



Embarcadero® Rapid SQL Developer™ 2.0 New Features Guide

Published: January 30, 2009

Copyright © 1994-2008 Embarcadero Technologies, Inc.

Embarcadero Technologies, Inc.
100 California Street, 12th Floor
San Francisco, CA 94111 U.S.A.
All rights reserved.

All brands and product names are trademarks or registered trademarks of their respective owners. This software/documentation contains proprietary information of Embarcadero Technologies, Inc.; it is provided under a license agreement containing restrictions on use and disclosure and is also protected by copyright law. Reverse engineering of the software is prohibited.

If this software/documentation is delivered to a U.S. Government Agency of the Department of Defense, then it is delivered with Restricted Rights and the following legend is applicable:

Restricted Rights Legend Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of DFARS 252.227-7013, Rights in Technical Data and Computer Software (October 1988).

If this software/documentation is delivered to a U.S. Government Agency not within the Department of Defense, then it is delivered with Restricted Rights, as defined in FAR 552.227-14, Rights in Data-General, including Alternate III (June 1987).

Information in this document is subject to change without notice. Revisions may be issued to advise of such changes and additions. Embarcadero Technologies, Inc. does not warrant that this documentation is error-free.

Contents

NEW FEATURES SUMMARY	4
Improvements to SQL Editor	4
Debugging Functionality	4
Enhancements to Data Source Explorer	4
SQL Templates	4
Data Source Indexing	4
Results Sets	4
Text File Encoding (Unicode Support).....	4
TOP NEW FEATURES	5
Improvements to SQL Editor	5
Debugging	9
Dependencies	11
Enhanced Data Source Explorer	13
Auto-Discovery and Import Features	14
SQL Templates	17
Data Source Indexing	19
Results Sets	21
Text File Encoding (Unicode Support).....	22
ADDITIONAL RESOURCES.....	23
Licensing Your Embarcadero Technologies Product	23
Embarcadero Technologies Product Support.....	23
Embarcadero Technologies Technical Support.....	23
Embarcadero Technologies on the Web	23

New Features Summary

Embarcadero Rapid SQL Developer simplifies development for application developers with many features for improving productivity and reducing code errors. A rich SQL IDE with code completion, real-time error checking, code formatting, and sophisticated object validation tools helps streamline coding tasks. The Rapid SQL Developer user interface improves overall productivity with database object look up functionality and side-by-side views of application code and SQL code. Rapid SQL Developer offers native support for IBM® DB2® for LUW, Oracle®, Microsoft® SQL Server and Sybase® as well as JDBC support for other DBMS. Rapid SQL Developer is available as a stand-alone application or as an Eclipse plug-in.

Improvements to SQL Editor

Improvements have been made to SQL Editor, including enhancements to Code Assist and Code Quality Check functionality. The look and feel of SQL Editor has changed as well to improve overall comprehension and process, and now features a bread crumb line in its window, which enables you to connect to data sources and execute code from within an individual file editor, rather than the old menu bar method of connecting and executing code.

Debugging Functionality

A fully-contained debug process has been introduced in Rapid SQL Developer 2.0, for the purpose of debugging stored routines in the development environment prior to code execution. The debug process is managed by the Debug view, and enables users to manage common debug functionality such as breakpoints and step over, as well as providing additional views and functionality to sort handle variables and compile dependencies.

Enhancements to Data Source Explorer

Rapid SQL Developer 2.0 provides enhancements to Data Source Explorer. This includes improvements to auto-discovery and import functionality, search, and filtering functionality.

SQL Templates

Rapid SQL Developer provides code templates that enable you to select a DML or DDL statement schematic and drop it into SQL Editor. Placeholder syntax can then be replaced manually with statement variables, or be otherwise customized to meet your development needs.

Data Source Indexing

The Data Source Cache has been renamed the Data Source Index to better reflect the nature of its functionality and to provide greater user comprehension with regards to its use and importance within the product.

Results Sets

1. The SQL execution process has been enhanced to provide more comprehensive result views, as well as strengthen output file options through the enablement of different file format options when saving those results.

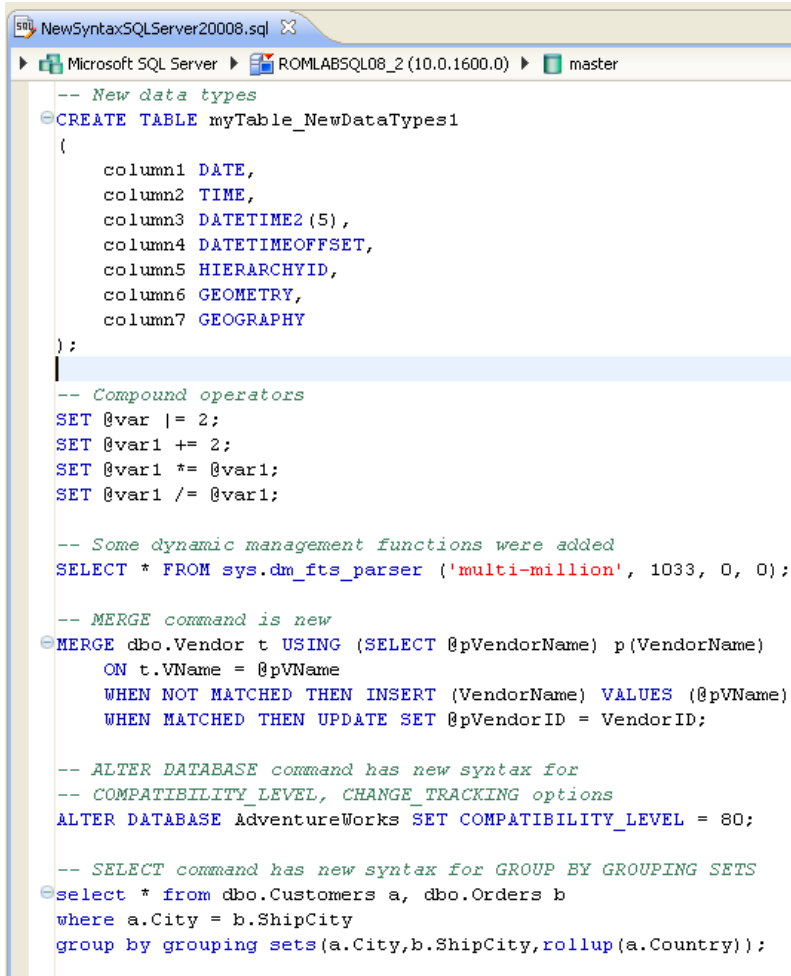
Text File Encoding (Unicode Support)

The Windows operating system uses CP1252 as its default encoding when you are editing and saving text files. This can be problematic for SQL code developers, especially when files need to be shared across multiple systems and platforms that use other encoding formats. Rapid SQL Developer 2.0 provides full Unicode support, which means you can set the system to encode files edited within the environment to save under a different encoding format. Additionally, you can specify or change different encoding formats for specific files, or groups of files, within the application.

