GUPTA

A Unify Company

Release Notes   GUPTA Team Developer™ 2005.1

Pre-installation Information

The Team Developer 2005.1 download comes with a Product Media Code that will be required for installation. The product is functional for 45 days from the date of installation.

If installing on Windows, you need Administrator privileges.

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.

A web browser is required in order to register the product.

If you are using the Oracle router, 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](#).

The outline version number of SQLWindows source files has changed. Applications that are saved in this version of Team Developer can't be opened by earlier versions.

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 2005.1 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 2005.1

XML Operations

SQLWindows applications can read, write, and manipulate XML documents using the Document Object Model. Details of these XML capabilities are available in a new chapter "Handling XML", in the book "Developing with SQLWindows" (dev.pdf). Similar information is available in online help, under the table-of-contents entry SQLWindows Programming / Handling XML.

UDV Serialization and Deserialization

Closely related to XML support is the ability to serialize the instance variables of a UDV (user-defined variable), or array, to an XML document, and to deserialize an XML document into a UDV or array. These abilities will be extremely useful, particularly to those developers who wish to save and reuse the "state" of an application.

To support serialization, there are three new choices available when you modify an instance variable in a class definition (either through Coding Assistant or by choosing "Add Next Level" in the right-click context menu):

- XML Serialize: allows you to indicate whether this specific instance variable should take part in serialization operations. Default is Yes.
- XML datatype: allows to to designate a general SQLWindows datatype such as "Number" as a more specific XML datatype such as "Currency". No default value.
- XML Nulls Allowed: allows you to indicate whether null values in the XML document are permitted when deserializing XML into a UDV. Default is Yes.

Find-and-Replace Enhancements

Find and Replace can now be accessed from the SQLWindows toolbar. A drop-down combo box holds recently used search strings, or you can type your own. There are buttons for Find, Find Next, Find Previous, and Help. Find, Find Again, and Replace are now also context menu choices at many places in the SQLWindows outline.

Report Builder Enhancements

- The Conditional Output feature that is available in many blocks of a report is now also available for the Page Header and Page Footer blocks.
- The Define Query dialog now shows the datatypes of columns that are available for use in the query.
- The Define Query dialog now allows you to edit the names of specific columns used in the Conditions tab (Column/Formula), the Sort tab (Sort by, Asc/Desc), and the Group Conditions tab (Column/Formula).
- The Define Query dialog now permits you to enter case-sensitive database table names when delimited by quotes. Because of this new feature, you must now take extra care when working with table names that contain one or more quotes as part of the name (some database vendors permit quote characters in the names of tables.) If you encounter such a quote character, you must "escape" it by appending a second quote character next to the one that is part of the table name. Otherwise the quote in the table name might be misinterpreted as the delimiter of a case-sensitive name, instead of part of the name itself.
- The Define Query dialog now permits you to control the sequence of columns in the result set by dragging columns to new postiions using the mouse.
- LOB datatypes (BLOB and CLOB) can be passed to an input item, and Report Builder will show a picture (if the BLOB is a known graphic type such as JPG) or a text box (if it is a CLOB containing text).
- The Format Block command now provides the ability to specify general formatting properties (font, color, etc.) for all fields within a report block using a single dialog. It is still possible to override those properties for an individual field.
- The "Between Lines" setting in the Borders page of Fields properties is now saved with the rest of the report design
- It is now possible to cut and paste report elements between multiple Report Builder windows within the same Report Builder instance.
- The Formula Editor dialog now gives you the choice of displaying all formulas, used formulas, or unused formulas. Thus you can delete all unused formulas by displaying a list of them, then deleting them.

Important note: The internal structure of Report Builder templates has changed. Templates that are saved in this version of Team Developer can't be opened by earlier versions.

Table Window Enhancements

New SAL function SalTblCreateColumnEx allows you to create new table window columns dynamically at runtime, specifying the data type of the column as well as other features such as title, display width, column position, and data length in characters.

Miscellaneous Enhancements

The outline version in SQLWindows has changed. Applications edited with this version cannot be opened by earlier versions.

For connections to Infomix databases, using function SqlGetParameter with parameter DBP_BRAND will now return value DBV_BRAND_INFORMIXONLINE rather than the more generic value 41.

