

[Pre-installation information](#)[New features in Gupta Team Developer 3.1](#)[New debugging tools](#)[Tracing](#)[Event logging](#)[XML support for table windows](#)[Resizable dialog boxes and toolbars](#)[Report Builder enhancements](#)[Table window GUI enhancements](#)[OLE DB enhancements](#)[Web Application Manager on Linux](#)[SQLBase 8.5 integration](#)[Miscellaneous enhancements](#)[New reserved words in SQLWindows](#)[Known Issues](#)[Prior versions](#)

Pre-installation information

The Team Developer 3.1 CD comes with a Product Media Code that will be required for installation.

If you are upgrading from a Test Drive version to the full version, there are two methods. (1) Use the Help, About menu item and choose the Purchase option. More information is available by opening the online help table of contents and viewing "Gupta License Management" under "About Gupta Team Developer". (2) Uninstall the Test Drive version and install Team Developer from the full version CD.

About 200MB of disk space is required by the installer in the temporary directory on the system drive. An additional 60 to 220MB of disk space is required for the actual Team Developer installation on the target drive, depending on features selected. If you use the same drive for both purposes, add the requirements together. These numbers are based on NTFS file systems; requirements for FAT file systems are about 25% greater. The large .MSI file used in installation is retained by Windows for use in future uninstall requests.

If you are installing the Test Drive version of Team Developer, you need Administrator privileges on the target machine.

The installer is based on InstallShield version 8, which is in turn based on Microsoft MSI. This means that Team Developer can't be installed to a mapped drive. Refer to the Microsoft KnowledgeBase article: <http://support.microsoft.com/default.aspx?scid=kb:en-us:318705>

If you are using the Oracle router for version 3.0 or higher of Team Developer, your client workstations must have version 8.1.7 or higher of the Oracle client software.

If you are migrating from a 1.x or 2.x version of Team Developer, you must be aware of some important changes that took place in version 3.0. You can read the 3.0 release notes [here](#).

Doing a full-text search of the Gupta Books Online collection requires version 6.0 or higher of Acrobat Reader.

Sample applications are not installed automatically in the main Team Developer installation process. Instead, an item is added to the Team Developer program group. Choosing this item, "Team Developer 3.1 Samples Installer", allows you to choose some or all of the available samples. See [Known Issues](#) for an important caution on choosing which directory to use for the samples. All samples have been developed and tested on Windows XP SP1 and using Internet Information Services (IIS) version 5.1 for the Web Extensions samples.

What's new in Gupta Team Developer 3.1

Team Developer 3.1 delivers big improvements in both usability and interface features.

Performance of both COM servers and COM clients has improved.

Report Builder has several new features for more flexible, better-looking output.

Web Developer applications can now run against Apache web servers on Linux.

Improved debugging capabilities

A new breakpoint management dialog box shows all breakpoints that are currently flagged in the application. A checkbox permits you to enable/disable the breakpoints from within the dialog.

In addition to breakpoints on a line of code, as in previous versions of Team Developer, the breakpoint management dialog also permits you to enter data expressions. When the value of such an expression changes, regardless of the line number, execution breaks and a message box informs you of the change. You can "browse" for a data expression by clicking the browse button next to the text box for expressions. This will display a separate Advanced Breakpoints dialog with a tree-structured view of all variables in your application. Note that variables from dynalibs are not available in this view.

The breakpoint management dialog contains a "number of iterations" field that you can use to indicate that you want the breakpoint to be suppressed until that number of iterations has occurred, then break.

For a breakpoint anchored to a line of code, you can also specify a condition (a data expression) to determine whether or not the breakpoint will actually break.

The Preferences dialog previously allowed you to set a different outline text color for lines with breakpoints. Now there are two colors available, one for enabled breakpoints and one for disabled breakpoints.

When a breakpoint occurs, you can use the Step Over and Step Into operations that were available in previous versions of Team Developer. You can also use two new operations. Step Out executes until control passes to the outline level higher than the current breakpoint. Run to Cursor executes until it reaches the line of code that you have highlighted, then breaks again.