Top New Features

Improvements to SQL Editor

In addition to overall improvements in terms of validation speed, responsiveness, and parsing, enhancements have been made to SQL Editor, including enhancements to Code Assist and Code Quality Check functionality. The look and feel of SQL Editor has changed as well to improve overall comprehension and process, and now features a bread crumb line in its window, which enables you to connect to data sources and execute code from within an individual file editor, rather than the old menu bar method of connecting and executing code.



```
Microsoft SQL Server > ROMLABSQL08_2 (10.0.1600.0) > master
-- New data types
CREATE TABLE myTable_NewDataTypes1
(
    column1 DATE,
    column2 TIME,
    column3 DATETIME2 (5),
    column4 DATETIMEOFFSET,
    column5 HIERARCHYID,
    column6 GEOMETRY,
    column7 GEOGRAPHY
);

-- Compound operators
SET @var |= 2;
SET @var1 += 2;
SET @var1 *= @var1;
SET @var1 /= @var1;

-- Some dynamic management functions were added
SELECT * FROM sys.dm_fts_parser ('multi-million', 1033, 0, 0);

-- MERGE command is new
MERGE dbo.Vendor t USING (SELECT @pVendorName) p(VendorName)
ON t.VName = @pVName
WHEN NOT MATCHED THEN INSERT (VendorName) VALUES (@pVName)
WHEN MATCHED THEN UPDATE SET @pVendorID = VendorID;

-- ALTER DATABASE command has new syntax for
-- COMPATIBILITY_LEVEL, CHANGE_TRACKING options
ALTER DATABASE AdventureWorks SET COMPATIBILITY_LEVEL = 80;

-- SELECT command has new syntax for GROUP BY GROUPING SETS
select * from dbo.Customers a, dbo.Orders b
where a.City = b.ShipCity
group by grouping sets(a.City,b.ShipCity,rollup(a.Country));
```

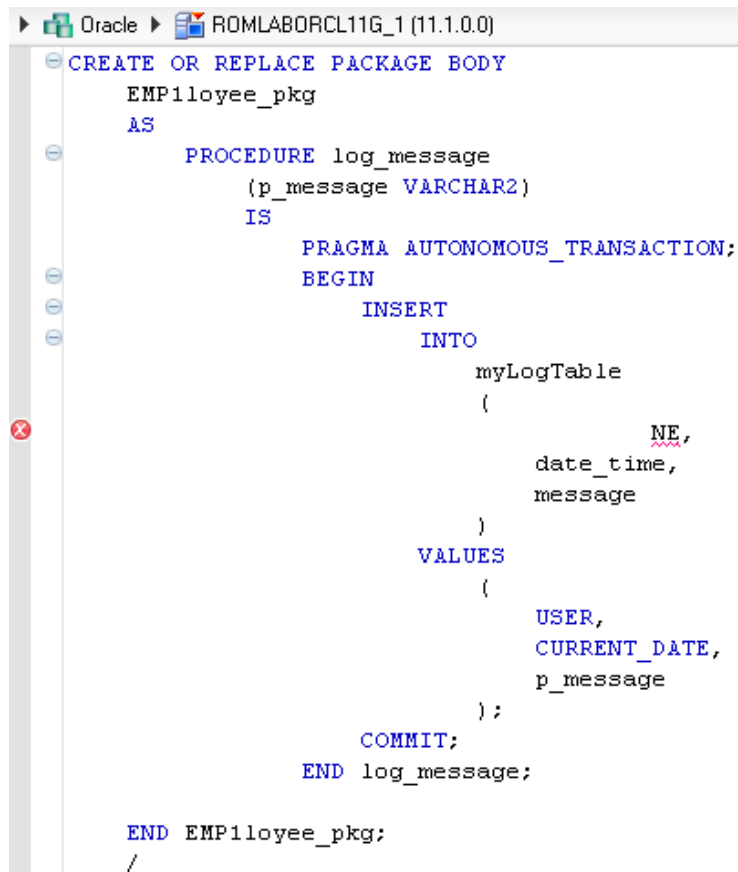
SQL Editor contains a number of important enhancements, including improvements to its code assist and code quality check features.

The following general improvements have been made to SQL Editor in version 2.0:

- **Code Assist Enhancements**
- **Code Quality Check Enhancements**
- **Look and Feel**

Code Assist Enhancements

The code assist feature has been enhanced to provide a more comprehensive code development experience, and to improve the process and speed by which it is written.



```
Oracle > ROMLABORCL11G_1 (11.1.0.0)
CREATE OR REPLACE PACKAGE BODY
EMPLOYEE_PKG
AS
    PROCEDURE log_message
        (p_message VARCHAR2)
    IS
        PRAGMA AUTONOMOUS_TRANSACTION;
        BEGIN
            INSERT
                INTO
                    myLogTable
                (
                    date_time,
                    message
                )
            VALUES
                (
                    USER,
                    CURRENT_DATE,
                    p_message
                );
            COMMIT;
        END log_message;
END EMPLOYEE_PKG;
/
```

Numerous enhancements have been made to the Code Assist feature in SQL Editor, including the ability to handle embedded code in DDL statements.

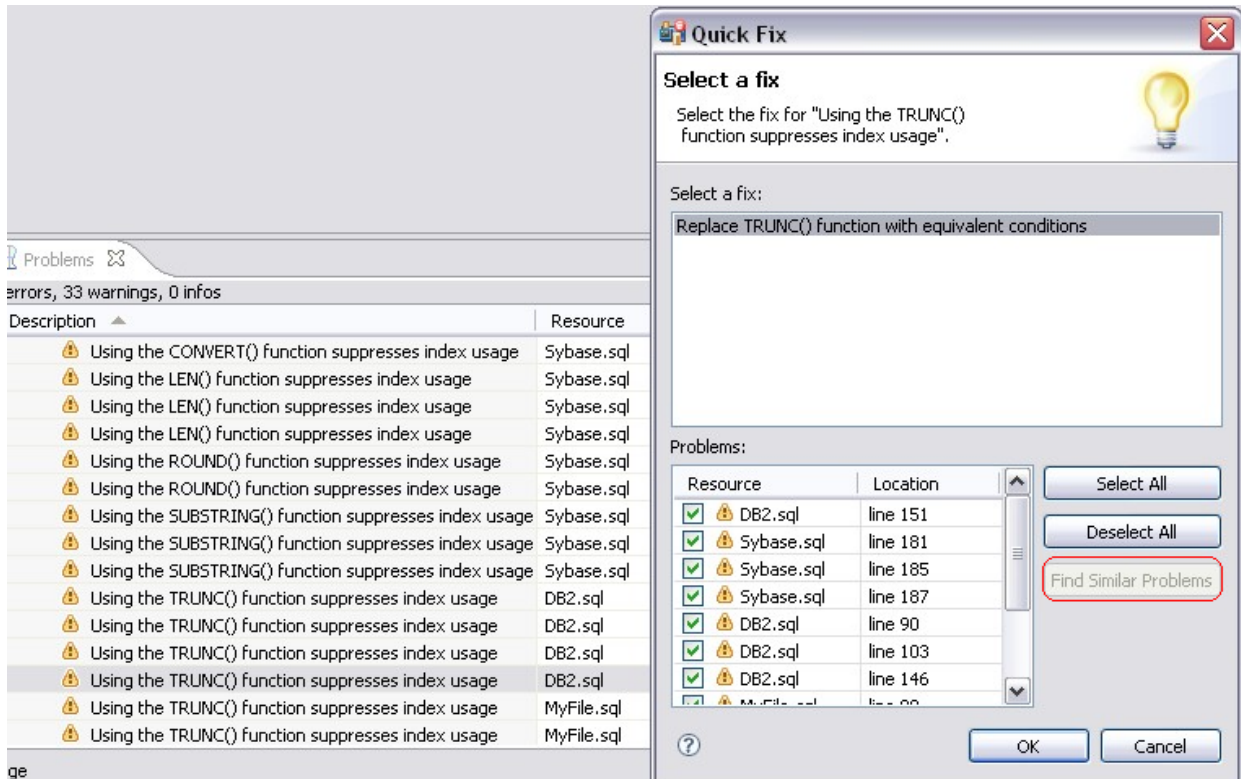
These improvements include:

