

***Teradata SQL Assistant
Version 13.0 (.Net)
Enhancements and Differences***

by
Mike Dempsey

Overview

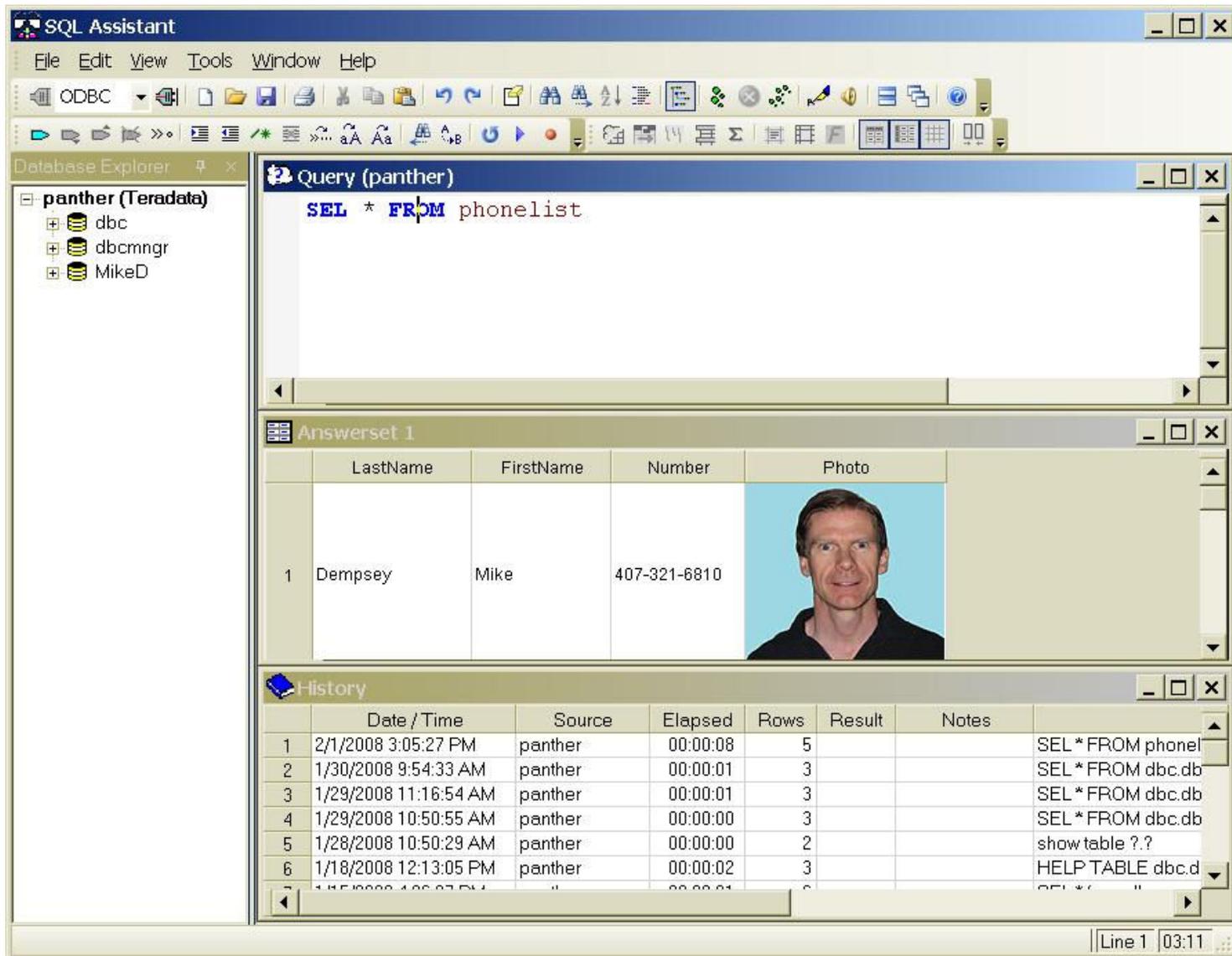
SQL Assistant 13.0 is an entirely new application that has been re-designed from the ground up.