Tracing

With tracing, you can direct detailed diagnostic information to several different output locations:

- The Windows event log (for Windows 98 and ME, a data file is used since the event log doesn't exist for these versions.)
- A named file.
- The SQLWindows output window normally used for displaying compiler errors. Note that this option is only available when the application is running in Debug mode.
- Directly to the "stdout" window, so that trace output can be integrated with third-party debugging tools.

For more information, see Team Developer online help and books for functions SalStartTrace, SalEndTrace, and SalTrace.

Event logging

Event logging will help applications, especially COM servers written in SQLWindows, to operate more smoothly. In earlier versions of Team Developer, applications would pop up message boxes when an error condition (such as array-index-out-of-bounds) occurred. Now you can instead route such error messages to the Windows event log. When you choose to use the event log, you can also indicate whether or not you would like the application to attempt to continue when an error is encountered. This is the programmatic equivalent of clicking "Continue" in an error dialog where the button choices are "Continue" and "Halt", or clicking "Yes" where the button choices are "Yes" and "No", etc.

Windows 98 and Windows ME do not offer system event logging, so logging entries will instead go to a data file whose name and location are noted in the Team Developer section of the registry.

For more information, see Team Developer online help and books for function SalUseEventLog.

XML support for table windows

New functions in SQLWindows permit you to write out the full or filtered contents of a table to an XML document and schema, and to read back an XML document and schema into a table window. See Team Developer online help under the index entry "XML" for a list of functions, constants, and messages. Also see the XML Support section in Chapter 15, *Table Windows*, in the book *Developing with SQLWindows*.

Resizable dialog boxes and toolbars

There are three new properties available for Dialog Box objects:

- Resizable: default value Yes.
- Horizontal scroll bar: default value Yes, enabled when dialog is resizable or dockable.
- Vertical scroll bar: default value Yes, enabled when dialog is resizable or dockable.

There is one new property available for toolbar objects: Resizable, with a default value of No. It's enabled when the toolbar is dockable. Toolbars are "stateful", remembering their sizes when docked and undocked.

New messages, SAM_DockResize and SAM_DockResizeNotify, give you programmatic control over a resizing event that is requested by the user. These messages are sent to dockable items: dialog boxes and toolbars. Some SAL functions that affect window layout must be used with caution from within the context of these messages.

Full details about these new features can be found in Team Developer online help or the *Developing with SQLWindows* book.

Report Builder enhancements

Determining the total number of pages in a report is now easy. A new function, TotalPages(), can be used in formulas and fields.

Date and time strings are now formatted in compliance with the Windows International Settings of the computer that is executing the report.

New buttons in the toolbar allow quick access to formatting for text color, background color, border color, and border line style.

Text wrapping is now supported in page headers and page footers.

Boxes surrounding fields now resize themselves as their associated text word-wraps and increases the height of the line.

Conditional expressions can be used in page breaks. The Behavior tab for line properties has a "PageBreakBefore" option, and once that is selected a condition, such as a formula, can be specified.

Conditional expressions can be used to format a report line. The format choices that can be associated with a condition are background color and border color. The conditions are specified in the Line Properties dialog.

Conditional expressions can be used to format a report field. The format choices that can be associated with a condition are background color, border color, text color, font, and font style. The conditions are specified in the Field Properties dialog.

You can output to multiple paper trays for the selected printer. The Report Format dialog has new combo boxes to select a tray for First Page and Other Pages. The Print dialog has an option to override these tray settings temporarily at the time of printing.

It is possible to print reports with page size set to custom if the printer connected supports custom printing for the particular page size.

The behavior of File Open has improved. The file type and directory used in the last File Open persist as the defaults for the next File Open.

Table window GUI enhancements

Dozens of new functions enable you to manipulate the appearance of entire tables, specific rows, specific columns, column headers, and specific cells. You can manipulate text font, foreground colors, background colors, images, row header bitmaps, and separator lines.