- An **Auto-activated code assist** feature now brings up the code assist dialog when you pause while typing. This works in an identical manner to the **Ctrl + Spacebar** command, but is no longer a manual process. Each time you stop typing, the code assist dialog will appear and provide suggestions, as applicable. You can turn this feature on and off via the **Enable Auto Activation** command on the **Window > Preferences > SQL Development > Code Assist** panel, as well as specify the amount of pause time before the dialog appears.
- **Code assist filtering** enables you to filter out the database objects that are returned in a code assist dialog, and only display SQL templates. This makes code assistance more comprehensive, especially if you are looking for a specific statement rather than the name of an existing object. When the code assist dialog is open, press **Ctrl + Spacebar** to remove all database objects from the view and only display the appropriate SQL templates. You can add objects back to the view by pressing **Ctrl + Spacebar** a second time.
- Rapid SQL Developer 2.0 now supports code assist for **embedded DML in DDL statements**. This includes providing suggestions, as well as flagging missing or incorrect objects for DML embedded in CREATE PROCEDURE, FUNCTION, TRIGGER, TABLE, and VIEW statements.

Code Quality Check Enhancements

The code quality check feature is now supported across all platforms. This includes MS SQL Server 2008.

Additionally, through the **Problems** view, fixes for problems that the code quality check function discovers can be made for all files Rapid SQL Developer detects have similar issues. Quick fixes are performed via the **Quick Fix Wizard**, and can be applied to numerous files at once via the **Find Similar Problems** command.



The Quick Fix Wizard provides a means to fix the same code quality issue across multiple files in the development environment.

To apply a quick fix to an issue in the Problem view:

1. Right-click on a problem from the list and select **Quick Fix** from the menu. The **Quick Fix** wizard appears.
2. Select a fix from the list provided and click **OK**. Rapid SQL Developer attempts to resolve the issue automatically.

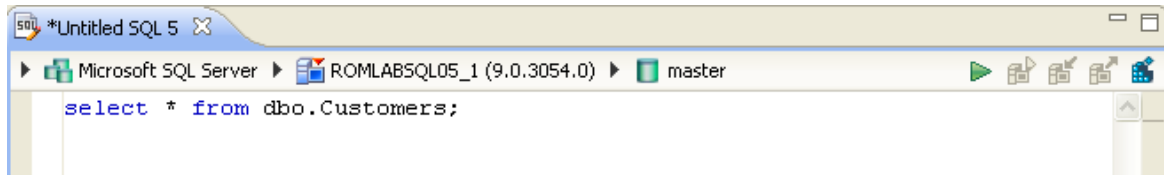
To find similar issues in the development environment:

1. In the **Quick Fix** wizard, click **Find Similar Problems**. The **Problems** list populates with all of the issues that are similar to your initial selection.
2. Use the check boxes beside the problems to select them, and then choose a fix and click **OK**. Rapid SQL Developer attempts to resolve all of the specified issues.

Look and Feel

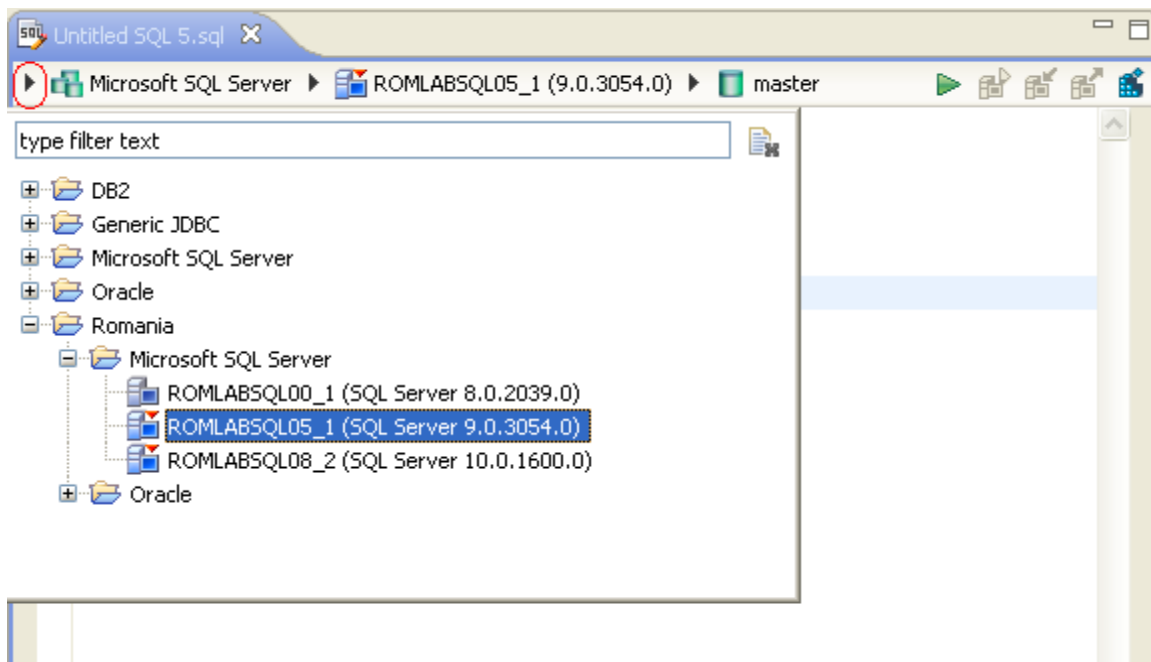
A bread crumb line has been added to SQL Editor that is used to display and specify a data source in relation the file. The menus contain lists of all registered data sources and catalogs (as applicable.) Additionally, the bread crumb line also contains execution commands.

These commands were previously located in the menu bar, but have been moved to better user comprehension overall, as well as provide an easier process when connecting to data sources and executing code.



The SQL Editor bread crumb line replaces the old location of data source association and execution menus/commands. This feature provides users with a more comprehensive interface and improves the code development process.

Additionally, the bread crumb line provides a menu that enables users to search and find registered data sources more quickly. The arrow icon located on the left-hand side of the bread crumb provides a drop down menu for the quick connection of data sources.