It has been designed to look and act very much like previous versions of the product, by default, but it now uses the Microsoft .Net Framework rather than Visual Basic 6.

This presentation will cover the major changes in the application.

It is strongly recommended that you download the latest patch release before installing SQL Assistant 13.0

Default Look and Feel - Classic SQLA

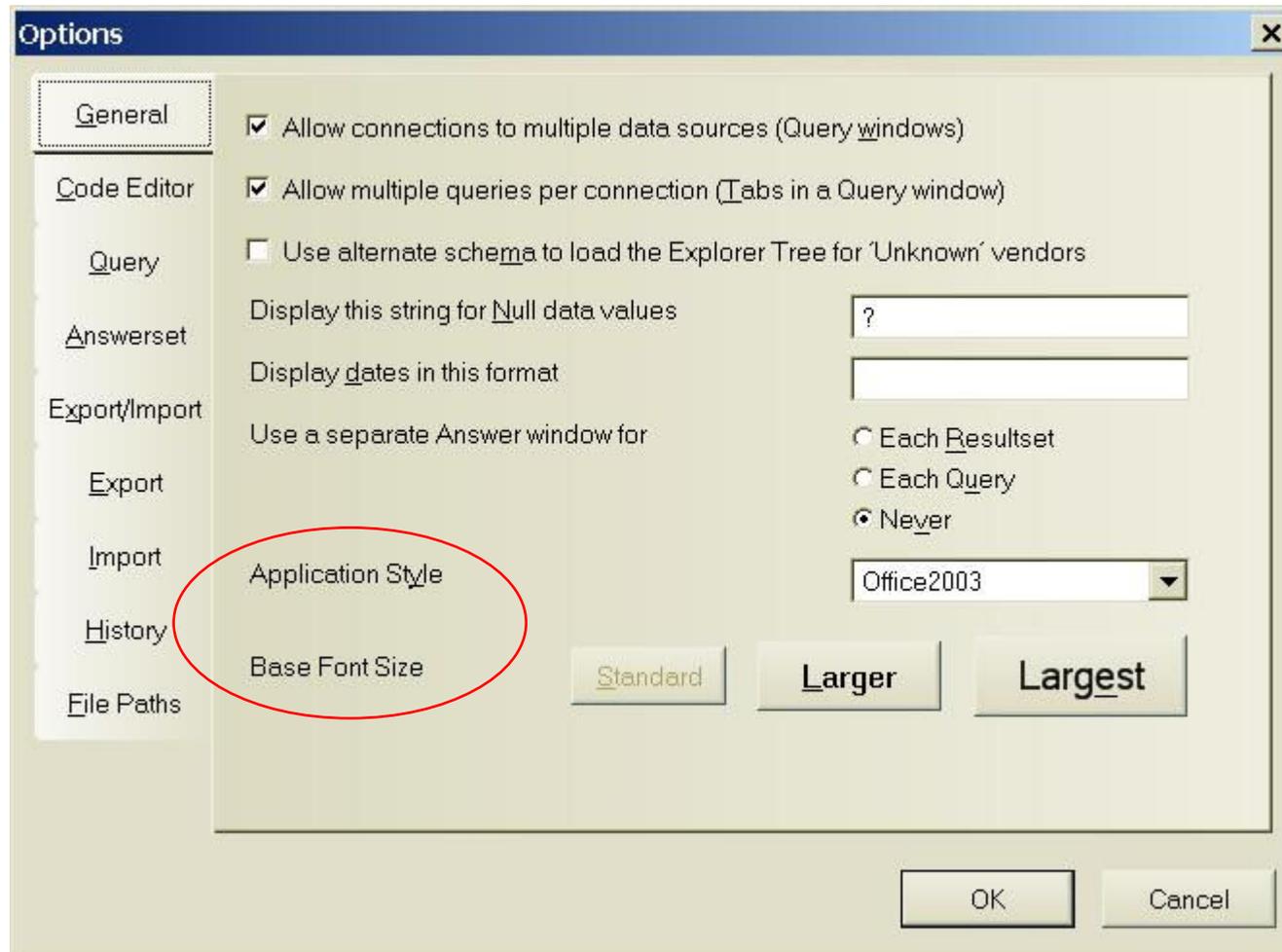


Different Appearances

The appearance of the product can now be changed in many ways:

