SQL Power DQguru User Guide

Version 0.9.6

SQL Power Group Inc.
[http://www.sqlpower.ca/software/]

.

SQL Power DQguru User Guide: Version 0.9.6

SQL Power Group Inc. [http://www.sqlpower.ca/software]

.

Table of Contents

Chapter 1. Introduction

About DQguru

You need your data to be clean, complete, and reliable. DQguru will cleanse your data, validate and correct addresses, identify and remove duplicates, and build cross-references between source and target tables, providing you with complete and accurate data and a consolidated view of your customers, products, business units, and other essential information.

DQguru is an easy-to-use, highly intuitive tool created by data warehouse designers. DQguru contains many features geared for anyone dealing with information systems, including:

- Intuitive transformation interface allows you to easily manipulate data conversion workflows
- Accepts user-defined data matching criteria
- Innovative interface for verification of duplicates
- Merges duplicate records and their related data
- Allows for backup of affected records prior to merging of data
- Validates and corrects address information
- Runs against the entire database to perform initial data cleanup or can be incorporated into the data load process
- Builds cross-reference tables to link source system identifiers (primary keys) to target database identifiers

Whether you're building a data warehouse, data mart or a CRM system, DQguru helps you ensure the integrity of your data.

About This Guide

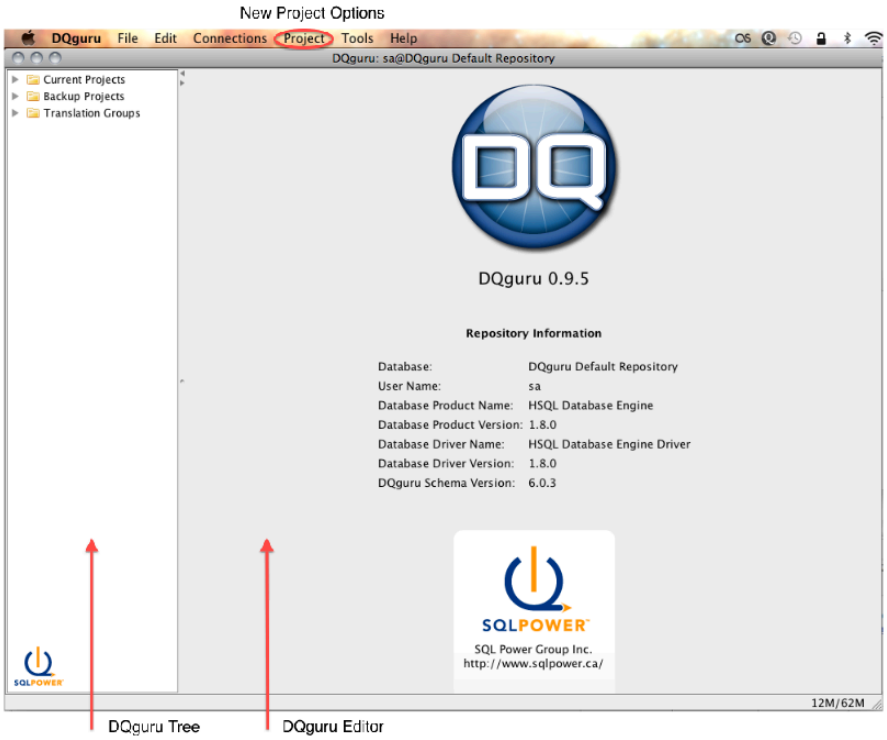
This user guide provides instructions for using DQguru. The guide assumes you are familiar with basic database operations and terminology (please refer to Chapter 7, *Glossary* for a list of some common database terms).

DQguru Licensing and Distribution

Chapter 2. Getting Started

Understanding the DQguru User Interface

DQguru contains two main areas: the Tree and Editor.



About the DQguru Editor

The DQguru Editor is your main work area. When you first start DQguru, information about the repository you're currently logged into is shown in the Editor. When you select an item in the DQguru Tree, details about the item are shown in the Editor.

About the DQguru Tree

The DQguru Tree contains the projects and translation groups within the current repository. You will use the Tree extensively when working on a project. You can click on items in the Tree in order to modify them (e.g., modify a transformation) or perform actions (e.g., run an engine).

The Tree includes three main folders:

- Current Projects - Active projects, organized in subfolders. For each project, the applicable processes, rules, engines, and other options are listed.
- Backup Projects - Backup copies of projects.
- Translation Groups - The translation groups and words you've created in the repository.

Using DQguru on Different Operating Systems

DQguru supports multiple operating systems, such as Windows, Macintosh and Linux. DQguru works the same on all operating systems, with a few minor exceptions:

- On Windows and Linux, CTRL is used as the accelerator key. On Macintosh, CMD is the accelerator key.
- On Windows and Linux, the DQguru menu bar is shown below the DQguru title bar. On Macintosh, the menu bar is shown at the top of the DQguru window.

Understanding DQguru Basic Concepts

About DQguru Projects

There are three types of DQguru projects that you can use to work with your data.

- De-duping project - Use to identify and merge duplicate records found within the same table.
- Cleansing Project - Use to reformat data to create a more uniform and professional appearance.
- X-refining Project - Use to create a cross-referencing table that links two different tables together. The cross-referencing table relates the primary key from one table to the primary key of the other table to show the relationship between the two records. Cross-reference projects will be implemented in a future release.

For more information, see the section called “Overview of Creating a Project” .

Transformations

Transformations are used to transform and manipulate your data. When you run a DQguru engine, DQguru will run each record from the source table through the transformations you've defined for the project.

Within a project, you can create one or more transformations A transformation must have at least one input step (usually the source table) and output step (either a results table or the source table), plus one or more intermediary steps that are used to transform data. You can use the following types of transformers:

- Address Correction - Splits address data into smaller chunks. Also provides validation.
- Boolean, Date, Number, String Constant - Returns a constant of the specified type, or a null value.
- Boolean, Date, Number to String - Returns a string from a boolean, formatted date, or number.
- Concatenation - Joins the contents of multiple inputs. The output is the result of the join.
- CSV Writer - Concatenates and writes the input values to a .csv file, separating the values using the given separator.
- Date to String - Formats a date as specified.
- Double Metaphone, Metaphone, Refined Soundex, Soundex - Codes the data phonetically according to the step's algorithms.
- Google Address Lookup - Looks up the given address string using a Google Maps API.

- Lowercase, Uppercase - Converts the data to the appropriate case.
- MD5 Checksum - Returns the calculated MD5 checksum of a string.
- Retain Characters - Retains only the specified characters and removes all other characters.
- Sort Words - Sorts a delimited list of words, outputting them using a specified output delimiter.
- String Substitution - Replaces all occurrences of a specified word with another word.
- String to Boolean - Compares a string against regular expressions or lists to determine a boolean return value.
- String to Date - Converts a string formatted as a date into a date type.
- Substring - Returns a part of the data.
- Substring by Word - Returns a subset of each word from the data.
- Translate Word - Replaces all occurrences of specified words with other words, as defined by the translation words you have set up. For more information, see the section called “Creating Translation Words and Groups” .
- Trim Spaces - Returns the string, with leading and trailing spaces removed.
- Word Count - Counts the number of words.

For more information, see the section called “Defining Transformations” .

Translation

You can use translation to search for one or more words and replace them with a different word (or words). For example, a common use is to replace abbreviations with the full term. You can define translation words and groups that you can use in any of your projects.

- Translation Word - Contains the instructions regarding the words to find and the replacement words.
- Translation Group - Collection of individual translation words. Typically contains related translation words.

For more information, see the section called “Creating Translation Words and Groups” .

Merge

Merge rules are used to combine duplicate records into single records. You can create column merge rules (which apply to a specific column) and table merge rules (a collection of column merge rules that apply to a table).