The command line used for invoking an external application (Tools / User Tools menu in SQLWindows) now requires an additional parameter: -MTX. This parameter ensures that shared outline memory is properly managed. Team Developer applications developed using the Developer Extension Kit (CDK) will also require this additional parameter. This requirement applies to applications that are executables - if the application is a SQLWindows .APP file, the -MTX parameter is not required.

New Reserved Words in SQLWindows

There are a large number of new function and constant names related to XML processing. However, these are stored not in the core outline of SQLWindows, but in a supplied library, **xmllib.apl**, which you may optionally include in your application. Examine that file to see the new names. In addition to those names, the following names are reserved:

- SalTblCreateColumnEx
- SalXMLSerializeUDV
- SalXMLDeserializeUDV
- SalXMLGetLastError

Known Issues

- The new ability to serialize and deserialize UDVs means that every application that uses UDVs now produces slightly larger executables than they did in prior versions.
- It is possible to copy and paste formulas from one report to another, when multiple reports are open in Report Builder. However, there are circumstances which may cause the paste to fail. These include:
    1. The formula being copied contains an input item, and the report to which it is being copied does not contain an input item of the same name.
    2. The formula contains a field from a table which does not exist in the destination report's query.
    3. The source report consists of complicated JOIN queries, which don't correspond to the queries used in the destination report.
    4. A user-defined conditional formula with a customized name will not get copied to the destination report. Formula names are expected to be "formulaXX" like formula10, formula11, ... in the source report.
    5. Formula for conditional display of the input item should be selected from the formulas list in report objects. If the user selects the formula from the input list of the report objects this will not get copied to the destination report.
- The ability to resequence columns in the Define Query dialog of Report Builder can be performed only once during report design. Further attempts to alter the column sequence will have no effect.

- The new Find/Replace toolbar shows a slight misalignment of the combo box with the other icons under some circumstances, such as when the toolbar has been made floating, then closed, then reopened.
- When Team Developer is uninstalled, program shortcuts are not being deleted. You should delete them manually.
- In Report Builder, if you use the menu command Report, Format, Block on a Detail block containing a CrossTab within it, then click the Apply All button, the CrossTab is being removed from the report. If instead you click the "Apply" and "OK" buttons the changes are made successfully and the CrossTab is retained.
- When a SQLWindows application is using the OLE DB Data Provider against SQLBase 9.0.1, and the application uses a bind variable as a parameter in a SQL statement or a stored procedure call, the statement or procedure fails.
- Database Explorer occasionally fails when you switch between tabs, particularly the SQLTalk Script tab.
- In Team Object Manager, invoking the File/Preferences menu item causes the Diff/Merge tool to crash.
- The SQLWindows compiler does not catch a coding error that will cause SQLWindows runtime to crash. The error occurs when a function whose return datatype is a UDV is itself used as a parameter to a second function, when the datatype of that parameter is an array of UDVs.
- The behavior of Find/Replace in SQLWindows has changed slightly. In the previous version of SQLWindows, either of the radio buttons for "Scope" (Entire Outline or Selection) might be enabled depending upon the location of the cursor in the IDE. Now the Scope is always presumed to be Entire Outline. If you wish to use Selection instead, you must manually check that radio button.
- A few dialogs that represent property pages of the Graph Control in Report Builder do not have online help support. A default message appears indicating this.
- The Web Application Manager binary files for Linux do not work correctly with Apache Server version 2.0.50 (which is the default for Red Hat Enterprise Linux version 4.0.) The binary files work correctly with Apache 2.0.49.
- SQLWindows applications running against SQLServer2005 OLEDB Provider do not work correctly when invoking stored procedures without any parameters but with return value (indicating status).
- Trying to register the notify.dll found under CDK samples creates the registry key "DesignHookDLL" under HKEY_CURRENT_USER instead of creating it under HKEY_CURRENT_USER\Software\Gupta\SQLWindows 4.1. The workaround for this is to manually create the key under HKEY_CURRENT_USER\Software\Gupta\SQLWindows 4.1.
- If, when installing the sample applications, Team Developer 2005 is present on the client machine, the samples installer will present a default target directory that corresponds to Team Developer 2005, not Team Developer 2005.1.

Prior versions

Team Developer version 2005

[Team Developer version 3.1](#)

[Team Developer version 3.0](#)