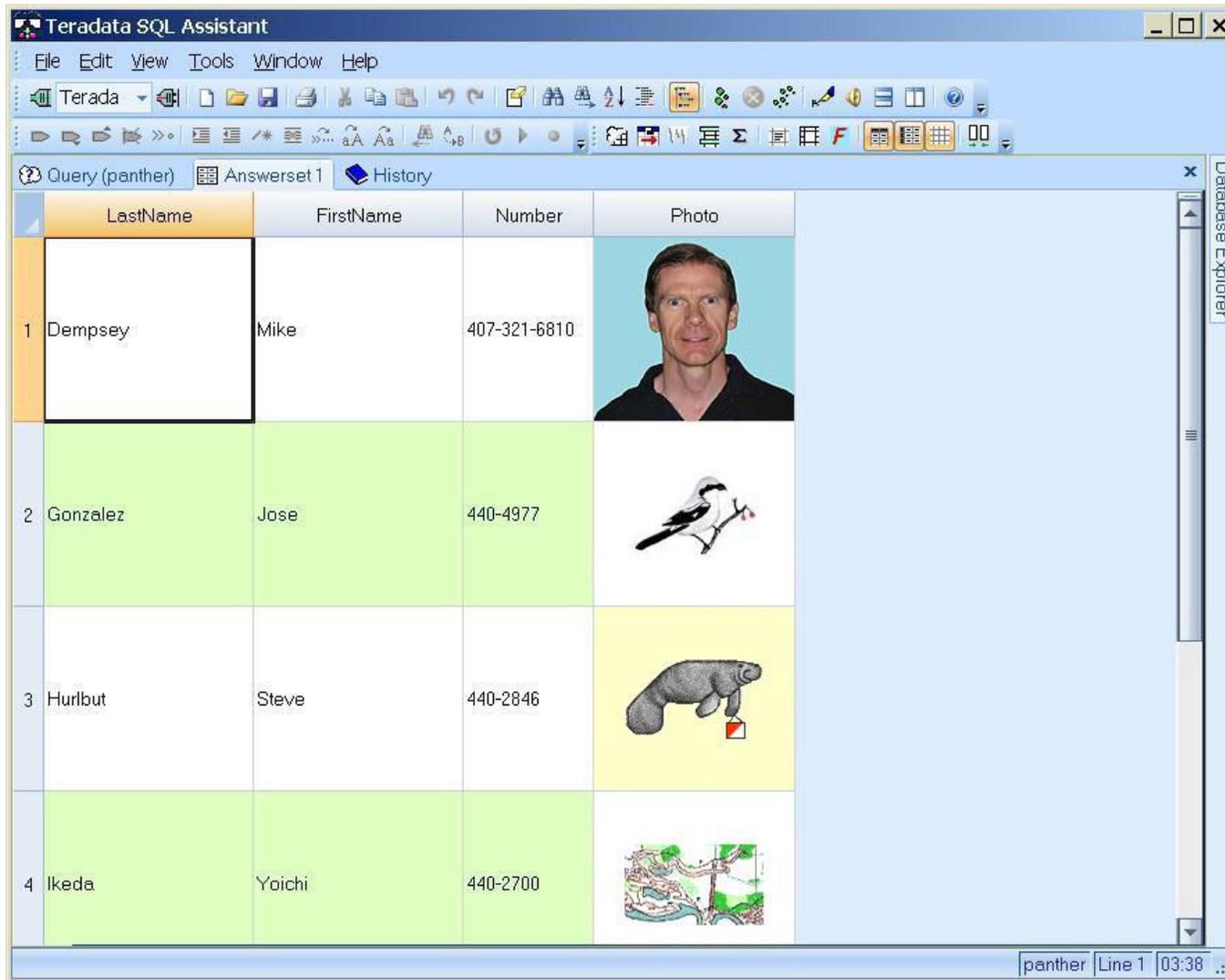
- **The Database Explorer Tree can be moved to the right hand side of the main window and ‘unpinned’ so that it slides into the right or left border when not in use.**
- **The usual Tiled/Cascaded/Maximized child windows can be replaced with the ‘Tabbed’ interface used in Visual Studio.**
- **The color scheme and ‘look’ of the controls can be changed to match various versions of Microsoft Office or Windows.**
- **The font size used for dialogs, status bar, etc. can be increased. This automatically increases the size of the dialogs also.**
- **Additional customization options are available for the menus and toolbars.**

