For TD x64 versions, the file needed is TDAppTools_x64.apl.

How to integrate in your projects

Copy these two files to the runtime folder of your application:

(TD x86 versions):

- TAppTools.apd
- TAppTools.dll

(TD x64 versions):

- TAppTools_x64.apd
- TAppTools_x64.dll

When you do not deploy TAppTools along with your application for the end user, you can instead copy the apd and dll to the TD installation folder on the development system.

Make sure in any case that when you want to start TAppTools features, the application must be able to locate these files on the system.

When the files are in the application runtime folder (along with your application exe) this will automatically be the case.

When deployed at another location, make sure the PATH setting on the system points to the folder where these files are deployed.

In your source project, include one library in the libraries section:

(TD x86 versions):

- TAppTools.apl

(TD x64 versions):

- TAppTools_x64.apl

This will add the needed functions to your project to start TAppTools features.

Never include the TAppTools.apd or TAppTools_x64.apd directly in your project, but use the include library.

Only then all TAppTools features will be available.

The library is small and will not interfere or enlarge your project. It only contains the interface to TAppTools.

Starting TAppTools features from your application

Now, determine how you want to start TAppTools features from your application.

Decide which users should be able to use TAppTools and how the user can access the TAppTools features.

For instance, depending on user authorizations in your application, a menu item or button could be visible to start TAppTools or maybe a hidden egg to enable it.

The following TAppTools features are available:

Gui Inspector

Purpose of this tool is to inspect the GUI structure of the running application and get details of GUI objects. It offers ways to change attributes at runtime and export data from child objects.

To start Gui Inspector from TAppTools:

The global function *PALTDAppTools_ShowGuiInspector* (from TAppTools.apl) will start the Gui Inspector and present the main window.

Window Handle *PALTDAppTools_ShowGuiInspector*(*phWndParent*, *pbEnableSaveSettings*, *phWndWindow*)

Starts the Gui Inspector.

Parameter	Type	Description
<i>phWndParent</i>	Window Handle	(OPTIONAL) Parent window of the Gui Inspector or hWndNULL
<i>pbEnableSaveSettings</i>	Boolean	TRUE will save settings to registry FALSE will only keep settings for the current session, no registry use
<i>phWndWindow</i>	Window Handle	(OPTIONAL) Immediately show info of this window when inspector starts or hWndNULL

Return value:

The window handle of the Gui Inspector dialog. hWndNULL when TAppTools is not installed

Sql Monitor

Purpose of this tool is to trace SQL actions from the running application and inspect the result sets. It offers changing of bind variables and SQL statements and exporting result sets.

To start Sql Monitor from TAppTools:

The global function *PALTDAppTools_ShowSqlMonitor* (from TAppTools.apl) will start Sql Monitor and present the main window.

Window Handle *PALTDAppTools_ShowSqlMonitor*(*phWndParent*,*pbEnableSaveSettings*,
pbEditingAllowed, *pbEditingAllowed*)

Starts Sql Monitor.

Parameter	Type	Description
<i>phWndParent</i>	Window Handle	(OPTIONAL) Parent window of Sql Monitor or hWndNULL
<i>pbEnableSaveSettings</i>	Boolean	TRUE will save settings to registry FALSE will only keep settings for the current session, no registry use

<code>pbEditingAllowed</code>	Boolean	TRUE -> in break mode, editing of SQL statement and data edit is allowed
-------------------------------	---------	--

Return value:

The window handle of Sql Monitor main window. `hWndNULL` when `TDAppTools` is not installed

Profiling Controller (available on TD 7.2 and higher)

Purpose of this tool is to control the TD profiling feature using a simple GUI. This profiling feature was introduced in TD 7.2. Using the controller, you can start/pause/resume and stop profiling.

To start Profiling Controller from `TDAppTools`:

The global function `PALTDAppTools_ShowProfiler` (from `TDAppTools.apl`) will start the Profiling Controller window.

Window Handle `PALTDAppTools_ShowProfiler(phWndParent, pbBlockApp, pbEnableSaveSettings)`

Starts TD profiler controller window.