- Column Merge Rules - Rules that contain the merging instructions for a specific column in the database. Possible merging actions are: Use the master value, Augment nulls, Concatenate the values, Use the maximum value, Use the minimum value, and Set the sum as the value.
- Source Table Merge Rules - Essential rules that deal with the merging on the source table in the database.
- Related Table Merge Rules - Additional rules that deal with the merging on children, grandchildren or other related tables of the source table in the database.

For more information, see the section called “Creating Merge Rules” .

DQguru Engines

The DQguru engines transform data based on the processes and rules you've defined.

- Match Engine - Finds duplicate records. To do this, the engine processes data in a source table according to the transformations you've defined. The engine then determines which rows are exact duplicates and stores the results in a results table. The Match engine does not modify the data in the source table.
- Merge Engine - Combines or removes duplicate records. To do this, the engine modifies the data in a source table according to the merge rules you've defined.
- Cleansing Engine - Replaces records with reformatted data. To do this, the engine modifies the data in a source table according to the transformations you've defined.

For more information, see the section called “Running a DQguru Engine” .

Chapter 3. Working with DQguru

Overview of Creating a Project

General Guidelines

You would typically follow these steps to create a project.

1. Login to the repository where you want to store the project. For more information, see the section called “Logging into a Repository” .

   Note: You can use the DQguru default repository or create your own repository. For more information, see Chapter 5, *Working with Repositories* .

2. Create a new project and set the project parameters. For more information, see the section called “Creating a New Project” . Or to work with an existing project, simply login to the repository where the project is stored.

   Note: You can also import a project from another repository. For more information, see the section called “Importing and Exporting Projects” .

3. Define the transformations you want to run on your data. These can include translation rules. For more information, see the section called “Defining Transformations” and the section called “Creating Translation Words and Groups” .

4. Define the merge rules you want to run on your data. For more information, see the section called “Creating Merge Rules” .

5. Run the appropriate engine to apply the processes and rules you've defined. For example, in a De-duping project, you would run the Match engine to process the transformations you've defined. After validating the results, you would run the Merge engine to process the merge rules you've defined. For more information, see the section called “Running a DQguru Engine” .

General Steps for a De-duping Project

In a De-duping project, you would typically do the following:

1. Define the transformations that will be used to identify duplicate data.

2. Run the Match engine to find duplicate records in the source table, based on your transformations. DQguru stores the results in a new results table.

3. Validate the matches in the results table.

4. Define the merge rules that will be used to handle the duplicate data.

5. Run the Merge engine to combine or remove the duplicate data in the source table, based on your merge rules.

Note: To view a demo of creating a Find Duplicates project, go to www.sqlpower.ca/page/DQguru [http://www.sqlpower.ca/page/DQguru] .

General Steps for a Cleansing Project

In a Cleansing project, you would typically do the following:

1. Define the transformations that will be used to cleanse the data.

2. Run the Cleanse engine to process the data in the source table, based on your transformations.

General Steps for an Address Correction Project

In an Address Correction project you would typically do the following:

1. Define the location of the Address Correction Database in the user preferences. An Address Correction Database can be obtained from www.sqlpower.ca/page/dqguru_address_data [http://www.sqlpower.ca/page/dqguru_address_data] and is required for address correction. The Address Correction Database contains all known valid mailing addresses in Canada.

2. Define the transformations that will be used to identify addresses.

3. Run the Address Correction engine to find address records in the source table, based on your transformations. DQguru stores the results in a new results table.

4. Validate the addresses in the results table.

5. Run the Commit Validated Addresses engine to correct the address data in the source table, based on your merge rules.

Logging into a Repository

Before creating a project, you must login to the repository where you want to store the project. When you start DQguru, you are automatically logged into your default repository. For more information on changing your repository login preferences, see the section called “Changing Your Repository Startup Preferences” . For information on creating a new repository, see Chapter 5, *Working with Repositories* .

Note: You can have multiple repositories open at the same time, with each repository in a separate DQguru window.

1. Click Database » Connect to Remote Repository.

2. Select the repository and enter the user ID and password.

3. Click Login. The projects stored in the repository are listed in the DQguru Tree.

Creating a New Project

All of the active projects within a repository are stored in subfolders within the Current Projects folder in the DQguru Tree.

1. Login to the repository where you want to store the project. Or to work with an existing project, simply login to the repository where the project is stored. For more information, see the section called “Logging into a Repository” .

2. Select a folder in the Current Projects folder or create a new folder. To create a new folder, right-click the Current Projects folder, then click New Folder. Enter the folder name and description, then click Save.

3. Make sure the folder is selected, then click a "new project" button in the top toolbar (New De-duping Project, New Cleansing Project, or New X-refing Project). You can also right-click the folder, click New Project, then click the type of project.

4. Enter the project parameters, then click Save.

Importing and Exporting Projects

You can use DQguru's import and export features to backup and restore projects, migrate projects from one repository to another, or share projects with colleagues. Projects are exported in XML format.

To export a project:

1. Click a project in the DQguru Tree, then select Project » Export. You can also right-click a project, then click Export.
2. In the Export Project dialog box, select a location and enter a filename, then click Save.

To import a project:

1. To import an active project, select a subfolder in the Current Projects folder in the DQguru Tree.
2. Select Project » Import.
3. In the Import Project dialog box, select the project to import, then click Open.

Creating Translation Words and Groups

You can create a translation group (containing one or more translation words) to use in your transformation.

1. Select Tools » Translate Words Manager.
2. Click New Translate Group.

   Note: To modify an existing group, click the group in the Translation Groups folder in the DQguru Tree. By expanding the you can view the translation words in the group.
3. Enter a name for the group.
4. You can now add translation words to the group. In the From field, enter the word(s) you want to find. In the To field, enter the word(s) you want to use as a replacement.
5. Click Create Translation. The From and To words are added to the group.
6. You can continue adding translation words to the group. You can reorder the words using the arrows on the right side of the window, and you can delete words by selecting the words and clicking Delete Selected Translations.
7. When you're done, click Save Group. The translation group is added to the Translation Groups folder in the DQguru Tree.
8. You can obtain a standard translation group of words from Google if you click on Get Online List.

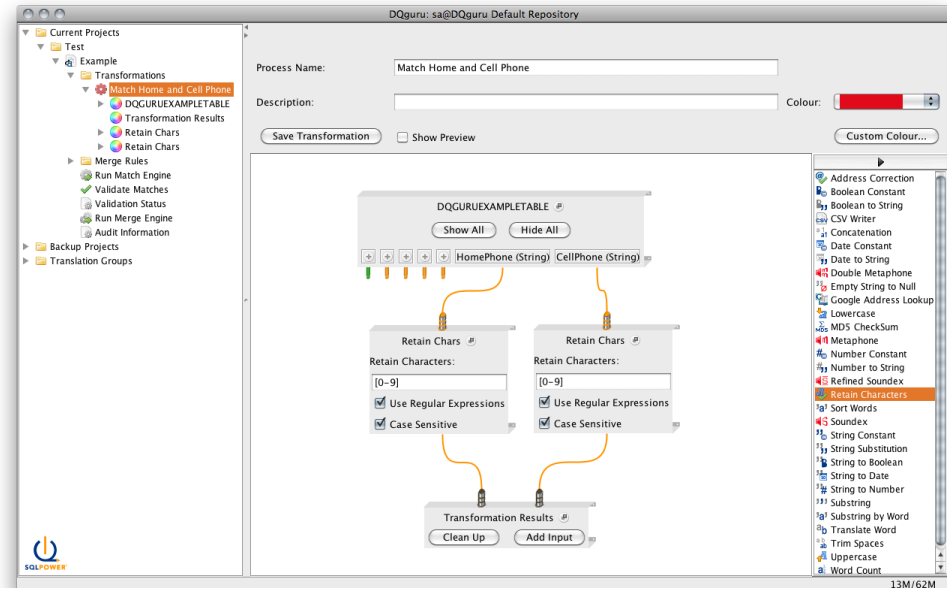
Defining Transformations

Within a project, you can create the transformations you want to use to transform your data. You can create as many transformations for a project as required.