Options Dialog



Use these options to change the overall appearance

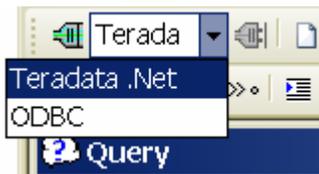
An Alternative UI



Using Office 2007 appearance, Tabbed MDI and unpinned Explorer Tree at the right

The Query Window

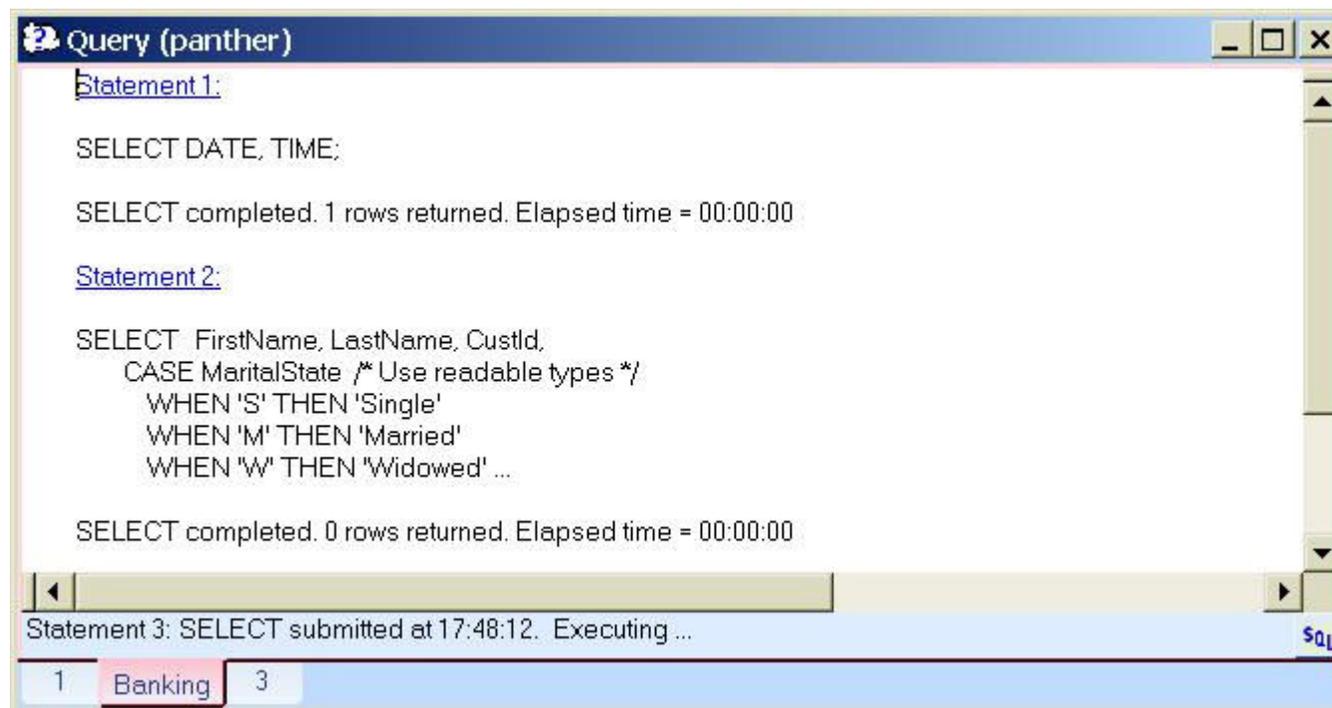
- You can now connect to multiple data sources at the same time. Each will open its own Query window and add its own root node into the Database explorer Tree.
- Use either ODBC or the Teradata .Net Data Provider with a Teradata database. Other vendor databases must use ODBC.