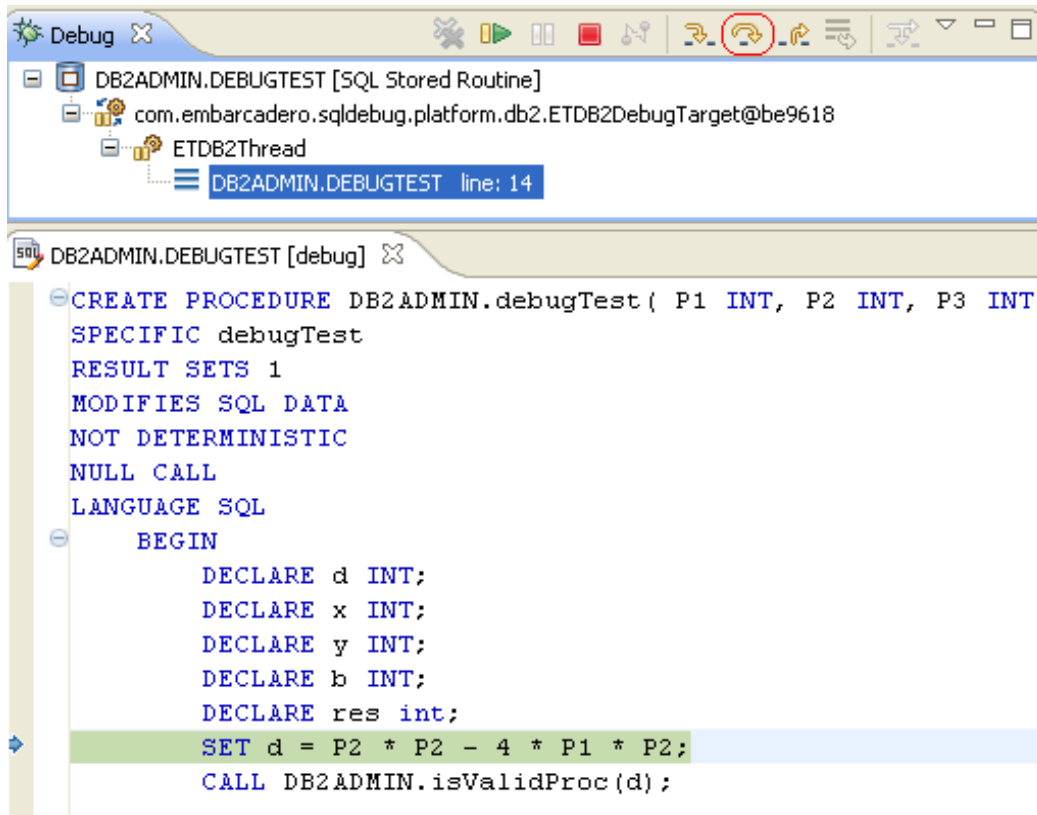
The arrow icon on the left-hand side of the bread crumb line provides quick access to registered data sources.

To change or associate a registered data source with a SQL file, or execute the file:

1. Click the database list and select the name of a registered database from the list that appears. You can filter the selection by typing the name or partial name of the data source you are searching for in the field above the tree.

Debugging

A fully-contained debug process has been introduced in Rapid SQL Developer 2.0, for the purpose of debugging stored routines in the development environment prior to code execution. The debug process is managed by the Debug view, and enables users to manage common debug functionality such as breakpoints and step over, as well as providing additional views and functionality to sort handle variables and compile dependencies.



The Debug view provides full capacity for debugging files prior to execution in Rapid SQL Developer.

You debug SQL Code from **Data Source Explorer**. Expand the data source tree until you find the file that you want to debug. Right-click on it, then select **Debug As** and specify how you want to debug it. The **Debug View** appears. You start the debug process via the debug commands in the **Run** menu.

The following additional commands are also available via the **Run** menu, and represent common commands needed for debugging and other operations:

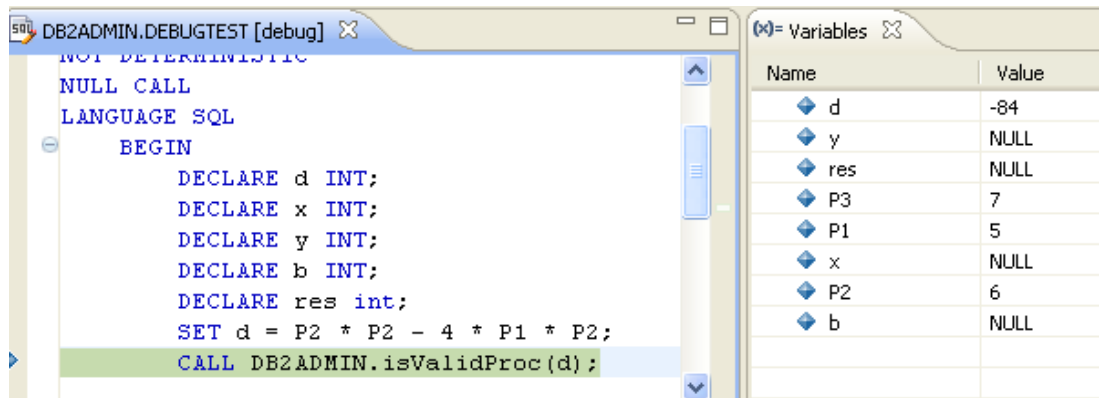
- **Breakpoint Management Commands** (Toggle Breakpoint, Toggle Line Breakpoint, Toggle Method Breakpoint, Toggle Watchpoint, Skip All Breakpoints, Remove All Breakpoints)
- **Step Over Commands** (Step Into, Step Over, Step Return)
- **Execution Commands** (Resume, Terminate, Run History, Run As, Run Dialog)
- **Other Commands** (Debug History, Debug As, Open Debug Dialog)

Furthermore, the debugging process also has full capacities to perform cross-language debugging, as it pertains to Java-to-SQL code. Using this feature, you can debug Java applications that contain stored SQL procedures embedded within the code.

In addition to common commands, the debugging feature provides a number of supportive views to handle things such as variables, breakpoints, and dependencies in stored procedures:

Variables

The **Variables** view is a complementary view to Debug view, and displays the values of a specified variable during the debug process.



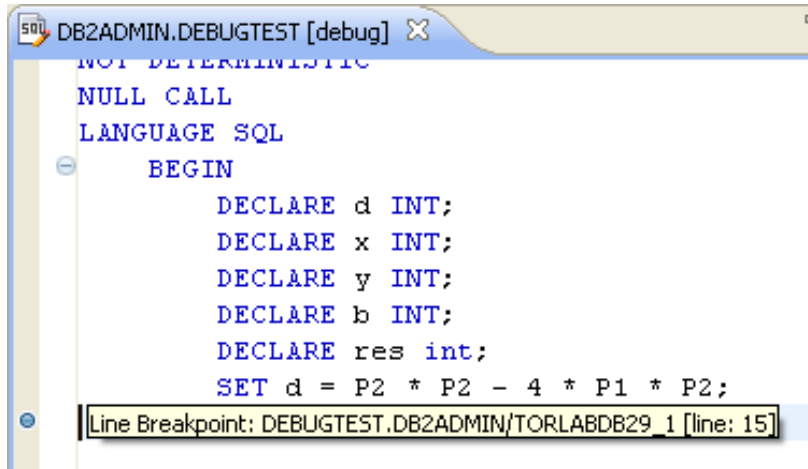
The Variables view displays the values of a specified variable during the debug process.

To view the value of variables:

- Click on a variable in the **Debug** results view. The value automatically appears in the **Variables** view.

Breakpoints

A breakpoint is a placeholder within code undergoing the debug process. It notifies the process to temporarily suspend execution. Breakpoints can be used to halt the debug process so modifications can be made to the code, or to change the debug execution point to only debug a certain section of a larger file.