1. In the DQguru Tree, expand the folder containing your project.
2. Right-click the Transformations folder, then click New Transformation. Or to modify an existing transformation, click the process in the Transformations folder.
3. Enter a name and description.

4. Select a colour. When you view the project results for a De-duping project, the results from each transformation will be identified by colour.

5. Use the large white area in the centre, called the Play Pen, to define the transformer to include in the transformation.



   - By default, the first step in the process (the input step) and the last step in the process (the output step) are already added to the Play Pen. The input step is typically one or more source tables. The output step may be a results table or a source table.

   - To view the columns in a source table, click Show All on the table.

   - To view the options in a results table, click the expand button on the table.

   - The transformers you can use to transform the data are shown on the right side of the Pen.

6. To add a transformer to the process, drag and drop a step into the Play Pen. You can also right-click in the Play Pen, click Add Transformer, then select the step.

   Note: You can combine a Translate Word step with another transformer. For example, you may want to concatenate first and last names using a Concatenation step, then strip out any "Mr." and "Mrs." salutations using a Translate Word step. You must set up translation words before you can add them to a transformation. For more information, see the section called “Creating Translation Words and Groups” .

7. To set the options for a step, click the expand button on the step and enter the options.

8. Connect the steps to the source and result tables by dragging the connectors from the top and bottom of the step to the appropriate tables.

9. Continue adding transformers as required.

10. Click Save Transformation.

11. The checkbox Show Preview gives the user an idea on the effect each transformer will have on the data going through it. With the checkbox selected, clicking on any one of the transformers will show a small preview on how that particular transformation will affect the first 5 rows of the column(s) attached to it.

Creating a PDF Copy of a Transformation

You can export a copy of a transformation diagram to a PDF file.

1. In the DQguru Tree, expand the folder containing your project.
2. Expand the Transformations folder, then click the process you want to export.
3. Select File » Export Play Pen to PDF.
4. Enter a filename and select a location, then click Save.

Creating Merge Rules

Within a De-duping project, you can create the merge rules you want to use to transform your data. Merge rules define how you want to deal with duplicate data and with the "child data" that references the duplicate data.

1. In the DQguru Tree, the source table for your project is shown in the Merge Rules folder. You can define merge rules for each column in the table.

   Note: To add another table to the Merge Rules folder, right-click the folder, then click New Merge Rule. Select the schema, table, index and catalog (if applicable), then click OK.
2. In the Merge Rules folder, click a table. The table columns are shown on the right side of the window.
3. Beside each column, select the action you want to perform on the column.
4. Click Save at the bottom of the window.

Running a DQguru Engine

After you've set up your project and defined transformers and/or merge rules, you can run a DQguru engine to apply the processes or rules you've defined. For more information on the engines, see the section called “DQguru Engines” .

1. Click a "run engine" button in the top toolbar (Run Match, Run Merge, or Run Cleanse). You can also click the appropriate "run engine" option in the DQguru Tree.
2. Enter the required parameters, then click the Run Engine button.

Example - Creating a De-duping Project

This section walks you through a step-by-step example to create a small De-duping project. For more information on creating projects, see the section called “Overview of Creating a Project” .

Note: To view a demo of creating a De-duping project, go to www.sqlpower.ca/page/DQguru [http://www.sqlpower.ca/page/DQguru] .

Step 1: Login to a Repository

Typically, you would first login to the repository where you want to store your project. For this example, you will use the DQguru Default Repository. You are logged into this repository by default.

Note: For more information on creating your own repository, see Chapter 5, *Working with Repositories* .

Step 2: Create a Sample Table

Next, you will create a sample table which you will use to practice finding and removing duplicate data. DQguru includes a feature you can use to quickly create this table.

1. Select Help » Build Example Table.
2. In the Data Source field, select the repository where you want to build the sample table. Leave this set to the DQguru Default Repository.
3. In the Schema field, select the schema to use for the table, such as Public.
4. In the Table field, enter the table name. You can leave this set to the default name, DQguruExampleTable.
5. Click Create.
6. In the Create Example Table dialog box, click Execute.

Step 3: Create a New Project

Next, you will create a new De-duping project.

1. First, create the folder where the project will be stored. In the DQguru Tree, right-click the Current Projects folder, then click New Folder. Enter the folder name and description, then click Save.
2. Next, create a new project within the folder. Right-click the folder, click New Project, then click New De-duping Project.
3. Enter a name and description for the project.
4. In the Source Table section, select the table you want to check for duplicate data. Select the catalog, schema and table for the DQguruExampleTable table you just created. (If you can't find the table you created, restart DQguru and try creating the project again.) You can leave the Unique Index field set to the DQguruExampleTable_pk, which is the primary key for the sample table.

   Note: The unique index option lets you choose which fields are needed to uniquely identify the records. The primary key is selected by default. If there is no primary key or the primary key is inappropriate, you can modify the unique index by clicking the Pick Columns button after you have saved the project.
5. In the Output Table section, select the table where the results of the data matching will be stored. You can leave the table name set to the default DQGURUEXAMPLETABLE_RESULT table and select a schema.

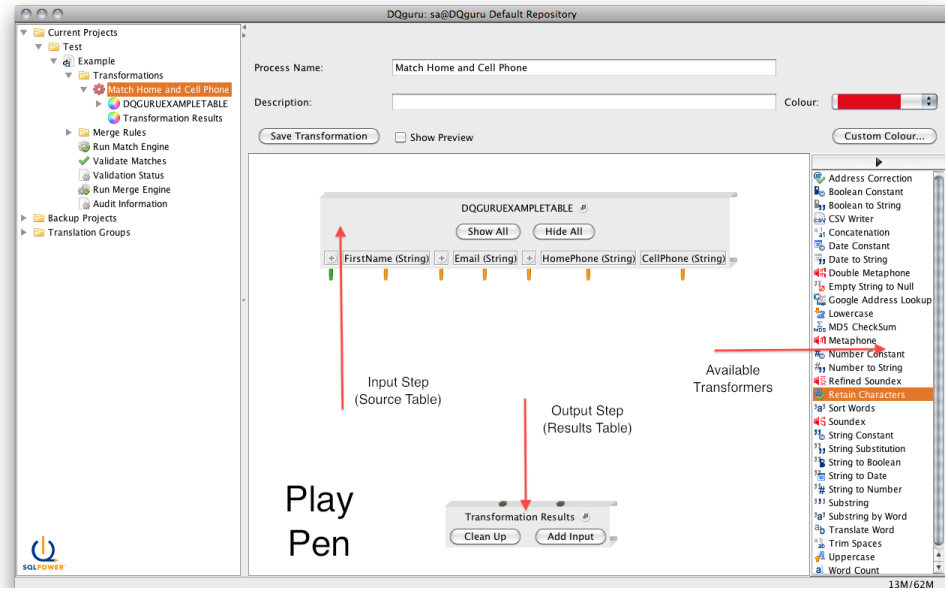
   Note: If you enter a table that does not exist, DQguru will create the table. If the table exists but is not correctly formatted, you will be given the option to drop the table and create a new one.
6. Click Save.

Step 4: Define Transformations

Next, you will create the transformations that DQguru will use to find duplicate records in the sample table.

1. In the DQguru Tree, expand the folder containing your project. Right-click the Transformations folder, then click New Transformation.

   You will use the large white area in the centre, called the Playpen, to define the transformers to include in the transformation.



   - The two default steps for a Find Duplicates project are already added to the Playpen. The first step (the input step) is the source table and is labelled with the table name. This step contains all of the initial data from your source table. The last step (the output step) is the results table and is labeled Transformation Results. This is where the transformed data will be stored so that it can be compared while matching.

   - The transformers you can use to transform the data are shown on the right side of the Playpen.