To use these new functions, include GTABLEX.APL in your SQLWindows application. To see a list of these new functions,

open GTABLEX.APL by itself and view the Internal Functions section, or check the Team Developer online help under index entry "table window extended GUI". Each function is described in detail in Team Developer online help and in the SQLWindows Function Reference, and also in source code comments in GTABLEX.APL. Supporting file GTABLEXI31.DLL is required to deploy applications that use these new features.

OLE DB enhancements

New system variable `SqlUDL` stores the name of a UDL file or an OLE DB data provider. When this variable has a value, function `SqlConnection` will use it to establish a connection. If any part of the connection string is still incomplete after processing `SqlUDL`, `SqlConnection` will retrieve the missing information from variables `SqlDatabase`, `SqlUser`, and `SqlPassword`.

Function `SqlCreateSession` looks for a ".UDL" suffix in its second parameter and, if found, presumes that the parameter value is a UDL file name, then fetches connection string information from that file. Again, any missing information is retrieved from variables `SqlDatabase`, `SqlUser`, and `SqlPassword`.

The advantage of this new behavior is the ease with which you can move existing applications from traditional router connections to OLE DB connections, with very few changes to your source code.

New message `SAM_SessionError` traps SQL errors that occur in OLE DB sessions rather than regular connections. Previously it was difficult to determine whether a SQL error was the result of a session or a regular connection. Now it will be easier to process session errors.

New function `SqlGetCommandText` retrieves the text of the last statement prepared for the connection that you are currently monitoring, whether or not an error occurred.

Web Application Manager on Linux

Linux modules have been created for some Web Application Manager components. This allows you to run WAM applications against Apache web servers running on Linux, greatly increasing the number of potential servers for your application. For details on how to install and configure WAM on Linux, see *Building Web Applications with Gupta* (webdev.pdf). In the TAR file supplied with WAM are two useful files. File `wam-install-help.htm` contains information on extracting and copying the files necessary for WAM on Linux. File `wam-install.sh` contains the actual scripts to copy files to target directories.

WAM on Linux is sensitive to the versions of the various components the interoperate. The documentation mentioned above contains tables of version combinations that have worked successfully with WAM.

SQLBase 8.5 integration

Team Developer 3.1 ships with SQLBase 8.5 Developer Edition. Developer Edition supports all the features of SQLBase with the following exceptions: Databases are limited to a maximum size of 1GB, `SQLWS32` and `SQLMPIPE` (TCP/IP and COM+) are the only available communication protocols, and there is no support for encrypted databases. Your license permits you to use SQLBase for developing applications, but not for deploying those applications in a production environment. For an overview of available licensing options please visit our website at http://www.guptaworldwide.com/products/SQLBase_howtobuy.aspx. To contact Gupta Technologies LLC or one of their partners, visit <http://www.guptaworldwide.com/company/contact.aspx>, or call (650) 596-3400 in the U.S.

One major feature of SQLBase 8.5 is multiple installations. This includes the ability to use any name and location for the configuration file (formerly always named `SQL.INI`). You can install Developer Edition alongside other SQLBase servers and it will work simultaneously with the other servers without interference. However, this smooth operation depends on your client applications referencing the correct copy of the configuration file, thus connecting to the correct database server. To make this happen, SQLWindows has some new capabilities.

- The Preferences dialog, General tab, has a new data field allowing you to choose the name and location of the configuration file for use at design time. This setting will also apply to tools launched from SQLWindows, such as Team Object Manager, Report Builder, and Database Explorer.
- SQLWindows now has a new system variable named `SqlINI`. This variable should contain the path and name of the configuration file to be used by client applications at runtime. If the variable is null, the client application will search for filename `SQL.INI` in the application directory. If only a file name is provided in `SqlINI`, that file will be searched for in the application directory. You cannot change the value of `SqlINI` in "midstream" - if it is changed while database cursors are connected, the next function call referencing those cursors will return a SQL error. You must close all connections before changing `SqlINI`.
- A new parameter, `DBP_SQLCONFIGFILENAME`, has been made available to the functions `SqlGetParameter` and `SqlGetSessionParameter`. This parameter will retrieve the name and location of the configuration file being used by the client application.