- You may now execute multiple queries at the same time. These will be executed in separate threads, but all data return will be performed in the main thread.
- The tab of a running query will change color to indicate that it is running, or that it has completed.
- The Query status bar now tells you the time at which the statement was submitted instead of displaying a time counter.

The Query Log

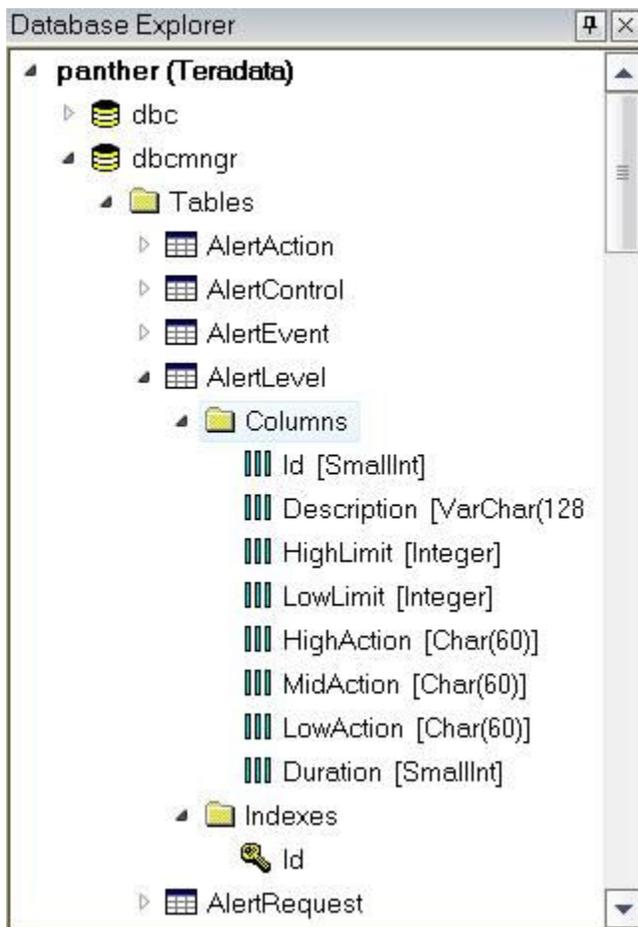
- While executing a Query that contains multiple statements it will display a log showing the status of each statement as they are executed. You can toggle the display between the log and the Query by clicking a button on the Query status bar.



The Database Explorer Tree

- **The root of the Explorer Tree is now the Data Source. This is required in order to allow connections to multiple data sources. Moving the cursor over this node will display the User Name and Default Database for this data source. (Note that the default database applies only to the primary session of this DS.)**
- **You can now select multiple objects in the Explorer tree and drag them all to the Query window at one time. (Note that all objects must be under the same parent.)**
- **Two new options allow you to specify whether objects should be qualified by their parent name, and whether object names should be enclosed in quotes.**
- **The quotes in which object names are enclosed are now vendor specific. For example, Microsoft Access uses [] not “ ”.**