2. On the DQguruExampleTable source table, click Show All. All of the columns in the table are shown.

3. As a simple start, drag the connector under the HomePhone column to the top of the results table. This will match all records in the table that have an identical home phone number. This is a good start but not all that useful, as there are many ways to enter a phone number and this will only find a match between two phone numbers if they were entered in the exact same way.

4. Remove the connection between HomePhone and the results table by right-clicking the connection line, then clicking Delete. You can also drag the connector out of the results table and drop it in an empty space in the Playpen.

5. A more reliable way to find a match is by using a transformer. Drag the Retain Characters transformer into the Playpen. This transformer allow you to take in a string and remove all but the given characters.

6. Next, you must configure the Retain Characters step. Click the expand button beside the step name to view the step options. Because this a phone number, you will want to strip all of the characters except the numbers. In the Retain Characters field, enter 0123456789. Alternatively, you can use a regular expression. To do so, enter [0-9] and select the Use Regular Expressions check box.

7. Next, you must create a connection from the source table to the Retain Characters step and from the Retain Characters step to the results table. Drag the HomePhone connector in the source table to the connector on the top of the Retain Characters step. Drag the connector from the bottom of the Retain

Characters step to the results table. The transformation will now match all records with the same digits in the home phone number, regardless of how the phone numbers were entered. For example, the following numbers will be considered possible matches: (416) 555-1234, 416-555-1234 and 4165551234.

8. You will now add a second set of data to use in the comparison. Click the expand button on the results table. The step options appear. Click Add Input. An additional plug is added to the top of the results table. You can use this plug to connect an additional transformer to the results table.

9. Create an identical Retain Characters step in the Playpen. Connect this step to the CellPhone column in the source table and to the results table. Now any two records that have both a home phone number and a cell phone number with the same digits will be considered possible matches.

10. Enter a descriptive name for the transformation. Use a name that will allow you to easily identify this process later.

11. You can ignore the Description and Colour options for this example.

12. Click Save Transformation.

Step 5: Check for Possible Matches

Next, you will check for duplicates by running the Match engine. The Match engine will use the transformation you created to find duplicate records and will store the results in the results table. The Match engine will not modify your source data in any way.

1. In the DQguru Tree, within your project folder, click Run Match Engine.

2. You can leave all of the options set to the default settings. The following options are available:

   - Log File - Enter a filename and location for the log file. Information will be logged here when the Match engine is run.

   - Records to Process - Select the number of records to check.

   - Debug Mode - Select this option if you want to run the Match engine in debug mode.

   - Clear Match Pool - Select this option if you want to remove all matches found previously.

   - Message Level - Select the amount of information to be displayed in the results. If you set the message level to DEBUG, you should limit the number of records to process, since processing all records at the DEBUG message level can be very time-consuming.

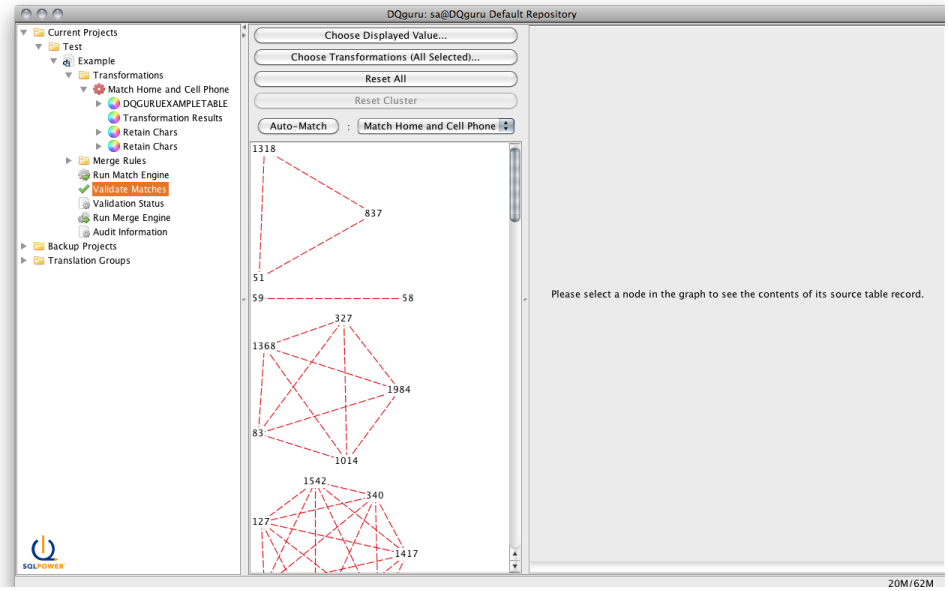
3. Click Run Match Engine. The engine runs and finds all records that share both an identical cell number and home phone number.

Step 6: Validate Matches

Next, you will review the matches to ensure they are valid duplicates.

1. In the DQguru Tree, within your project folder, click Validate Matches.

   The match groups are shown on the left side of the window. Each group contains records that have the same home and cell numbers. The records within the group are linked by dashed lines, showing the relationship between the records. The colour of the line represents the transformation responsible for the match. In this example, all of the lines will be the same colour since we used only one transformation.



2. By default, the records in the group are labelled by their unique index. You can choose to label the records using other values. For this example, you will use full name as a label. Click Choose Displayed Values, select the FirstName and LastName columns, then click OK.

3. Next, you will manually look at the matches and verify if they're really duplicates. Click on a record in the first group. On the right side of the window, the field values for the record are shown in the first column. The remaining columns contain the field values for the matching records. Fields that are different than the record in the first column are shown with a dark background and identical fields are shown with a light background.

4. The buttons above each column can be used to connect the records. The following buttons are shown above the first column: Master of All, No Match to Any, and Unmatch All. These buttons affect all of the records in the group. The following buttons are shown above the remaining columns: Master, Duplicate, and No Match. These buttons only affect the record in that column.

   Experiment with these buttons until you are satisfied with the matches in the first match group. As you experiment, the lines on the match group will change.

   - Dashed Line - The relationship is undecided.
   - Dotted Line - The records are not a match.
   - Solid Line (with a circle on one end) - The records are a match. The record with the circle is the master record and will be kept after the records are merged.

   Note that there is an Auto-Match button at the top of the window. Clicking this button will declare all undecided relationships as matches and a master record will be chosen at random. Using this option is not recommended in most cases, since serious data loss could occur if the matches are not manually checked.

Step 7: Merge Duplicate Records

Next, you will create merge rules and merge the records you've marked as matched. When you merge the records, the data in your source table will be modified.

A few notes about merge rules:

- You can define child and grandchild tables that will also be merged when the matched records are merged, to prevent orphaning records. There are no child tables in this example.
- When you click the Merge Rules folder, several buttons are shown at the bottom of the window. You can use the Derive Related Tables button to find other tables in the database that may be related to your source table (based on the primary key) and create merge rules for these tables. You can use the New Merge Rule button to manually find other tables and create rules. You do not need to use these buttons in this example.

1. In the DQguru Tree, within your project folder, expand the Merge Rules folder.
2. Click the DQguruExampleTable table in the Merge Rules folder. The table columns are shown on the right side of the window.
3. You will now define the action to be taken when merging two records for each column. For this example, you will leave the action for all columns as Augment nulls. This action will not modify the columns in the master record unless the column contains a null. If so, DQguru will attempt to fill in the missing data based on the duplicate records.
4. Next, you will merge the data. In the DQguru Tree, within your project folder, click Run Merge Engine. For this example, you can leave the Merge engine options set to the defaults.
5. Click Run Merge Engine. The engine runs and merges the matched records.
6. You can now view the status of the records. In the DQguru Tree, within your project folder, click Validation Status. The records that were matched are shown as merged. The records marked as no match or unmatched have not been modified.

Example - Creating an Address Correction Project

This section walks you through a step-by-step example to create a small Address Correction project.