In addition to the SQLWindows changes noted above, supporting tools have also been changed to conform with multiple installations. Team Object Manager, SQLTalk, and Report Builder can all be run from the command line, and these executables now accept an additional command-line argument that specifies the unqualified name or fully qualified name of the configuration file to be used upon startup. See the specific books for these tools for details of the argument syntax.

Miscellaneous enhancements

OLE DB connections have been simplified through new system variable SqlUDL. This variable is designed to hold the name of a UDL file, which contains appropriate keywords for an OLE DB connection string. Function SqlConnection has been modified to examine SqlUDL and, if it finds a filename, to use that UDL file to make an OLE DB connection. If the variable is null, SqlConnection uses the older logic of connecting using the values of SqlDatabase, SqlUser, and SqlPassword. Function SqlCreateSession also first examines SqlUDL and uses its information in preference to the second parameter of the function, which would normally contain a connection string.

By making SqlConnection pay attention to SqlUDL, Gupta has made it easy for you to migrate your application to OLE DB data providers without having to change all your SqlConnection calls to SqlCreateSession.

Internal improvements in the efficiency of COM+ method calls mean that Team Developer applications will now run faster when using COM+ components. New function SalComCleanupMode lets you choose between early cleanup (frugal with resources) and late cleanup (faster performance) in COM servers built with Team Developer.

The Oracle router now supports OCI 8 APIs, allowing LOB object datatypes (BLOB and CLOB) for reading and writing from Team Developer applications. New SQL.INI keyword USELOB and parameter DBP_ORAUSELOB help control this capability. Note that for a given Oracle error event, the error code returned by OCI 7 may be different from that returned by OCI 8.

Previously undocumented SQLWindows functions SalGetWindowLabel, SalPause, and SqlGetCursor have now been documented.

Documentation has changed for SQLWindows functions SalNumberArcCos, SalNumberArcSin, SalNumberExponent, SalNumberLog, and SalNumberLogBase10. Previously the documentation stated that if an invalid argument was supplied (such as a number outside the range of -1 to 1 for SalNumberArcCos), the function would return zero. Instead the functions actually return NUMBER_Null, not zero.

The Edit, Replace menu item can now be invoked with accelerator key Ctrl+R

On top-level windows with "Accessories Enabled" property = Yes, the hardcoded control IDs for some of the parts of the window have changed.

| Component | Old ID | New ID |
|-------------------|--------|--------|
| Gupta:MDIClient | 0x7FF0 | 0xE900 |
| Gupta:Form | 0x7FF0 | 0xE900 |
| Gupta:StatusBar | 0x7FF1 | 0x7FF1 |
| Gupta:FormToolBar | 0x7FF2 | 0xE830 |

New reserved words in SQLWindows

The following section lists global names of functions, constants and variables related to new features in SQLWindows. Be sure that your existing applications do not reference these names. Names listed below have explicit help topics associated with those names in Team Developer online help, unless otherwise noted.

EVENT_ERROR | EVENT_INFORMATION | EVENT_WARNING

TRACE_Event | TRACE_File | TRACE_Output

SalStartTrace | SalEndTrace | SalTrace

SalComCleanupMode | SalUseEventLog | SalGetWindowLabel | SalPause | SqlGetCommandText | SqlGetCursor