The Database Explorer Tree



This example shows the Database Explorer Tree using the 'Vista' appearance

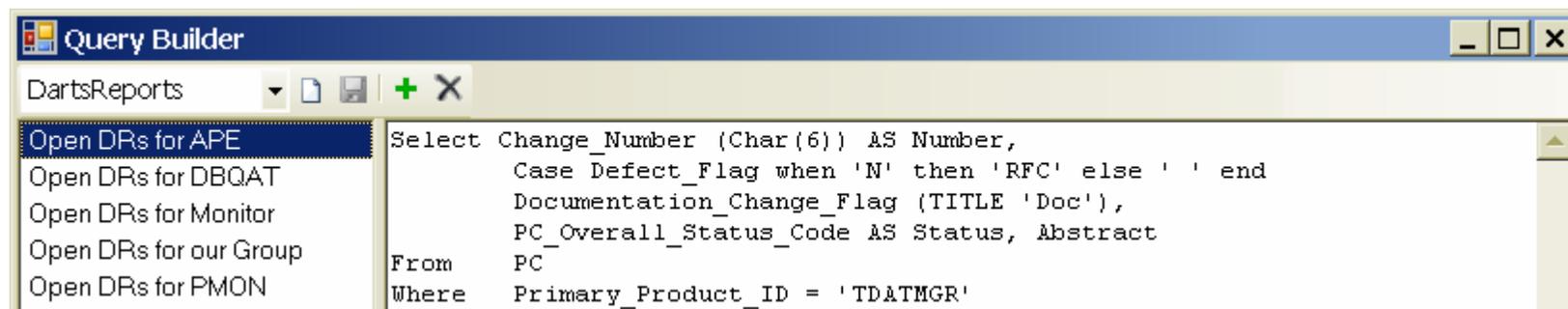
Note the root node showing the Data Source Name and Database Type

Hover over this node to see the User Name and Default DB

The Query Builder

- The provided SQL Syntax files can now be edited from within the Query Builder.
- You can create any number of additional files, from within the Query Builder, to contain your common Reports, or SQL Snippets
- These files can also be created by a DBA and distributed along with the product. The files will be named ***.qbs**. The location of these files will be:

[\Documents and Settings\\Application Data\Teradata\SQL Assistant](#)



The Answer set window

- Quick Sort by clicking the  icon in the Column Header
- Filter any column by a selected value by clicking the  icon

Note - These icons will only appear when the mouse moves over a column header.

- Use Outlook style row grouping

Click on the *Grouping* tool button  to display the Group Bar above the Answerset

Then drag a column header to the Group Bar to group by that column

Division ▲			
	Division	Department	Expenses
1	+ Division: 111 (3)		
5	- Division: 222 (3)		
6	222	ggg	700.00
7	222	bbb	200.00
8	222	bbb	500.00
9	+ Division: 333 (3)		
13	+ Division: 444 (3)		
17	+ Division: 555 (3)		

- Right click and select *Toggle SQL / Results* to see the SQL that generated the Answer set

The Answer set window

- Each Answer set can be split into 2 separately scrollable regions, both horizontally and vertically.
- *Wrap Text* is now hierarchical. If you wrap a single cell, then unwrap the entire column, that cell will remain wrapped.
- Save the answer set as a PDF or Microsoft Excel 2007 file
- Output from *SHOW*, *HELP* and *EXPLAIN* is now displayed in a rich text format in order to highlight keywords and phrases
- Decimal and BigInt values now supported up to 28 digits
- *Find* in Answer grid or History now defaults to searching the current column only. (Performance)
- *Page Setup* and *Print Preview* are now available for the Answer and History spreadsheets.