If you want to walk through this example in the DQguru while reading along you will need a table with addresses in it. You will also need a copy of the Address Database and define it's location in the user Preferences. An Address Correction Database can be obtained from www.sqlpower.ca/page/dqguru_address_data [http://www.sqlpower.ca/page/dqguru_address_data] and is required for address correction.

For more information on creating projects, see the section called “Overview of Creating a Project” .

Step 1: Login to a Repository

Typically, you would first login to the repository where you want to store your project. For this example, you will use the DQguru Default Repository. You are logged into this repository by default.

Note: For more information on creating your own repository, see Chapter 5, *Working with Repositories* .

Step 2: Create a New Project

Next, you will create a new Address Correction project.

1. First, create the folder where the project will be stored. In the DQguru Tree, right-click the Current Projects folder, then click New Folder. Enter the folder name and description, then click Save.

2. Next, create a new project within the folder. Right-click the folder, click New Project, then click New Address Correction Project.

3. Enter a name and description for the project.

4. In the Source Table section, select the table you want to check for incorrect addresses. Select the catalog, schema and table for the table that contains address that need to be corrected.

   Note: The unique index option lets you choose which fields are needed to uniquely identify the records. The primary key is selected by default. If there is no primary key or the primary key is inappropriate, you can modify the unique index by clicking the Pick Columns button after you have saved the project.

5. In the Output Table section, select the table where the address validation information will be stored. This table will store the original values in the source table to user modified values. You can leave the table name set to the default and select a schema.

   Note: If you enter a table that does not exist, DQguru will create the table. If the table exists but is not correctly formatted, you will be given the option to drop the table and create a new one.

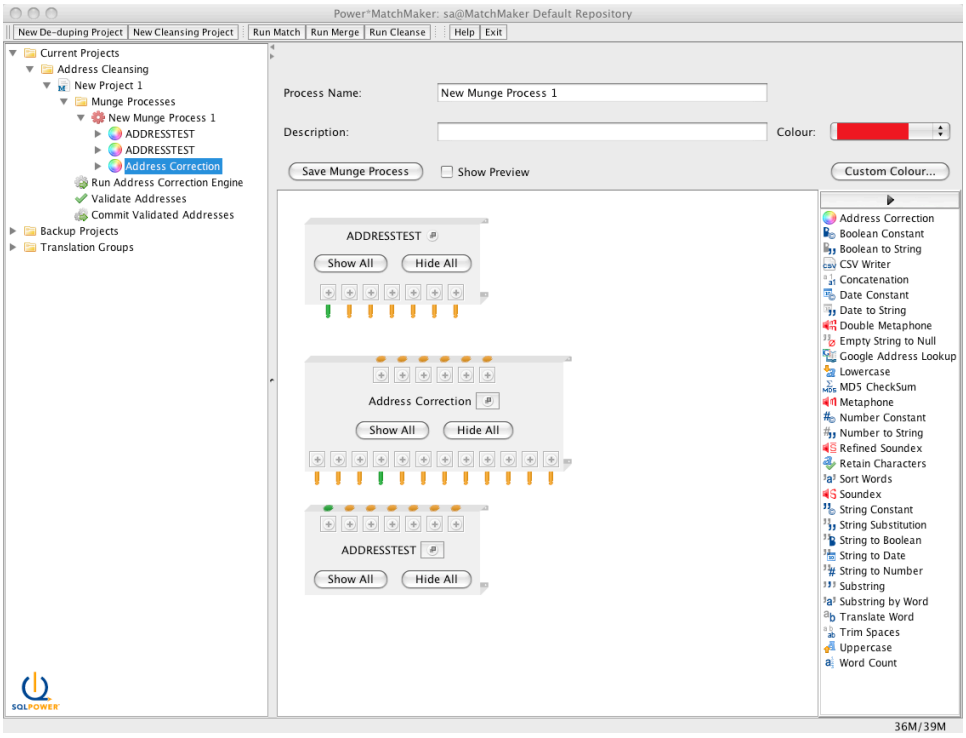
6. Click Save.

Step 3: Define Tranformations

Next, you will create the transformations that DQguru will use to find addresses in the table.

1. In the DQguru Tree, expand the folder containing your project. Right-click the Transformations folder, then click New Transformation.

   You will use the large white area in the centre, called the Playpen, to define the transformers to include in the transformation.



- The three default steps for an Address Correction project are already added to the Playpen. The first step (the input step) is the source table and is labelled with the table name. This step contains all of the initial data from your source table. The second step is the Address Correction step. This step decides if the address is valid and looks for suggestions. Depending on the engine settings this step will also write back automatically corrected addresses. The last step (the output step) is the same as the input step. This step should have values from the Address Correction step placed in it, possibly after transformations. This tells the Address Correction step how to write the addresses back to the table after modifications.

- The transformers you can use to transform the data are shown on the right side of the Playpen.

2. On the Address Correction step press the Show All button to see what each connection will give.

3. To feed data into this step some columns from the input table may need to be merged with Concatenation steps. At current the address line 1 is the connection that needs to be passed the street or route information. Address line 2 is considered additional data and is passed as is. Following address line 1 the municipality, province, country, and postal code must be given. If only Canadian addresses are being corrected and that information is not stored in the table a String constant step can be used to provide the country.

4. After plugging values into the Address Correction step, values need to be fed into the result step. Values can be placed into the result step as desired.

Step 4: Find Addresses to Correct

Next, you will find addresses by running the Addresss Correction engine. The Address Correction engine will use the transformation you created to find duplicate records and will store the results in the results table. The Address Correction engine may change the source data depending on the Validation Settings.

1. In the DQguru Tree, within your project folder, click Run Address Correction Engine.

2. The validation settings is one of the more interesting settings in the Address Correction engine. The validation settings decide what types of addresses to show for validation through the filter settings. Auto-validation can also be set from the validation settings if some addresses are to be automatically corrected.

   The Address Filter Settings have the following options:

   - Include all records - All records will be displayed in the Address Validation step even if the engine marks the address as valid.

   - Include SERP invalid or different format - Addresses that are defined as invalid based on SERP specifications will be displayed in the Address Validation step. Additionally, if a different format is found for the record by the parser than it will also be displayed in the Address Validation step.

   - Include only SERP invalid records - Addresses that are defined as invalid based on SERP specifications will be displayed in the Address Validation step.

   The Auto-validate Settings have the following options:

   - Do not auto-validate - No automatic correction of addresses will be done. The source table will not be modified and all corrections will be done in the Address Validation screen.

   - Auto-validate only SERP correctable addresses - This will write back to the source tables all addresses that are correctable based on SERP specifications.

- Auto-validate any address with only one suggestion - If an address is given only one suggestion by the Address Correction step then this setting will write that suggestion back to the source table.
- Auto-validate any address with a suggestion - If an address is given at least one suggestion by the Address Correction step then this setting will write the first suggestion back to the source table. The suggestions are ordered from most likely to least likely but it is not guaranteed that the first suggestion is not as likely as any subsequent suggestion.