SalTbIWriteXMLandSchema | SalTbIWriteXMLandSchemaEx

| | |
|--------------------|-------------|
| VisTblIsSubClassed | VisTblPaint |
|--------------------|-------------|

| | | | | |
|---------------------|----------------------|-------------------|----------------------|--------------------|
| VisTblSetRowColorEx | VisTblSetRowHdrColor | VisTblSetColColor | VisTblSetColHdrColor | VisTblSetCellColor |
|---------------------|----------------------|-------------------|----------------------|--------------------|

| | | |
|----------------------|----------------------|--------------------|
| VisTblGetRowColor | VisTblGetRowHdrColor | VisTblGetColColor |
| VisTblGetColHdrColor | VisTblGetCellColor | VisTblSetColorEdit |

| | | |
|--------------------|---------------------|---------------------|
| VisTblSetTableFont | VisTblSetRowFont | VisTblSetRowHdrFont |
| VisTblSetColFont | VisTblSetColHdrFont | VisTblSetCellFont |

| | | |
|--------------------|---------------------|---------------------|
| VisTblGetTableFont | VisTblGetRowFont | VisTblGetRowHdrFont |
| VisTblGetColFont | VisTblGetColHdrFont | VisTblGetCellFont |

| | | | |
|---------------------|---------------------|--------------------|-----------------------|
| VisTblAddRowBitmap | VisTblAddBitmap | VisTblRemoveBitmap | VisTblRemoveRowBitmap |
| VisTblSetCellBitmap | VisTblGetCellBitmap | VisTblSetColBitmap | VisTblGetColBitmap |

| | |
|------------------|------------------|
| VisTblLoadBitmap | VisTblFreeBitmap |
|------------------|------------------|

| | |
|--------------|------------|
| COL_NOCHANGE | COL_NOT_BG |
|--------------|------------|

For the table below see help topic "Table window extended GUI row flags"

| | | |
|------------|-------------|---------------|
| ROWNEW_BMP | ROWEDIT_BMP | ROWDELETE_BMP |
|------------|-------------|---------------|

For the table below see help topic "Table window extended GUI line flags for rows"

| | | |
|--------------------|-----------------------|------------------------|
| TBL_FLG_NOROWLINES | TBL_FLG_SOLIDROWLINES | TBL_FLG_DOTTEDROWLINES |
|--------------------|-----------------------|------------------------|

| | | |
|--------------------|-----------------------|------------------------|
| TBL_FLG_NOCOLLINES | TBL_FLG_SOLIDCOLLINES | TBL_FLG_DOTTEDCOLLINES |
|--------------------|-----------------------|------------------------|

| | | | | | |
|----------------|--------------|---------------|-----------------|---------------|----------------|
| TBL_BMP_COLOR | TBL_BMP_MONO | TBL_BMP_SCALE | TBL_BMP_BESTFIT | TBL_BMP_TOFIT | |
| TBL_BMP_CENTER | TBL_BMP_LEFT | TBL_BMP_RIGHT | TBL_BMP_VCENTER | TBL_BMP_TOP | TBL_BMP_BOTTOM |

| | |
|--------|--------|
| SqllNI | SqIUDL |
|--------|--------|

Known issues

- The Evaluation Edition of SQLBase 8.5, shipped with Team Developer, displays a dialog box at startup that requires user intervention to continue. Since some Gupta applications are capable of starting SQLBase automatically, it's possible that such applications may fail to connect to the database server because the dialog has not yet been closed. For example, the Repository Setup Wizard for Team Object Manager can fail with an error message "Cannot open script file Tm_sb6.sql". However, the actual error is an inability to connect because the server startup process has been delayed while the dialog box waits for an answer.
- When you install Team Developer, and don't install SQLBase at the same time, it builds a Database Explorer configuration in which the default name of the SQLBase server is "Server1". If you are using a SQLBase server with a different name, you need to configure that name in the Database Explorer tool. In SQLWindows, open Database Explorer, expand the nodes in the left pane, right-click on the SQLBase node, and choose Add to List. Then enter configuration information for your SQLBase server.
- If you install only Report Builder, and do not install other Team Developer components such as SQLWindows, your Report Builder installation may have difficulty connecting to your databases. The Team Developer installation process builds a prototype SQL.INI which probably won't match your actual database server environment. Furthermore, the installation process creates a registry entry that points to that prototype SQL.INI. The registry key is: HKEY_CURRENT_USER\Software\Gupta\ReportBuilder 3.1\Settings and the specific value is INIFilePath. To fix this problem, either modify the registry value to point to a valid configuration file, or remove the registry value so that Report Builder can search for SQL.INI using the search algorithm from older versions of Team Developer. (You may also wish to remove the prototype SQL.INI file so that it doesn't interfere with the search for the valid configuration file.) For more information about the new SQL.INI capabilities in Team Developer, see [SQLBase 8.5 integration](#).