Parameter	Type	Description
<code>phWndParent</code>	Window Handle	(OPTIONAL) Parent window of the Profiling Controller or <code>hWndNULL</code>
<code>pbBlockApp</code>	Boolean	TRUE -> starts tool while application waits for it to end. FALSE -> starts asynchronously
<code>pbEnableSaveSettings</code>	Boolean	TRUE will save settings to registry FALSE will only keep settings for the current session, no registry use

Return value:

The window handle of the Profiling Controller window. `hWndNULL` when `TDAppTools` is not installed

App Info

Purpose of this tool is to inspect and/or change general application (process) and system settings which are in use. It also offers a simple WMI query tool.

To start App Info from `TDAppTools`:

The global function `PALTDAppTools_ShowAppInfo` (from `TDAppTools.apl`) will start the App Info window.

Window Handle `PALTDAppTools_ShowAppInfo(phWndParent, pbBlockApp, pbEnableSaveSettings)`

Starts App Info

Parameter	Type	Description
<code>phWndParent</code>	Window Handle	(OPTIONAL) Parent window of the App Info window or <code>hWndNULL</code>
<code>pbBlockApp</code>	Boolean	TRUE -> starts tool while application waits for it to end. FALSE -> starts asynchronously
<code>pbEnableSaveSettings</code>	Boolean	TRUE will save settings to registry FALSE will only keep settings for the current session, no registry use

Return value:

The window handle of the App Info window. hWndNULL when TDApTools is not installed

App Info Export

Purpose of this is to generate App Info and save it to file or use it as string within the application for custom logging without the App Info GUI.

To start App Info Export from TDApTools:

The global function *PALTDAppTools_AppInfoExport* (from TDApTools.apl) will start the App Info Export functionality.

Boolean PALTDAppTools_AppInfoExport(psFile, pnAppInfoFlags, pnExportTo, pbOpenFile, rpsExportData)

Exports App Info to file or returns the App Info data as string

Parameter	Type	Description
psFile	String	(OPTIONAL) Filename to export to. When empty the data will be returned as string
pnAppInfoFlags	Number	Use any of the ncAPPINFO_ flags. Combine them by OR
pnExportTo	Number	One of the ncAPPINFO_EXPORT flags
pbOpenFile	Boolean	TRUE will open the file after export.
rpsExportData	Receive String	(OUTPUT) The export data

Return value :

Boolean = TRUE if the function was ok, FALSE otherwise

WMI Query

Purpose of this is to execute WMI queries and get the results as string and/or save it to file without the App Info GUI.

To start WMI queries from TDApTools:

The global function *PALTDAppTools_WMIQuery* (from TDApTools.apl) will execute the given WMI query and return the results.

Boolean PALTDAppTools_WMIQuery(psNamespace, psQuery, pbMOFOutput, psFile, pnExportTo, pbOpenFile, rpsQueryResult)

Executes a WMI query and returns the result and/or saves it to file

Parameter	Type	Description
psNamespace	String	WMI namespace for the query
psQuery	String	The WMI query
pbMOFOutput	Boolean	TRUE will use the MOF format for query results FALSE will use a table format
psFile	String	(OPTIONAL) Filename to export to. When empty the data will be returned as string
pnExportTo	Number	One of the ncAPPINFO_EXPORT flags
pbOpenFile	Boolean	TRUE will open the file after export
rpsQueryResult	Receive String	(OUTPUT) The result data

Return value:

Boolean = TRUE if the function was ok, FALSE otherwise

TDAppTools deployment check

You can check in your project if TDAppTools can be started at runtime.

This will check whether the dynalib is deployed. You can present an error message to the user when TDAppTools is not present on the system.

bool PALTDAppTools_IsInstalled()

Checks if TDAppTools.apd is deployed with the application.

Parameter	Type	Description
No parameters		

Return value:

TRUE when installed. FALSE when not.

TDAppTools version

To get the current version of TDAppTools.

String PALTDAppTools_GetVersion()

Returns the TDAppTools version

Parameter	Type	Description
No parameters		

Return value:

TDAppTools version as string.

Simple start screen or popup menu

Using just one line of code you can present all TDAppTools features in a simple start screen or popup menu.