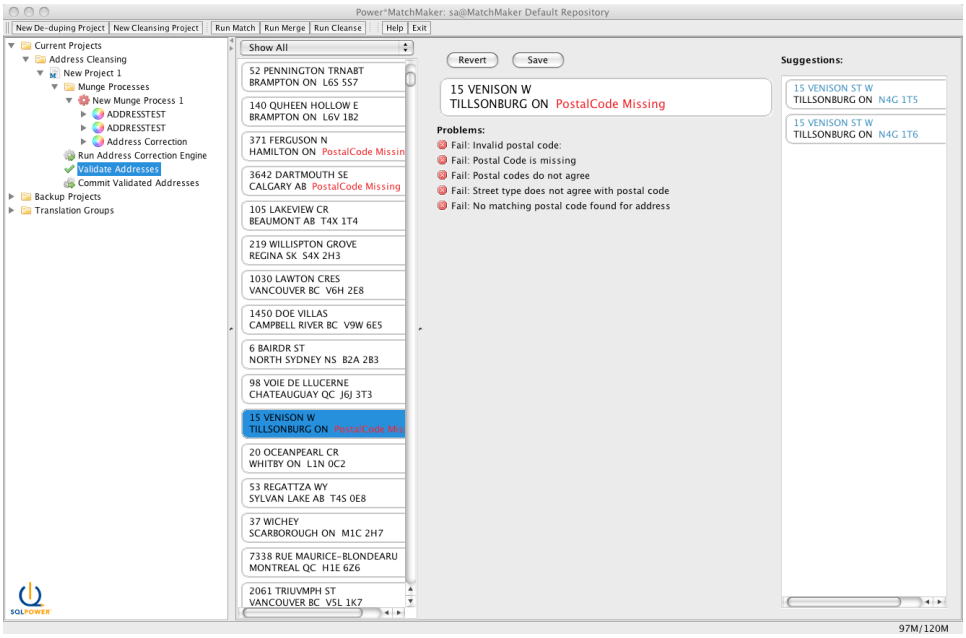
3. You can leave all of the other options set to the default settings. The following options are available:
   - Log File - Enter a filename and location for the log file. Information will be logged here when the Address Correction engine is run.
   - Records to Process - Select the number of records to check.
   - Debug Mode - Select this option if you want to run the Address Correction engine in debug mode.
   - Clear Match Pool - Select this option if you want to remove all addresses found previously.
   - Message Level - Select the amount of information to be displayed in the results. If you set the message level to DEBUG, you should limit the number of records to process, since processing all records at the DEBUG message level can be very time-consuming.
4. Click Run Engine. The engine runs and finds all addresses that needs some correction.

Step 5: Validate Addresses

Next, you will review the addresses to ensure they are valid. Additionally corrections can be done manually and suggestions can be selected.

1. In the DQguru Tree, within your project folder, click Validate Addresses.

   The addresses are shown on the left side of the window. Each address can be selected from the left list to show an editor for that address on the right of the list.



2. Selecting an address from the list shows the address editor to the right of the list. Below the address in the editor is the list of errors the address currently has. To the right of the address is a list of suggestions that are offered by the Address Correction step.

   To correct the address select an address from the suggestion list or click on parts of the address to edit it by hand. The most likely suggestions are at the top of the suggestion list while less likely suggestions appear lower. When the address is correct a green check mark will appear to the left of the address. Each change will save the address. To revert the address to what was originally in the source table press the revert button.

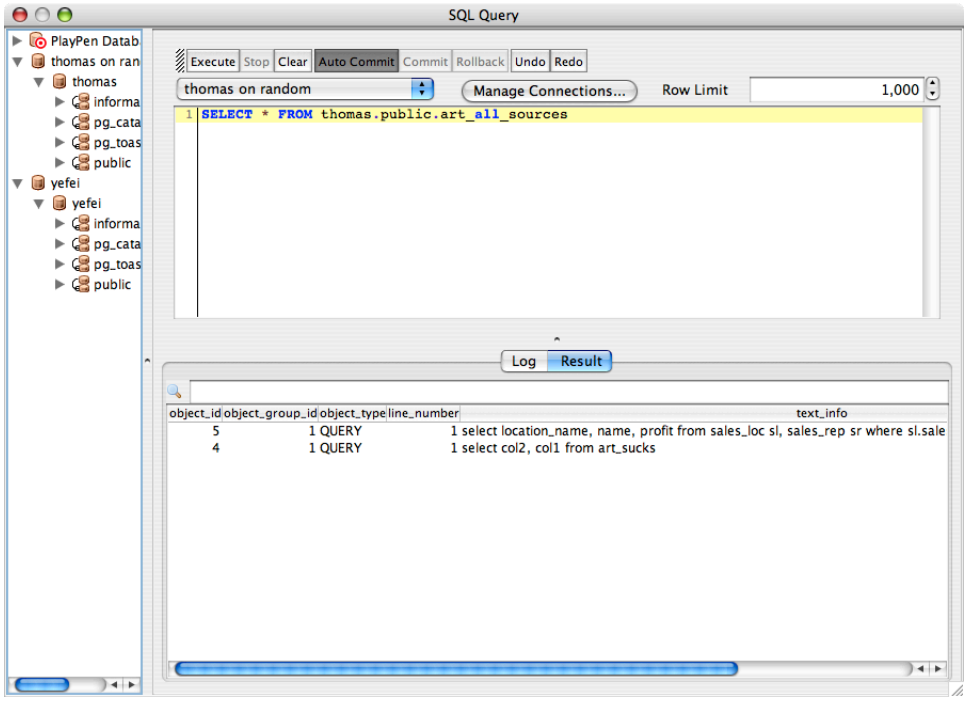
3. When an address is corrected and is valid a green check mark will appear to the left of it in the list. Once all of the addresses have been corrected the addresses can be committed to the source table. Select the Commit Validated Addresses from the project tree and press the Run Engine button to write all modified addresses to the source table.

Universal SQL Access

Universal SQL Access is a "fall-back" tool that lets you work at the raw SQL command level. This is an advanced topic and should only be used by those familiar with the intricacies of SQL commands and the details of your database.

Universal SQL Access is started from the menu entry under the Tools menu, and begins with the GUI window shown below. The first thing you should do is select which database connection you wish to use. The list of Connections is the same as the main program uses, as set up in the JDBC Connections window.

The basic steps to using Universal SQL Access are to type a command in the top (SQL Command) window and click the Execute button; the results are displayed in the bottom (SQL Results) window. To save you some typing, you can just drag the databases, tables and columns shown in the DB tree of the Universal SQL Access and drop on the top (SQL Command) window.



SQL Statements can be entered multiple at a time, can be more than one line long, but be careful! if you have multiple commands, do end the previous ones with a semicolon, which is unnecessary if there is only

one command. These statements are not interpreted by Universal SQL Access itself, so anything that the given database and driver accepts can be used. For example, with Oracle, you can use PL*SQL statements. With most drivers you should be able to use stored procedures. Each SQL statement is executed in its own transaction context if you turn the Auto Commit on, that is, changes are committed immediately (so be careful!). If you turn the Auto Commit off, you can decide to Commit or Rollback later.

Output (Results) Window

Command Output appears in the SQL Output window. A scrollbar will appear if the information cannot all be seen at once.

A visual indication of the success or failure of the command is displayed below the output: green for success, red for failure. As well, failures will be accompanied by a pop-up window containing details on the failure.

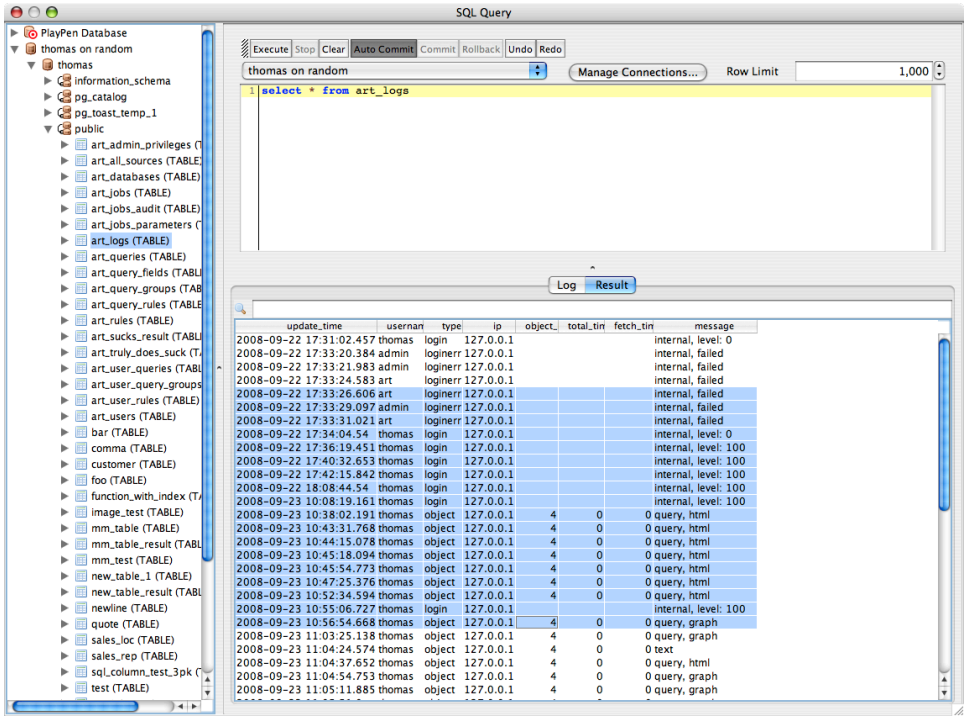
The Clear Output button clears the contents of the output window.