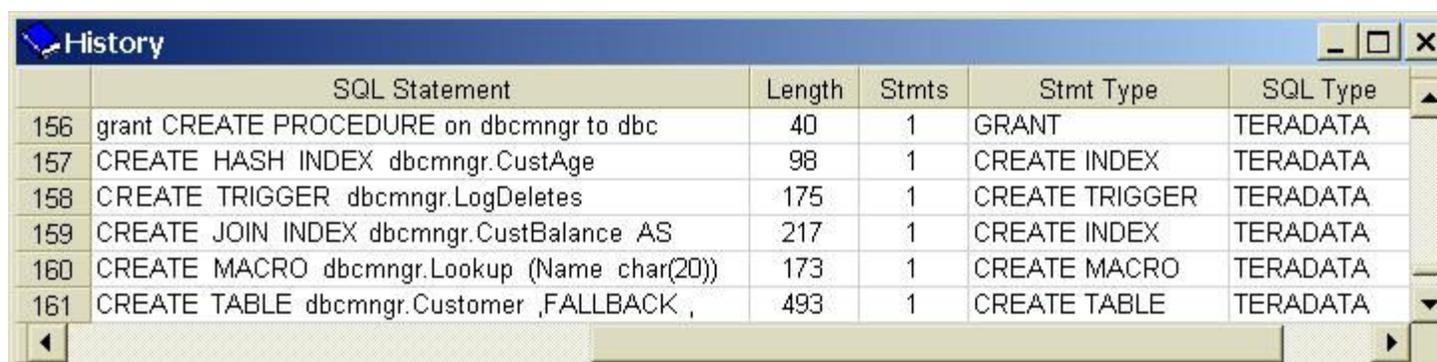
Page Setup

The image shows a 'Page Setup' dialog box with a blue title bar and a close button (X) in the top right corner. The dialog is divided into several sections:

- Margins:** A rounded rectangular box containing four spinners. The 'Top' spinner is set to 0.70", 'Bottom' to 0.30", 'Left' to 0.70", and 'Right' to 0.40".
- Default Orientation:** Two radio buttons: 'Landscape' (selected) and 'Portrait'.
- Paper Size:** Three radio buttons: 'Letter' (selected), 'Legal', and 'A4'.
- Print in Color:** A checkbox that is currently unchecked.
- Print a border around the data:** A checkbox that is currently unchecked.
- Page Title:** A text input field that is currently empty.
- Buttons:** 'OK' and 'Cancel' buttons at the bottom right.

The History window

- **Additional columns have been added to history for:**
 - The type of statements in the query (Select, Insert, etc)
 - The number of statements in the query
 - The length of the query



	SQL Statement	Length	Stmts	Stmt Type	SQL Type
156	grant CREATE PROCEDURE on dbcmangr to dbc	40	1	GRANT	TERADATA
157	CREATE HASH INDEX dbcmangr.CustAge	98	1	CREATE INDEX	TERADATA
158	CREATE TRIGGER dbcmangr.LogDeletes	175	1	CREATE TRIGGER	TERADATA
159	CREATE JOIN INDEX dbcmangr.CustBalance AS	217	1	CREATE INDEX	TERADATA
160	CREATE MACRO dbcmangr.Lookup (Name char(20))	173	1	CREATE MACRO	TERADATA
161	CREATE TABLE dbcmangr.Customer ,FALLBACK ,	493	1	CREATE TABLE	TERADATA

- **Elapsed times can now optionally display milliseconds**
- **The column order can be changed by dragging the column headers in the History window**
- **Option to Display the SQL as a single line (without carriage returns) or in standard multi-line format.**

Optional Tooltips display the full SQL when using multi-line mode

History Filter

History Filter

Date >

The previous 'n' days

Data Source LIKE

Use current Data Source

SQL Text LIKE

User Name =

Statement Type =

Statement Count >

Row Count >

Elapsed Time >

Show successful queries only

OK Cancel Clear

History rows can be filtered by the column content.

