

# **TEAM DEVELOPER™**

New & Changed Features

Product Version 7.2.3

Team Developer™: New & Changed Features

### **Open Text Corporation**

275 Frank Tompa Drive, Waterloo, Ontario, Canada, N2L 0A1

Tel: +1-519-888-7111 Toll Free Canada/USA: 1-800-499-6544 International: +800-4996-5440

Fax: +1-519-888-0677

Support: <http://support.opentext.com>

For more information, visit <https://www.opentext.com>

Copyright © 2020 Open Text. All rights reserved. OpenText is a trademark or registered trademark of Open Text. The list of trademarks is not exhaustive of other trademarks, registered trademarks, product names, company names, brands and service names mentioned herein are property of Open Text or other respective owners.

One or more patents may cover this product. For more information, please visit <https://www.opentext.com/patents>.

#### Disclaimer

No Warranties and Limitation of Liability. Every effort has been made to ensure the accuracy of the features and techniques presented in this publication. However, Open Text Corporation and its affiliates accept no responsibility and offer no warranty whether expressed or implied, for the accuracy of this publication.

Warning: This software is protected by copyright law and international treaties. Unauthorized reproduction or distribution of this program, or any portion of it, may result in severe civil and criminal penalties, and will be prosecuted to the maximum extent possible under the law.

# Table of Contents

- TEAM DEVELOPER 7.2.3 ..... 4**
  - SAL API ..... 4
  - UI ..... 4
  - REPORTBUILDER ..... 4
- TEAM DEVELOPER 7.2.2 ..... 4**
  - IDE ..... 4
  - SAL API ..... 5
  - NATIVE ROUTER ..... 5
  - MISCELLANEOUS ..... 5
- TEAM DEVELOPER 7.2.1 ..... 5**
- TEAM DEVELOPER 7.2 ..... 6**
  - IDE ..... 6
  - SAL API ..... 6
  - RUNTIME PROFILING ..... 6
  - UI ..... 7
  - MISCELLANEOUS ..... 7

# Team Developer 7.2.3

## SAL API

- Added new table flag, TBL\_Flag\_DisableVisTblSort, to disable sorting of VisTbl properties (row colors, images) during SalTblSortRows. This improves performance by about 10 percent.
- Added new table flag, TBL\_Flag\_DisableScrollRange, that can be set to TRUE before a loop that calls SalTblInsertRow, and then reset to FALSE when the loop completes.
- New SAL function, SalTabEnableCtrlTab  
**bOk = SalTabEnableCtrlTab( hWndTab, bEnable )**  
Enable or disable Ctrl-Tab key on tab bar. By default, the tab bar handles the Ctrl-Tab key.
- New SAL function, SalSetButtonChecked  
**bOk = SalSetButtonChecked( hWnd, nCheck )**  
Sets whether a radio button, option button, or check box is checked. A radio button can only be checked. Other radio buttons in the same group are unchecked.  
**Parameters:**  
hWnd Window Handle. The handle (or name) of a radio button, option button, or check box.  
nCheck Number. 0 = Unchecked, 1 = Checked.  
**Return Value:**  
Returns whether checked/unchecked operation successful.
- New SAL function, SalYieldSetInterval  
**bOk = SalYieldSetInterval( nInterval )**  
Sets the number of statements to execute between yield handling. Default is 48.

## UI

- The grid will no longer scroll to the beginning when it is disabled.

## ReportBuilder

- We cannot find out the way to open this old style WMF image. Instead ReportBuilder now save such WMF image data as external file and shows a dialog to indicate its location. Customer can convert it to EMF image manually and embed it to ReportBuilder instead of original WMF image.

# Team Developer 7.2.2

## IDE

- When opening text outlines from TD 7.0, 7.1 or 7.2 with TD 7.3, the user will be

prompted for conversion unless the value of Settings/AutoConvert in the registry is non-zero.

## SAL API

- Added COL\_NoEditText flag (0x200) for SalTblSetColumnFlags to prevent paste into column when set.
- VisArrayInsertItem to accept max index of array as inserting position and act like as VisArrayAppend.
- Allow LBS\_ThemedCombo style to be ORed with with LBS\_VisExtensions in call to VisListSetStyle in VTM\_Create for VT list boxes:  
Call VisListSetStyle( hWndItem, LBS\_VisExtensions| LBS\_ThemedCombo, lParam)
- Number greater than 2147483647000000 should not be formatted by using default scientific notation if picture format is specified.
- Changes for TD 7.2.1 introduces backward compatibility issue. SqlSetLongBindDatatype followed by SqlPrepareAndExecute does not work anymore due to that change. So, now SqlPrepareAndExecute is modified to not clear long bind data type setting. But this requires to reset long bind data type setting before calling next SqlPrepareAndExecute which does not use Binary bind variable explicitly. To do this new Sal function, SqlClearLongBindDatatype, is added. Also modified SalCommit/SalRollback to clear long bind data type. Following Sal functions, SqlPrepare, SqlCommit and SqlRollback, now clear long bind data type setting.