Breakpoints are set to temporarily suspend the debug execution process, or to mark sections of code within a large file.

To set a breakpoint:

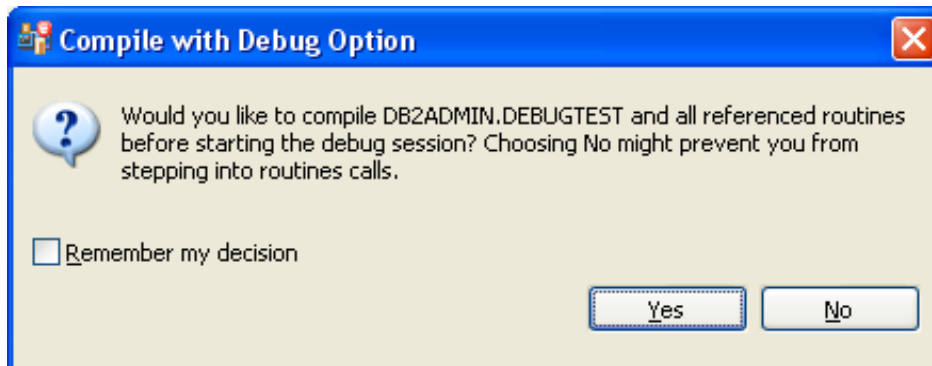
- Click the line in the code where you want to place a breakpoint. The breakpoint is indicated by a blue circle icon in the left-hand column.

To disable a breakpoint:

- Right-click on the breakpoint icon and choose **Disable Breakpoint**. The blue circle is replaced with a white circle, indicating that the breakpoint will no longer halt the debug process. You can enable a breakpoint again at any time via the **Enable Breakpoint** command in the context menu.

Dependencies

When you debug a stored procedure, all routines that are referenced by the procedure appear in the **Dependency** view. If these routines have not yet been compiled via the debug option, Rapid SQL Developer prompts you to compile the dependencies prior to executing the debug process.



Rapid SQL Developer automatically compiles dependencies for stored procedures if you have not already done so.

Cross-Language Debugging

The debugger features Java-to-SQL cross-language debugging. That is, debugging functionality in Java files that contain stored SQL routines.

In order for this feature to be active, Rapid SQL Developer must be installed as an Eclipse plug-in. Once you have added a Java application to the application, a number of configuration steps must be performed in order for cross-language debugging to work correctly.



```
62
63
64     private void executeProcWithParams( Connection conn ) throws SQLException
65     {
66         CallableStatement execute_stmt = null;
67         execute_stmt = prepareProcWithParams( conn );
68         execute_stmt.execute();
69
70         print( "[SQL OUTPUT]: " + execute_stmt.getString( 1 ) );
71
72         execute_stmt.close();
73     }
```

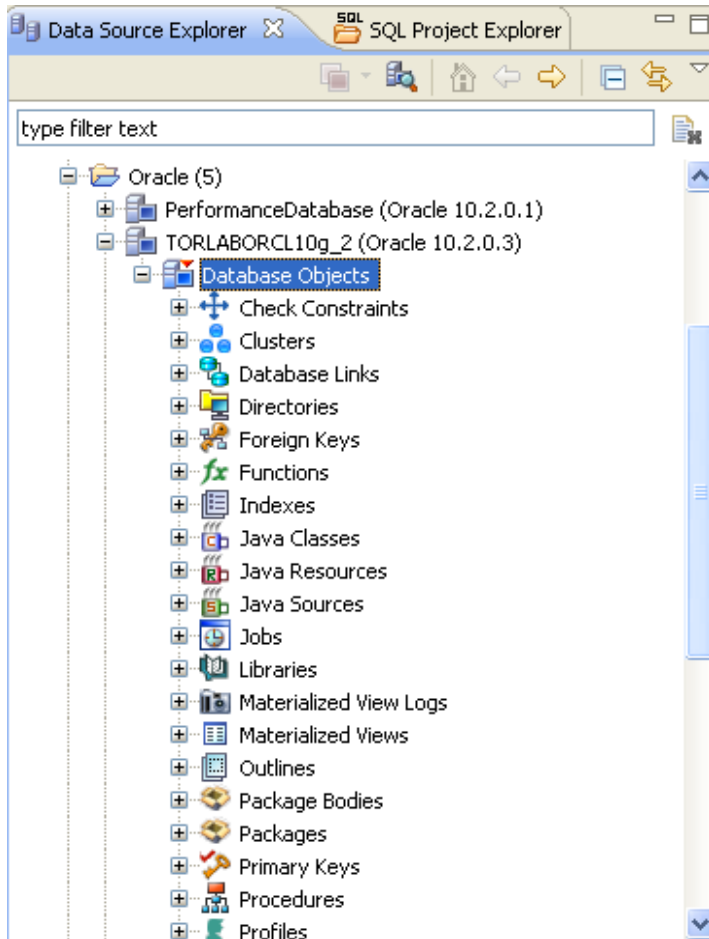
Cross-language debugging enables you to debug SQL procedures embedded within Java code applications.

Once you have associated the Cross Language Debugger library, added driver protocols, and created a new launch configuration for the java application you want to debug, Rapid SQL Developer will begin to debug the Java code. When it reaches the stored SQL procedure, you can step into the code and the system will begin debugging SQL in a smooth transition. When it is finished, it will resume debugging the rest of the Java application.

Enhanced Data Source Explorer

Rapid SQL Developer 2.0 provides enhancements to Data Source Explorer in the following categories:

- **Auto-Discovery and Import Features**
- **Search Functionality**
- **Filter Options**

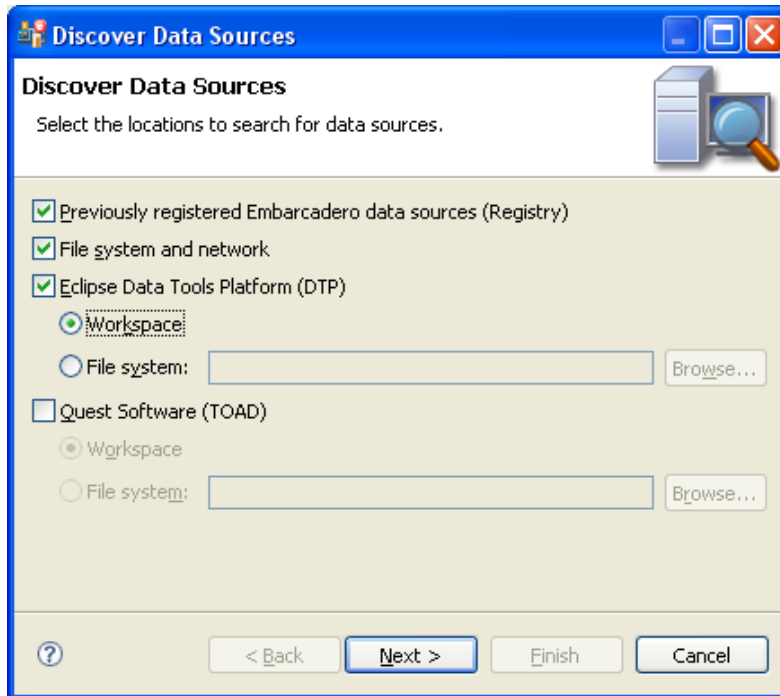


Data Source Explorer features enhancements to auto-discovery and import features, as well as more robust search and filter options.