Output Formats

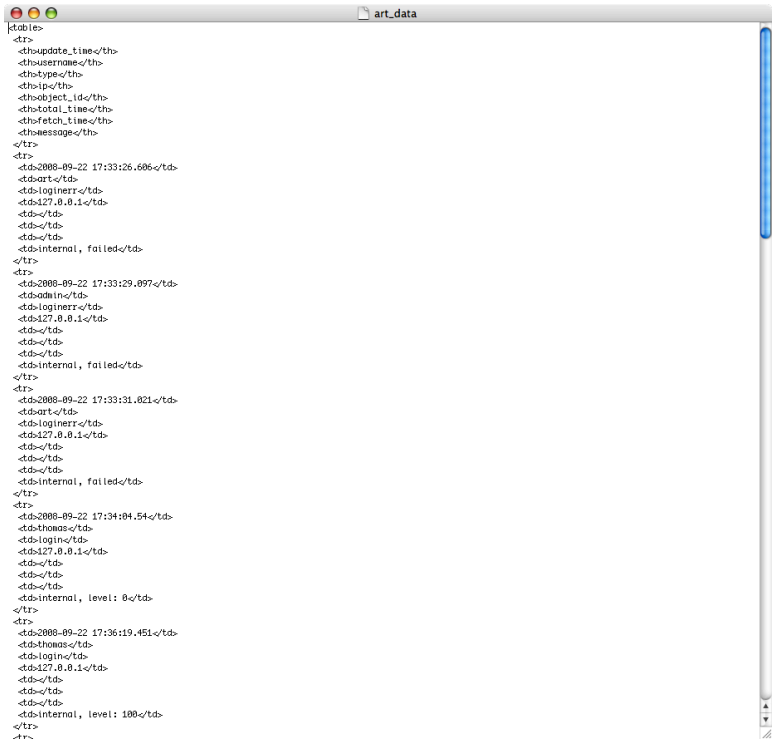
The output format in the result window is always Table mode, which provides a friendlier interface which ensures all of the columns are lined up properly. If you right-click on a table or multiple tables,you can generate a HTML file or CSV file. HTML mode generates an HTML table to display the results of a Select. CSV mode generates a CSV file, which is a specially formatted plain text file which stores spreadsheet or basic database-style information in a very format.

CSV files are often used as a simple way to transfer a large volume of spreadsheet or database information between programs, without worrying about special file types. For example, transferring a home-made address book from power*architect into a database program such as Filemaker Pro could be done by exporting the file as a CSV from Excel, then importing that CSV into Filemaker.

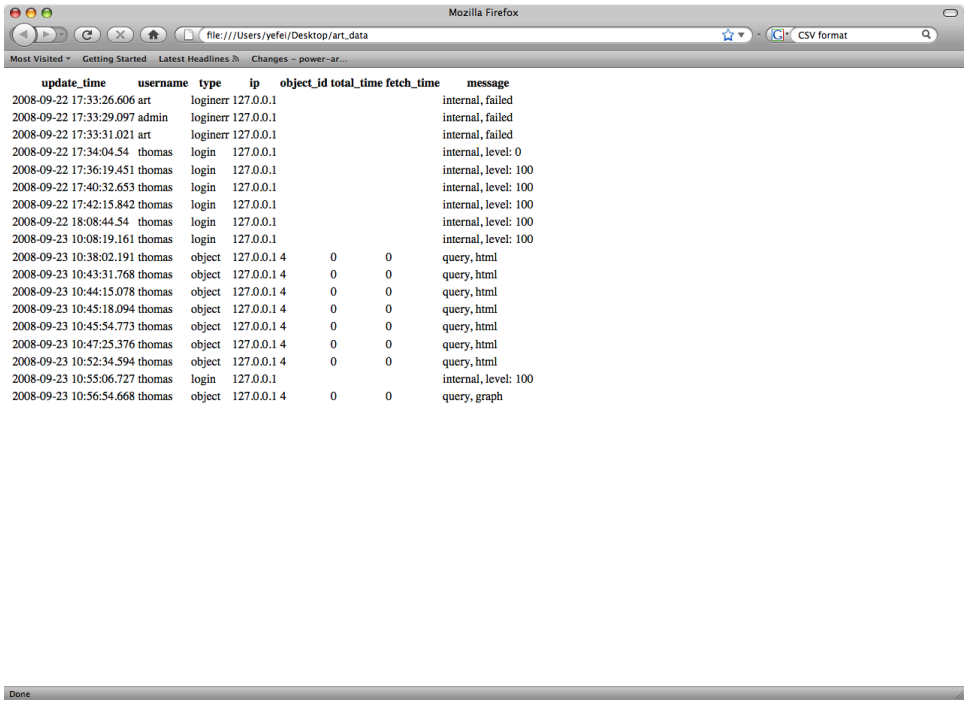
For example, a "select * from art_logs" looked like this:



You can view this same data in HTML just by right-click the selected table and choose "Exported selected to HTML..". When tried to open this file just by text editor, it looked like this:



when used a web browser to open this file, the output liked this:



update_time	username	type	ip	object_id	total_time	fetch_time	message
2008-09-22 17:33:26.606	art	loginerr	127.0.0.1				internal, failed
2008-09-22 17:33:29.097	admin	loginerr	127.0.0.1				internal, failed
2008-09-22 17:33:31.021	art	loginerr	127.0.0.1				internal, failed
2008-09-22 17:34:04.54	thomas	login	127.0.0.1				internal, level: 0
2008-09-22 17:36:19.451	thomas	login	127.0.0.1				internal, level: 100
2008-09-22 17:40:32.653	thomas	login	127.0.0.1				internal, level: 100
2008-09-22 17:42:15.842	thomas	login	127.0.0.1				internal, level: 100
2008-09-22 18:08:44.54	thomas	login	127.0.0.1				internal, level: 100
2008-09-23 10:08:19.161	thomas	login	127.0.0.1				internal, level: 100
2008-09-23 10:38:02.191	thomas	object	127.0.0.1	4	0	0	query, html
2008-09-23 10:43:31.768	thomas	object	127.0.0.1	4	0	0	query, html
2008-09-23 10:44:15.078	thomas	object	127.0.0.1	4	0	0	query, html
2008-09-23 10:45:18.094	thomas	object	127.0.0.1	4	0	0	query, html
2008-09-23 10:45:54.773	thomas	object	127.0.0.1	4	0	0	query, html
2008-09-23 10:47:25.376	thomas	object	127.0.0.1	4	0	0	query, html
2008-09-23 10:52:34.594	thomas	object	127.0.0.1	4	0	0	query, html
2008-09-23 10:55:06.727	thomas	login	127.0.0.1				internal, level: 100
2008-09-23 10:56:54.668	thomas	object	127.0.0.1	4	0	0	query, graph

With a bit of formatting, or even a CSS style sheet, this HTML page could be made quite usable.

Universal SQL Access is not perfect, but it is adequate for many purposes involving direct use of SQL (We will add more useful features to it later).