- SQLBase Management Console is not being registered automatically upon installation of SQLBase 8.5. You can register it manually by running the REGSVR32 command in Windows and referencing the SQLBMMC.DLL file.
- When running the "Team Developer 3.1 Samples Installer", you should choose the Team Developer program directory as the target directory for the installation. If you choose a different directory, an AXLIBS subdirectory will be built in that other target directory. But for the samples to work correctly, there should be only one AXLIBS, with all relevant APLs in it, under the Team Developer program directory. When using a different target directory, you will need to manually copy all the files in that AXLIBS subdirectory to the AXLIBS directory used by the main Team Developer installation. In addition, in order to run sampler.exe from the Team Developer program group, you must choose the Team Developer program directory as the target of the samples installer to get the correct directory structure expected by sampler.exe.
- Web Application Manager, when installed, overwrites the registry entries of any earlier version that is still installed. As a result, starting and stopping WAM from the Services window always starts and stops the most recently installed version, no matter which version was invoked.
- When configuring Web Application Manager as a service, be sure that the property "Allow service to interact with desktop" is enabled. If it is disabled, web applications won't work properly.
- The default protocol for SQLBase has changed from APIPE to WS32. At install time, SQL.INI is created with a default entry in the [WIN32CLIENT.WS32] section as follows:

```
Serverpath=server1, localhost/island
```

This means that at first you will only be able to connect to database ISLAND. If you change this line to use "*" in place of "island", you will be able to connect to all databases on "server1". However, doing this can cause all other router entries to be ignored, making the WS32 connection to Server1 the only one available.
- If you do use other routers and you need to work around this issue, you can make one of the following changes in the WIN32CLIENT.DLL section of SQL.INI:
 - If you don't need to use the WS32 protocol, comment out the line `comdll=sq_lws32`
 - If you do need to use WS32 and other routers, make sure that `comdll=sq_lws32` is not the first active router in the WIN32CLIENT.DLL section.
- The PATH environment variable is altered during installation to point to the Team Developer 3.1 program folder prior to looking for other Gupta program folders. Uninstallation on Windows 98 and Windows ME does not remove this path entry. For these versions, you may wish to manually edit the PATH statement after uninstallation.
- The SQLRouters for CA-Ingres and Informix have been removed from the Team Developer 3.1 installation process, and will be made available via download from our Web site in future.
- Report Builder crashes when a report file saved in Report Builder 3.1 is opened in earlier versions. This is as designed.
- Calling SalModalDialog during the processing of message SAM_DockChange against a dockable object causes the dockable object to also close.
- A GPF occurs using the Microsoft OLEDB provider when trying to insert into a table using a bind variable that is bound to a function. When the bind variable with the function is moved to the end of the insert list it works. This is a Microsoft limitation, and the limitation and workaround also apply to ODBC.
- Calling an Oracle Stored procedure passing a IN numeric parameter containing 40 digits after the comma (as the result of a division) will cause a GPF at execution.
- When using SalTbIPopulate with Oracle's OLE DB Provider, selecting the new Timestamp datatype results in a GPF. This is an Oracle provider problem.
- Using the Oracle OCI 8 API causes errors while retrieving number values with more than 20 digits.
- Using ARRAY datatypes with the Oracle OLE DB data provider and PLSQL procedures causes errors. Oracle does not support this datatype under these conditions.
- In Web Application Manager, starting a service (a Team Developer application registered with WAM) and immediately stopping that service can cause the following error: "Error encountered loading library: qweb31.dll".
- In Web Application Manager, stopping a service and immediately starting that service can cause the following error: "Web App Manager could not create a mapped file".
- Conditional breakpoints do not break when the expression used in them evaluates to NULL. Writing a conditional breakpoint expression that tests for NULL will not cause a break. An expression that tests for NOT NULL will work, however.
- The behavior of disabled breakpoints has changed. When a user clicks on "Disable All" during a debug session that disability will remain in effect even after the end of the current debug session (in earlier versions it would automatically reset back to "Enable All" when the user ended a debug session). Now the user is required to manually click "Enable All", if desired.
- Oracle date formatting in its OLE DB provider cannot be changed during a session by using the ALTER SESSION SET NLS_DATE_FORMAT command. The format is fixed at YYYY-MM-DD HH24:MI:SS. However, when using dates as parameters in queries, the date format is controlled by the Regional Settings in your Windows control panel.
- In the SQLWindows IDE, the right-click context menu for a File Include contains item "Explore ActiveX Library". This item is visible when the included file has a .APL extension, but not when the file has a .APT extension.
- SQLWindows user-defined variables and constants with names longer than 64 characters will cause error error "Not enough memory to save compiled code and data" during compilation, and the compile will fail. User-defined symbol names must not exceed 64 characters.