Auto-Discovery and Import Features

When Rapid SQL Developer is started, it prompts you to discover data source catalogs that have been created by any previously installed Embarcadero products, or other instances of Rapid SQL Developer. Additionally, the system scans your machine for client software of supported third-party platforms (TOAD, Eclipse Data Tools Platform, etc.). These data sources are automatically added to the data source catalog, and appear in Data Source Explorer.

Once you are using Rapid SQL Developer, the application also features a **Discover Data Sources** command, which enables you to import a data source catalog from another source.



The Discover Data Sources command enables you to import a pre-existing catalog of data sources for registration purposes in Rapid SQL Developer.

The following sources are valid data source catalog formats for the purpose of adding data sources to your environment:

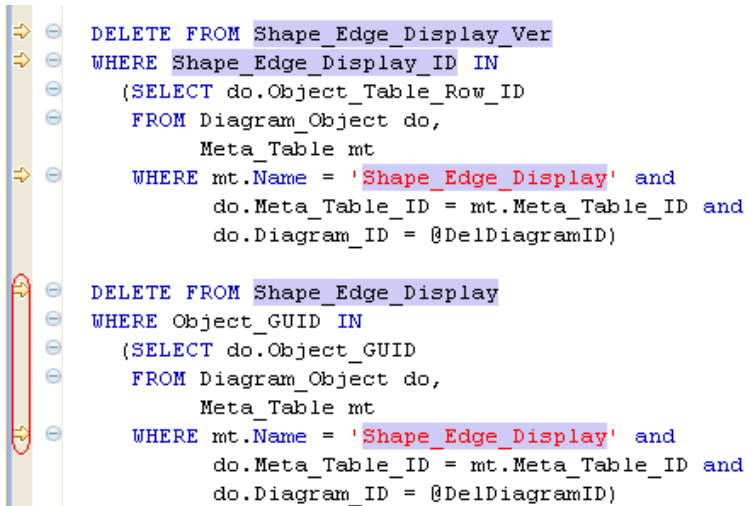
- **Embarcadero Windows registry (DBArtisan, Rapid SQL, DBOptimizer)**
- **Eclipse Data Tools Platform (DTP)**
- **Quest Software (TOAD)**

To import data sources:

1. Select **File > Import**. The **Discover Data Sources** dialog appears.
2. Choose **Embarcadero > Data Sources** from the import tree and click **Next**.
3. Choose a category from which you want to import the data sources. You can choose to import data sources from the DTP (Eclipse Data Tools Platform), an existing Embarcadero data source catalog stored in the Windows registry, or from TOAD. Click **Next**.
4. Specify the location of the source and click **Finish**. **Data Source Explorer** automatically populates with the new data sources.

Search

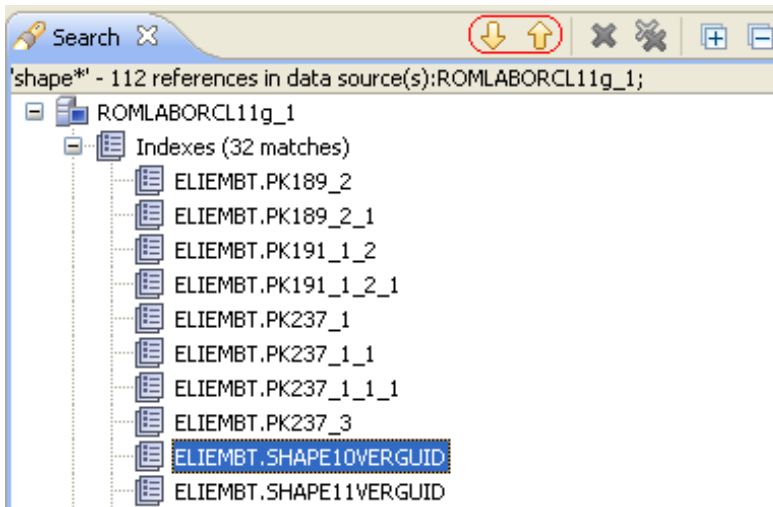
When you search for database objects from **Data Source Explorer**, search hits are now highlighted in the file with search annotations in the left-hand column. These annotations are represented by arrow icons, and enable you to better locate the code you are looking for.



The image shows two SQL queries in a code editor. The first query is: `DELETE FROM Shape_Edge_Display_Ver WHERE Shape_Edge_Display_ID IN (SELECT do.Object_Table_Row_ID FROM Diagram_Object do, Meta_Table mt WHERE mt.Name = 'Shape_Edge_Display' and do.Meta_Table_ID = mt.Meta_Table_ID and do.Diagram_ID = @DelDiagramID)`. The second query is: `DELETE FROM Shape_Edge_Display WHERE Object_GUID IN (SELECT do.Object_GUID FROM Diagram_Object do, Meta_Table mt WHERE mt.Name = 'Shape_Edge_Display' and do.Meta_Table_ID = mt.Meta_Table_ID and do.Diagram_ID = @DelDiagramID)`. In both queries, the string 'Shape_Edge_Display' is highlighted in red. On the left side of the editor, there are yellow arrow icons pointing up and down, indicating search annotations.

Search functionality has been improved by adding a number of comprehensive features to the Data Source Explorer search interface.

In addition to annotations, the Search function also features **Up** and **Down** arrows along the top of the **Search** view. These commands enable you to move between search hits within the same file or between files, if the search located multiple places containing the code.



Yellow arrow icons enable you to browse across files and instances of your search criteria.

To search for database objects:

1. Select **Search > Database** and specify the search criteria as prompted in the **Search** dialog.
2. Click **Search**. The results of your search are generated in the **Search** view. You can scroll through your results using the **Up** and **Down** arrows. When you open a matched file, references to the keyword are flagged with yellow arrow icons that appear in the left-hand column of the editor.

Filter

On Sybase and MS SQL Server platforms, SQL Filters can now be applied at the database level. This enables users to set different filters on different databases within the same data source.

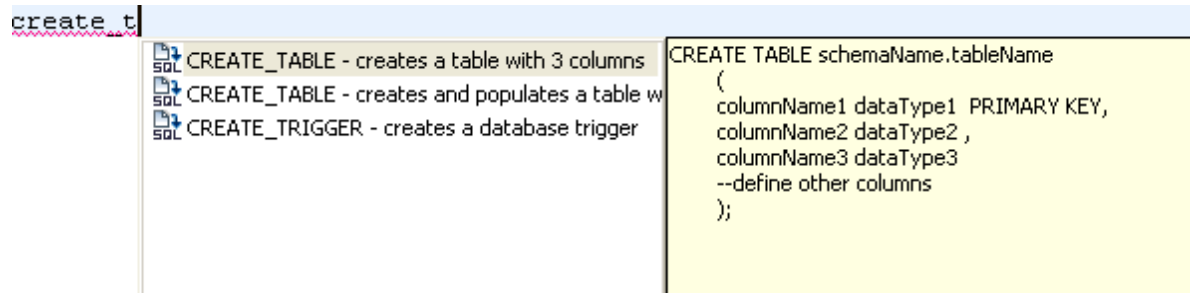
To filter at the database level:

1. In Data Source Explorer, right-click the data source and select Properties. The Properties dialog appears.
2. Select the **SQL Filter** node and de-select **Enable Data Source Specific Settings**. The other controls on the dialog become enabled.
3. Click **Add**. The **Filter Template** dialog appears.
4. Specify the parameters of the filter. Click **OK**. The filter property specification is added to the Filter Template.
5. When you are finished, click **OK**. The template name is added to the **Properties** dialog. It can be enabled or disabled by selecting the check box beside its name. Objects you have filtered out do not appear in Data Source Explorer unless the filter is disabled.