## Native Router

- Currently TD hard coded which ODBC driver supports back end result set. Now customer can specify ODBC driver supporting back end result set by themselves by using brsdriver option in [odbcctr] section of sql.ini like as follows. If actual ODBC driver name starts with this name TD enable back end result set for it. Note that this might introduce some side effect (performance issue or so) TD cannot handle.

```
[odbcctr]
brsdriver=msodbcsql
```

## Miscellaneous

- Now deployment installer has "Web service components" feature in Select Features dialog.

# Team Developer 7.2.1

- VisArray functions now works with UDV arrays.
- Binding data type for long data will be cleared when executing SqlPrepare. Because of this change SqlPrepareAndExecute, executing SqlPrepare and SqlExecute internally, does work with SqlSetLongBindDatatype anymore. If application uses Long String variable as bind variable for binary data, application should split SqlPrepareAndExecute with SqlPrepare and SqlExecute and calls SqlSetLongBindDatatype between them.

# Team Developer 7.2

## IDE

- New Debugging tab added to application properties dialog. This tab allows users to specify debug command line arguments to be passed during debug sessions, as well as setting a specific working directory.
- New REST Import Wizard to easily create UDV class structures necessary to serialize/deserialize JSON data using SalJSON API. The wizard can fetch JSON from a URL, or a pasted JSON string. It will then generate an APL file containing the UDV class structure needed by the API.

## SAL API

- SalStrEncrypt/SalStrDecrypt added to easily encrypt/decrypt strings using AES encryption algorithm.
- SalTabWindowActivated – Returns whether the child windows on the specified tab page are already activated.
- SalTabAttachWindowAsChild – Assigns a child window created at runtime to the specified tab page as a child of the current tab container
- SalGridsRowExcludedByFilter – Tests if a row has been excluded due to a filter applied by the user
- SalGridBackColorSet/Get & SalGridTextColorSet/Get – New functions for setting colors of the grid for various grid states
- SalJSONDeserializeUDVEx/SalJSONSerializeUDVEx – These new functions have array parameters to allow users to provide custom request header values and query parameters.
- SalRegexMatch/SalRegexSearch/SalRegexReplace/SalRegexReplaceAll – New regular expression functions for searching, replacing text. Team Developer uses the ECMAScript regular expression grammar as described here: <https://docs.microsoft.com/en-us/cpp/standard-library/regular-expressions-cpp?view=vs-2017>
- SqlConnectEx – New connection function that accepts database, user and password fields rather than using global variables such as SqlUser etc. This is useful for connecting to databases from a background thread, where use of globals can cause issues.
- SalStrCompressEx/SalStrUncompressEx – Compression functions that accept a Binary receive parameter to store the compressed data.
- SalTblSetTableFlags supports several new flags:
  - TBL\_Flag\_VirtualScroll, which is only valid against grid windows. When using this flag in combination with SalTblPopulate, the grid will use virtual scrolling to improve populate performance. Note that certain grid features like grouping/sorting etc. will be disabled with this mode turned on.
  - TBL\_Flag\_CtxMenuSortAlpha, when true will sort the columns in the column chooser context menu in alphabetical order instead of by column position.
- SalBinaryToString/SalStringToBinary supports two new flags, ENC\_BIN (no code page conversion) and ENC\_BASE64 (base64 encoding)
- SalNavSetGroupTitle – updates a navigation bar group title

## Runtime Profiling

- Applications can be profiled at runtime by turning on profile logging. This feature allows you to view call graphs and timing information related to application functions and message execution. Logs can be analyzed using a new Profile Viewer application. Profiling can be enabled using:
  - Environment Variable
    - TD\_PROFILING\_LEVEL: 1 or 2 can be specified
    - TD\_PROFILING\_FILE: file name for the log
  - Command line arguments:
    - --profilingLevel [1|2] --profilingFile <ouput file>
  - Sal API:
    - SalProfileStart(filename, profileLevel)
    - SalProfileSuspend()
    - SalProfileContinue()
    - SalProfileStop()
    - SalProfileGetInfo()
- Level 1 profiling includes messages and internal functions. Level 2 includes Sal functions and externals.

## UI

- New Office 2016 theme. New constant THEME\_Office2016\_R2\_Metro can be used with SalThemeSet (Win32 only).
- Windows Notifications – Ability to create native Windows notifications using new API functions and events. Notifications can be created in the system tray, or a balloon notification can be created.
  - SalNotifyAddIcon/SalNotifyUpdateIcon/SalNotifyDeleteIcon/SalNotifyShowBalloon
  - SAM\_NotifyIcon – event raised when user clicks on notification
- MDI Window has a new property, MDI Tabs. This can be set to a Top, Bottom etc. to have child windows render as tabs like the IDE.

## Miscellaneous

- Preference setting under General tab has been added to force TD to use UTF-8 encoding instead of the default UTF-16. Some source control systems work better with UTF-8.
- Windows minidump: New options to generate minidump files if a crash occurs in a Win32 application. This can be turned on using --minidump command line option, or by setting an environment variable TD\_MINIDUMP to 1, ON or YES. Once enabled, a minidump file will be created in the same directory as the application executable in the format <exe-name>-yyyyMMdd-HHmms.dmp. These dump files can be used by OpenText support to find the root cause of an application crash.