- A side effect of the 64-character limitation for symbol names is that ActiveX Explorer will abandon its naming convention if it generates a symbol name larger than 64 characters. For example, when working with COM objects from Microsoft Word, constants are usually named in the format <ENUM_NAME>_<VALUE_NAME>. But that can lead to an excessively long name like Word_wdPhoneticGuideAlignmentType_wdPhoneticGuideAlignmentZeroOneZero. In such a case, the ENUM portion of the name is dropped, and only the VALUE is used: wdPhoneticGuideAlignmentZeroOneZero.
- The XML support functions for table windows are aware of special characters (& < > ' "), and these characters are "escaped" properly when they are part of the data value of a column. However, if these characters are present in a table or column name, they will prevent the XML functions from working properly.
- If you use functions in the Developer Extension Kit (CDK) to load an outline from a SQLWindows version prior to 3.1, the outline will not be converted to the new version and you will get an error message, "CDK is unable to open an older outline version." You should convert outlines to 3.1 manually before opening them with the CDK.
- Microsoft OLEDB Provider for Oracle should not be used to insert binary data (such as a picture) stored in a Long String SQLWindows datatype into an Oracle column that is *not* a long datatype.
- When bind variables are used in SQL queries, the error messages that appear are different than those that appear when literals are used. For bind variables and DML queries, the error code presented is generic, and the details of the specific error that actually occurred are embedded in the description of the generic error. For bind variables and DDL queries, the specific error code and error description are presented.
- When you submit an invalid SQL statement to Oracle using the Oracle OLE DB provider, the error code returned is 364, in contrast to the code 900 expected according to Oracle documentation. When you use the Microsoft OLE DB provider for Oracle, the code returned is 900.
- When using the SaITrace* functions, if the destination is set to TRACE_File and the file name is specified without any path, the directory used for storing the file will differ in different versions of Windows. In Windows ME and 98, the trace file is created under the Team Developer program directory. In other versions of Windows the trace file is created under the WinNT\System32 directory.
- If Web Application Manager was installed as part of Team Developer, and you uninstall Team Developer, you must reboot your machine immediately after uninstallation.
- In some cases uninstallation of Team Developer will cause inadvertent deletion of COUNTRY.SQL. As a precaution you should manually backup this important file before uninstallation.

Prior versions

[Team Developer version 3.0](#)

Copyright © Gupta Technologies LLC. Gupta, the Gupta logo, Centura, and all Gupta products are licensed or registered trademarks of Gupta Technologies, LLC., All other products are trademarks or registered trademarks of their respective owners. Copyright © 2001 and 2002 Gupta Technologies