Multiple criteria can be 'And'ed together

Use > < or = operators

Specify actual date or a number of days

Other Changes

- The **'??'** parameter can now be used to import a hexadecimal string into a binary column.
Note that **'?B'** must now be used to import a BLOB.
- When you right click a tab the following command will apply to the tab you clicked.
(In previous versions it applied to the 'current' tab)
- Additional startup parameters have been added:
 - db <database_name>
 - cs <character_set_name>
- If an unexpected error occurs error info will be written to:
[\Documents and Settings\\Application Data\Teradata\SQL Assistant\SQLError.txt](#)
- All configuration settings and options are now written to files in the above directory. This allows a DBA to pre-configure the Options, Data Sources, Menus and Toolbars before distribution.

Product Installation

The following prerequisites are required:

- **Microsoft .Net Framework 2.0**
- **.Net Data Provider for Teradata 12.0 or Higher (13.0 recommended)**
- **An ODBC driver for any vendor database you plan to connect to**

No ODBC drivers are required before installation.

No ODBC driver is required for Teradata if you plan to use only the .Net Data provider.

Upgrade from earlier versions

The first time you execute SQL Assistant 13.0 it will look for a previous version and perform the following actions:

- Read your user preferences from the registry and create a **UserOptions.config** file
- Read all the existing Teradata ODBC Data Sources and write the information to **DataSources.config** for use by .Net connections (This includes all the databases that have been defined to be automatically added to your Database Explorer Tree)
- Read your History database (**queryman.mdb**) and create a new one with additional information added. (**SQLHistory.mdb**)

The history file will be created in the same directory as your previous History. The other files will be created in:

\\Documents and Settings\\<user>\\Application Data\\Teradata\\SQL Assistant

Upgrade, continued

Under some circumstances SQL Assistant may be unable to find your previous history database.

In this case it will open an empty History window. You will need to use the **File Paths** tab of the **Tools, Options** menu to specify the location of your old history database.

If you still have the previous version of SQL Assistant installed you can find this location by looking at the **Directory** tab of the **Tools, Options** menu.

Otherwise the default locations used to be:

\Documents and Settings\\My Documents\NCR\Queryman
or the directory in which the application was installed.

or you can use **Search** to look for the file **Queryman.mdb**

If you use the option “**Use separate history databases by data source**” then the file names will be **<DSName>.mdb** where **<DSName>** is the first [up to] 8 characters of your Data Source name.

Performance

Data transfer is usually a little faster when using the .Net provider as opposed to ODBC.

Large improvement in Import compared to SQLA 12.0

Import 1000 rows - 21 sec vs 83 sec

The time required to initially launch the application is much longer with SQLA 13. (It is even longer the first time after a reboot).

Launch app and load 200 history rows - 5 sec vs 1 sec

Memory requirements are considerably larger for SQLA 13.

Base memory with no connection or History window is 10MB vs 5MB

Answer sets and History require approximately double the memory

Suggestion – Those with large History databases should make use of the History Filter to reduce the number of rows that are loaded. Maybe load only the most recent 14 days of queries. When an older query is required apply a filter to find the likely candidates. (By SQL Text, Statement Type, #Statements, etc.)

Suggestion 2 – Unless you need to record multiple executions of the same query (eg. For performance testing) you can limit the size of the History database by using the ‘**Do not save duplicate queries in history**’ option.

Limitations & Restrictions

General Limitations

- You will not be able to save a multi-tab Answerset as a single Excel workbook.
- Exported UTF8 or UTF16 text files will always have a BOM prefix.

ODBC Specific Limitations

- Decimal values with more than 28 digits may be displayed incorrectly
- Retrieving Interval, Period, UDT or vendor specific type data, the functionality and appearance of the result set may be limited.
- CLOBs over 2K can only be returned using the UTF16 Session Character Set.

Teradata.Net Specific Limitations

- A WITH (summary) clause in a Select statement will cause the data return to fail.
 - Always uses ANSI date mode. All date literals must be entered as 'YYYY-MM-DD'
- If using a version older than 13.0:
- The data type displayed in the Database Tree for columns or parameters will be the .Net data type instead of the actual Teradata type.
 - Global Temporary tables will not be distinguished from regular tables
 - Index cardinalities will not be displayed after expanding an Index node.