Window Handle PALTDAppTools_Start(phWndParent, pbAsPopupMenu)

Shows the main TDAppTools GUI from where the several tools can be started.

To be used for devs to quickly present all TDAppTools features from one location.

Option to show a popupmenu instead of a starter window.

Parameter	Type	Description
phWndParent	Window Handle	(OPTIONAL) Parent window of the TDAppTools starter window or hWndNULL
pbAsPopupMenu	Boolean	TRUE then shows popupmenu at cursor position, FALSE will show a starter window.

Return value:

The window handle of the main window.

hWndNULL when TDAppTools is not installed or when using the popup menu option

Deployment

You have two options for deployment of TDApTools:

1. Deploy TDApTools (dll and apd) along with your application runtime (setup) from the start. TDApTools is then present at end user side and could be started the way you have implemented. (probably you have disabled the feature based on user authorizations or created an egg option).
2. Only deploy TDApTools (dll and apd) at end user side when you actually need the features. So by default, TDApTools is not present on the system. While running your application you can decide to use TDApTools features by copying the TDApTools runtime files to the application folder or in a folder defined in the PATH settings. This can be done while your application is already running. There is no need to restart the application. The TDApTools interface will automatically check if it is deployed when called.

In both cases, you need to deploy these files on the system where you want the features to be available:

(TD x86) -> **TDApTools.apd** and **TDApTools.dll**

(TD x64) -> **TDApTools_x64.apd** and **TDApTools_x64.dll**

Make sure these files can be found by your application by putting a *PATH* specification to them or by deploying it in the application runtime folder. Mostly in the same folder as the application executable.

Support, remarks and suggestions

TD comes in several versions (ANSI, UNICODE, x86 and x64) and each version and sub version (PTF/SP) could behave differently when using TDApTools features.

Also when running on different Windows OS (Win7, Win10, 32 vs 64 bit) there could be differences.

As it is difficult to ensure that TDApTools works flawlessly on all those combinations, there is no guarantee that all features work correctly. But the effort is to make it as stable as possible.

When you encounter issues, please contact me and supply the needed information like

- TD version (SP, PTF level) and bitness (x86 and x64)
- Windows version and bitness (32 vs 64)
- TDApTools feature which fails or shows issues.
- GUI object which is inspected and fails, SQL statement having issues etc
- Any workflow steps to reproduce

Any suggestions for extra features?

Please contact me at:

daverabelink@xs4all.nl

If you would like to support this development, please consider making a donation.

The DONATE button can be found on TDWiki, at the left side in the toolbar:

https://wiki.tdcommunity.net/index.php/Main_Page

Or use this direct donate link:

<https://www.paypal.me/DaveRabelink>

Version history

Current version: 3.1

Version 3.1 09-07-2020 Dave Rabelink

General	
New	TD x64 support (TD 7.0 and higher)
Fixed	TDApTools.dll: Fixes crashes on NULL buffers in some API calls
Fixed	Better handling loading TDApTools.dll and compatibility checks
Fixed	QOL improvements on GUIs and TD specific glitches on several TDApTools features
Gui Inspector	
New	Added TD 7.3.1 messages SAM_NotifyToast, SAM_InitTrackMenu for MessageMonitor
Changed	"Send Message to" GUI improvements
Fixed	Blocking apps/windows will not hang TDApTools anymore
Sql Monitor	
New	Support for SQLite (sqlite3.dll): sqlite3_libversion, sqlite3_open_v2, sqlite3_open16, sqlite3_open, sqlite3_close, sqlite3_prepare16_v2, sqlite3_prepare_v2, sqlite3_bind_text16, sqlite3_bind_text, sqlite3_bind_int, sqlite3_step, sqlite3_column_int, sqlite3_column_text16, sqlite3_column_text, sqlite3_column_blob, sqlite3_finalize, sqlite3_reset
New	Added support for SqlClearLongBindDatatype

Version 3.0 12-02-2020 Dave Rabelink

General	
Changed	TDApTools.dll: new version to support App Info features
App Info	
New	Initial release having: general info, loaded dll's, PATH and environment settings, opened handles, running threads, SAL system variables and a WMI query tool
TD Profiling controller	
Changed	Block application option added when starting profiling tool
Gui Inspector	
Fixed	After changing windows styles of objects, they are now repainted correctly