SQL Templates

Rapid SQL Developer provides code templates that enable you to select a DML or DDL statement schematic and drop it into SQL Editor. Placeholder syntax can then be replaced manually with statement variables, or be otherwise customized to meet your development needs.

Code templates are available for DML, ALTER, DROP, CREATE, and platform-specific commands. 273 templates come packaged with Rapid SQL Developer, across all platforms and versions. Additionally, you can modify or create new templates via the **SQL Templates** panel on the **Preferences** dialog.



SQL templates enable you to select a DML or DDL statement and then drop it into SQL Editor. Placeholder syntax is provided that you can replace manually or otherwise customize to your specific needs. There are 273 templates packaged with Rapid SQL Developer.

To use a template:

1. In SQL Editor, initiate the code assist feature by pressing **(Ctrl + Spacebar)**. Rapid SQL Developer automatically returns a list of template suggestions based on the code.
2. Select a template from the dialog. The code is inserted into SQL Editor with placeholder symbols that you are free to modify.

To create and modify a template:

1. Select **Window > Preferences > SQL Development > SQL Editor > SQL Templates** from the Menu. The **Templates** panel appears.

Templates		
Create, edit or remove templates:		
Name	Context ▲	Description
<input checked="" type="checkbox"/> ROLLBACK_TRA...	SQLServer (200...	rolls back an explicit or implicit transaction to the beginning of th...
<input checked="" type="checkbox"/> SAVE_TRANSA...	SQLServer (200...	sets a savepoint within a transaction
<input checked="" type="checkbox"/> TRUNCATE_TABLE	SQLServer (200...	removes all rows from a table without logging the individual row ...
<input checked="" type="checkbox"/> WAITFOR	SQLServer (200...	blocks the execution of a batch, stored procedure, or transactio...
<input checked="" type="checkbox"/> WHILE	SQLServer (200...	sets a condition for the repeated execution of an SQL statement..
<input checked="" type="checkbox"/> ALTER_DATABASE	SQLServer (2008)	Sets certain database behaviors to be compatible with the specif..
<input checked="" type="checkbox"/> SELECT	SQLServer (2008)	writes a SELECT statement with GROUPING SETS clause
<input checked="" type="checkbox"/> INSERT	SQLServer (2008)	writes an INSERT with multiple values list
<input checked="" type="checkbox"/> INSERT	SQLServer (2008)	writes an INSERT with dml_table_source clause
<input checked="" type="checkbox"/> MERGE	SQLServer (2008)	Performs insert or update operations on a target table based on ..
<input checked="" type="checkbox"/> MERGE	SQLServer (2008)	Performs insert or delete operations on a target table based on t..
<input checked="" type="checkbox"/> ALTER_DATABASE	Sybase (12, 15)	increases the amount of space allocated to a database

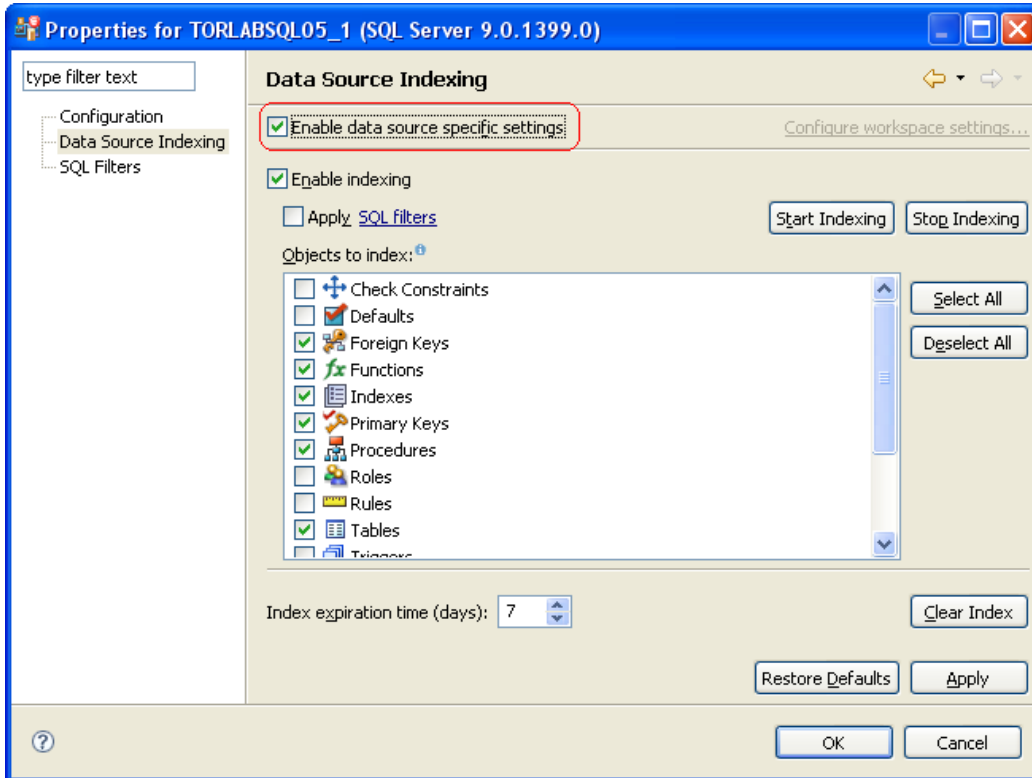
The **Templates** panel provides a list of SQL templates, as well as a description. It is here you create and modify them for your development needs.

2. Select the template you want to modify or change, or create a new template using the commands provided in the panel.

Data Source Indexing

The Data Source Cache has been renamed the Data Source Index to better reflect the nature of its functionality and to provide greater user comprehension with regards to its use and importance within the product.

Additionally, the data source index has been greatly improved to provide faster indexing, which translates into faster results when developing code in SQL Editor. To further improve the speed at which the application functions, the index now supports automatic incremental updating when CREATE, ALTER, and DROP statements are executed, or when Data Source Explorer is refreshed. This is a more efficient means of indexing, rather than having to re-index the entire system each time these commands are run in the system.



The renamed Data Source Index has been improved to provide faster indexing, and more granular choices on a data source specific basis.

Furthermore, Rapid SQL Developer 2.0 introduces indexing options that enable you to specify indexing at the individual data source level. This level of indexing can be specified on a data source's Properties panel, accessed via the right-click menu in Data Source Explorer.

To specify indexing at the individual data source level:

1. In **Data Source Explorer**, right-click on the data source you want to specify indexing for, and select **Properties**. The **Properties** dialog appears.
2. Choose **Data Source Indexing** and modify the properties, as required:
 - Choose **Enable Data Source Specific Settings** to indicate to Rapid SQL Developer that you want to specify individual indexing properties for this data source. Data sources that do not have this option selected will index under global indexing parameters.