Version 2.4 30-12-2019 Dave Rabelink

General	
New	TD 7.3 version support
Changed	TDApTools.dll is now TD version independent (build using VS2017)
Sql Monitor	
New	Support for SqlConnectEx, SqlImmediate, SqlClearImmediate, SqlImmediateContext, SqlGetResultSetCount, SqlGetRollbackFlag, SqlGetModifiedRows, SqlOraPLSQLStringBindType, SqlStore, SqlRetrieve, SqlPLSQLCommand
New	Enable/disable single hooks to ignore certain traces or improve trace performance
Changed	Significant performance improvements on hooking/unhooking to TD runtime
Changed	Having SqlVarSetup bind values are now correctly evaluated
Changed	Improved performance while tracing
Changed	Edited SQL statements (break mode) will be used in tracing and bind evaluations
Changed	Enable/disable trace and clearing history is now displayed as info in trace
TD Profiling controller	
New	Annotate feature. Starting with TD 7.3

Version 2.3 28-10-2019 Dave Rabelink

General	
New	TD 7.2 version support
New	Added start screen or popup menu to open the several tools from
New	About screen. Button for TD community forum
Changed	Exports will use a unique filename using counters when file already present
	All tools saving GUI options is now enabled by default
Changed	About screen. URL changed to https and tdcommunity. Direct donate link
Changed	
TD Profiling controller	
New	Initial release. Only available for TD 7.2 and up
Gui Inspector	
New	Tool to send/post messages to objects. Pick messages from fixed list (SAM and WM) or use custom messages
New	Set objects TOPMOST/NOTOPMOST from the actions/z-order menu
New	Set state of window (maximized, normal, minimized) from new Window state menu
Changed	TD themes screen, added Office 2016 theme (TD72 and up)
Changed	All Gui Inspector windows remember their screen locations
Changed	Now uses windows fontpicker instead of the Sal one to correctly preselect fonts
Fixed	Added missing CDK attribute strings up until TD72
Fixed	Window tree view shows now all missing icon for known TD objects

Version 2.2 13-01-2017 Dave Rabelink

General	
New	About screen. Buttons linked to YouTube, TDWiki and TD Samples
Fixed	Check runtime dll function now reloads dll when not found using full path
Sql Monitor	
New	support for ODBSal API functions OdrPrepareProc, OdrExecuteProc, OdrGetNextResults and OdrPrepareNextResults
New	support for SqlPrepareSP, SqlGetNextSPResultset, SqlCloseAllSPResultSets and SqlDropStoredCmd
New	options screen, About button
Changed	API attached functions list is now sorted
Changed	number values are now shown using the actual decimals (eg 1,05 or 234,32345 or 2)
Gui Inspector	
New	options screen, About button
New	options screen, checkbox to enable/disable DWM feature for better bounding rectangle display
Changed	column values in attributes and properties window can be selected (read only) (for copy to clipboard)
Fixed	SAL attribute screen repopulate issue when mode Keep bounds was enabled
Fixed	Now using DWM instead of GetWindowRect to paint bound rectangle top level objects (when API is present)
Fixed	TD51-TD63 -> TabPage and NavBarPage template names are now correctly determined (TD70 and up is ok)

Version 2.1 17-10-2016 Dave Rabelink

Sql Monitor	
New	Break in source feature. Jump to location in source when action is executed
Changed	attaching to API is now faster and more stable
Gui Inspector	
New	Message Monitoring (experimental)

Version 2.0 23-09-2016 Dave Rabelink

Sql Monitor (new)		
New		Initial version of Sql Monitor to trace SQL actions
Gui Inspector		
New		TD Themes selector. Change TD theme on the fly
Changed		Item name (qualified display) will show > for owner and . for parent relationships
Changed		Text export will open registered application for txt files (it was always using notepad)
Changed		HTML export will escape newlines now

Version 1.0 25-06-2015 Dave Rabelink

Gui Inspector (new)		
New		Initial version introducing Gui Inspector