- The tree view displays a list of database objects as they are organized in the Database Explorer view. Use the check boxes beside each object to specify the data sources that are to be included in the indexing process.
- Select **Clear Index** to delete the Object Index each time the application is started.
- Select the **Apply SQL Filters** tab if you want to apply any pre-defined filters to the index.
- If you are having performance problems due to a caching issue (such as a configuration error), the Stop Indexing, Clear Indexing, and Start Indexing buttons enable you to stop, clear, and/or restart the index process, respectively.
- The **Objects to include in index pane** contains a list of data source objects. Select or clear the check boxes beside each data source object to indicate the specific data source objects that are included and excluded, respectively, from the indexing process.
- The **Index Expiration Time (hours)** setting indicates that an indexing job will not start automatically until the specified number of hours have passed. The index can also be started manually via Start Indexing.

Results Sets

The SQL execution process has been enhanced to provide more comprehensive result views, as well as strengthen output file options through the enablement of different file format options when saving those results.

The **Results** view displays the outcome of any files that have been executed in Rapid SQL Developer. In Rapid SQL Developer 2.0, commands have been added to **Results** that enable you to specify the format in which these results are displayed.

Once you have executed SQL code, Rapid SQL Developer 2.0 enables you to export the results data into a specified file type, or view it on the application interface.

You can choose to export results into HTML, XML, Excel, and/or delimited text files, as needed, and/or view them from the interface in XML, HTML, or delimited text formats.

HTML

#	JOB_ID	FIRST_NAME	LAST_NAME	SALARY
1	PRE	James	Smith	800,000
2	MGR	Ron	Johnson	350,000
3	WOR	Fred	Hobbs	140,000
4	MGR	Susan	Jones	200,000

XML

```
<?xml version="1.0" encoding="UTF-8" ?>
- <rowset>
- <row>
  <job_id>PRE</job_id>
  <first_name>James</first_name>
  <last_name>Smith</last_name>
  <salary>800,000</salary>
</row>
- <row>
  <job_id>MGR</job_id>
  <first_name>Ron</first_name>
  <last_name>Johnson</last_name>
  <salary>350,000</salary>
</row>
</rowset>
```

The Results view enables you to view and save sets in different formats such as HTML or XML.

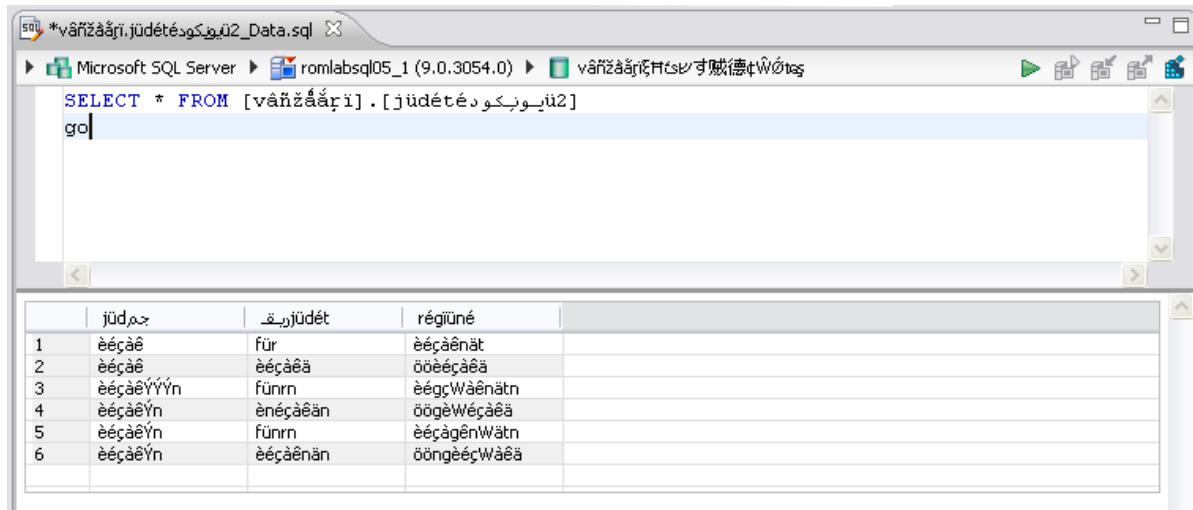
To export results to different file formats:

1. After you have executed a file, right-click on the **Results** view and select **Save Data**. The **Save Data** dialog appears.
2. Select the project name to which you want to save the results, enter a file name, choose the file parameters, and then choose a file format from the drop down menu. You can select from delimited text file, Excel, or HTML formats.
3. Click **Finish**. The results are saved in the directory location and format that you specified.

Text File Encoding (Unicode Support)

In the past, text file encoding, which is often platform specific, could be problematic for SQL code developers using Rapid SQL Developer, especially when files need to be shared across multiple systems and platforms, but developed on the Windows operating system from a single application.

Rapid SQL Developer 2.0 now provides full Unicode support, which means you can set the system to encode files edited within the development environment, to save under a different encoding format. Additionally, Rapid SQL Developer provides a further level of encoding granularity, by enabling you to specify or change different encoding formats for specific files, or groups of files, within the application.



Full Unicode support enables text file encoding in multiple formats across the files in your development environment.

By default, the Windows operating system sets text file encoding to use CP1252. If you need to change the encoding of the files you are editing in Rapid SQL Developer, the application now enables you to specify and change text file encoding globally, or on a file-by-file basis.

To set text file encoding globally:

1. Select **Windows > Preferences > General > Workspace** and click the **Other** option under **Text File Encoding**.
2. Use the drop down menu and select an encoding mode from the list provided. Click **Apply** to save your changes. All files in the application will now be encoded under your new selection.

To set text file encoding for a specified file:

1. Right-click on the file, folder, or project that you want to modify and choose **Properties**.
2. Modify the encoding selection on the **Resource** properties page that appears. The specified file now encodes under your new selection.

Additional Resources

Licensing Your Embarcadero Technologies Product

All Embarcadero Technologies products include a 14-day trial period. To continue using the product without interruption, we recommend that you license it as soon as possible. To license your product, use the License Request Wizard found in the Help menu of your respective product. If you have not yet purchased your Embarcadero Technologies product, contact sales@embarcadero.com, or uk.sales@embarcadero.com for sales in the EMEA region.

Embarcadero Technologies Product Support

The Embarcadero Technologies Web site is an excellent source for additional product information, including white papers, articles, FAQs, discussion groups, and the Embarcadero Knowledge Base. Go to www.embarcadero.com/resources, or click any of the links below, to find:

- [Documentation](#)
- [Online Demos](#)
- [Technical Papers](#)
- [Discussion Groups](#)
- [Knowledge Base](#)

Embarcadero Technologies Technical Support

If you have a valid maintenance contract with Embarcadero Technologies, the Embarcadero Technical Support team is available to assist you with any problems you have with our applications. Our maintenance contract also entitles registered users of Embarcadero Technologies products to download free software upgrades during the active contract period. Evaluators receive free technical support for the term of their evaluation (14 days).

We encourage you to open technical support cases via the [Technical Support request form](#) at the [Embarcadero Technologies Web site](#). For additional information about Embarcadero Technologies Technical Support, go to the Support page on our Web site.

Embarcadero Technologies on the Web

To download evaluations of other Embarcadero Technologies products or to learn more about our company and our products visit us at www.embarcadero.com.