Chapter 4. Setting up Database Support

There are several reasons you will need to connect to a database when using DQguru. When creating a new project, DQguru must connect to the database containing the data you want to work with. You can also save your DQguru project in a remote repository; if you choose to do this, DQguru must connect to the repository.

In order to retrieve data from a database or save your DQguru project to a remote repository, you must first set up a connection to the database where the data or repository is located. Setting up a database connection involves the following steps:

1. Define general settings and drivers for the database platform (such as SQL Server or Oracle). For more information, see the section called “Setting up Database Types” .

2. Create a connection for a specific database. This connection uses the general settings and drivers you have configured for the database platform. For more information, see the section called “Setting up Database Connections” .

Setting up Database Types

You must define general settings for the database platforms used by your databases (such as SQL Server, MySQL, Oracle, DB2, etc.). These settings will be used by DQguru when you set up a connection to a specific database.

Note: At this point you are configuring general settings only and are not setting up a connection to a specific database. For more information on creating a database connection, see the section called “Setting up Database Connections” .

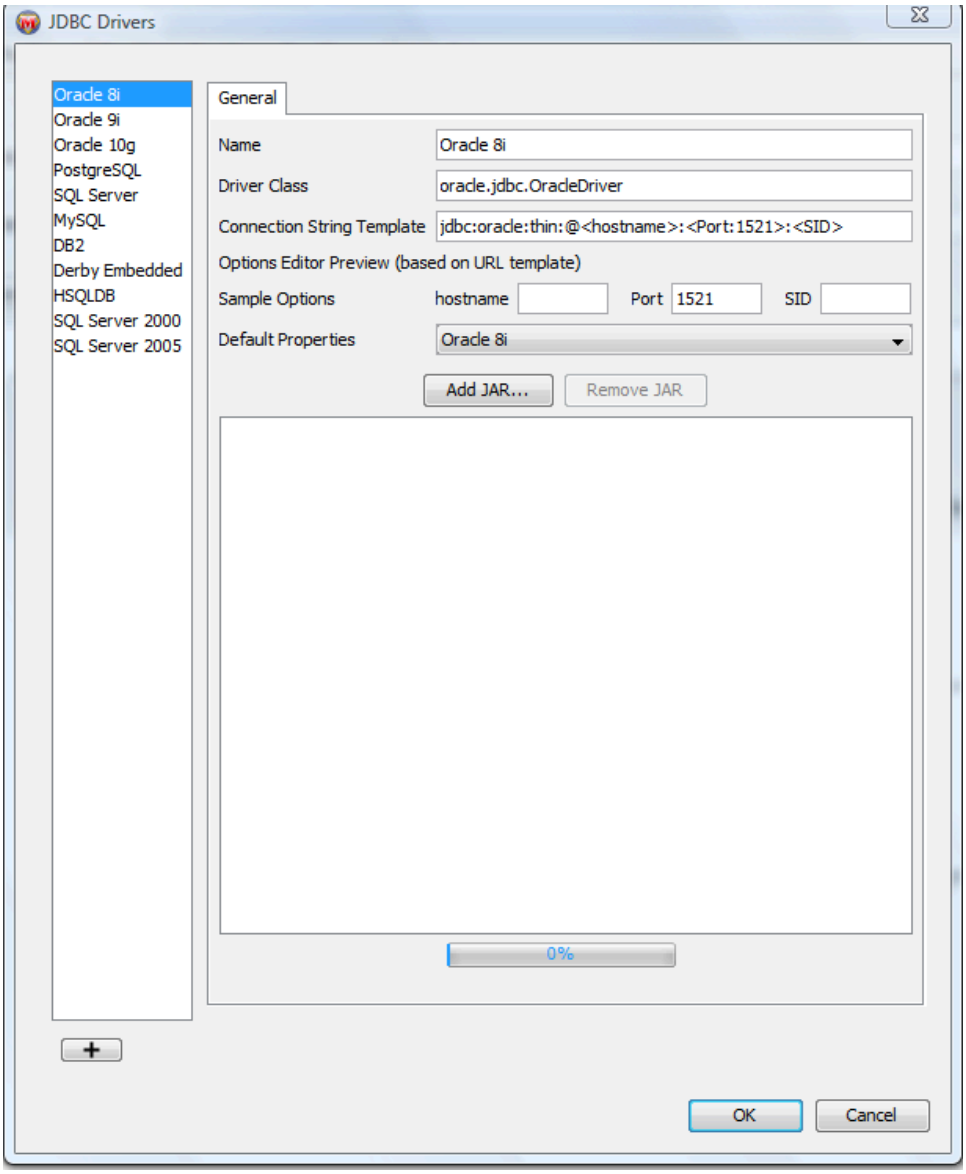
General settings for several database platforms are already pre-configured in DQguru. If you plan to work with one of these database platforms, all you need to do is define the location of the JDBC driver. For more information, see the section called “Defining the JDBC Driver” . You can also define additional database platforms in DQguru. For more information, see the section called “Adding a New Database Type” .

Adding a New Database Type

To add a new database type:

1. Select Database » Manage Database Connections. The Database Connection Manager dialog box appears.

2. Click JDBC Drivers. The JDBC Drivers dialog box appears. Existing database types, including the pre-defined database types included with DQguru, are listed on the left.



Note: You can modify an existing database type by clicking on it in the list.

3. Click + below the list of database types.
4. Enter the following information on the General Tab:

Name	Name for the database type (for example, PostgreSQL or SQL Server).
Driver Class	Java class name of the driver. This is the driver class within the JDBC driver JAR file that will be used for database connections.
Connection String Template	General format of the JDBC URL for the database platform. Important: You are not creating a connection for a specific database - you are entering a generic

In this field ...	Enter the following information ...
	connection string that applies to the database platform. Later on, when you set up a connection to a specific database, DQguru will use this template to create the URL to connect to the database. The connection string template must conform to a specific pattern that includes literals and variables. • Literals are entered like normal text but may not contain angle brackets (< or >), which are reserved for defining variables. As the name implies, literals appear in the URL in the same position and way they appear in the template. • Variables are used to for values that change often, such as the schema or database name you wish to connect to. To define a variable in the template, use the format <variable_name:default_value> (to include a default value) or <variable_name> (if you don't want to include a default value). If you use a default value, it is entered automatically when you create a database connection. You can modify the value if the database you are connecting to is configured to use a different value. Each variable you define is shown below the Connection String Template field. This provides you with a preview of the values you will be able to modify when creating a database connection. For example, the connection string template to connect to a Microsoft SQL Server database might look like this: jdbc:sqlserver://<Hostname>:<Port:1433> When you create a connection to a specific SQL Server database, DQguru will use this template to create the connection URL. In this example, the template will create the URL "jdbc:sqlserver://:1433", where 1433 is the default port value. Since SQL Server databases listen to port 1433 by default, it makes sense to include this value in the template. When you're creating the actual database connection, you can change the port value if the database you're connecting to is configured differently.

5. Click OK.

Next, you must define the location of the JDBC driver for the database type. For more information, see the section called “Defining the JDBC Driver” .

Defining the JDBC Driver

Whether you are adding a new database platform to DQguru or want to use one of the pre-configured platforms, the last step in setting up a database type is to locate the JAR file (or files) that contain the JDBC drivers for the database platform.

Note: At this point, you are just telling DQguru where the drivers are. You must set up a database connection in order to connect to a specific database (for more information, see the section called “Setting up Database Connections”).

Unlike most applications, which need a distinct driver program to communicate with each type of database, DQguru uses Java-based drivers. These drivers normally come from the database vendor in the form of JAR (Java Archive) files. JAR files are an extension to the file format used by PKZip/WinZip archives.

Most database platforms provide drivers that are fully backward compatible. This means that it is best to use the newest driver available, regardless of the software version on the specific database server you intend to connect to. One exception to this is the Oracle database. It is important to match the major version number of your JDBC driver with the major version number of the Oracle database server you connect to. For example, if you are connecting to an Oracle 10g database, use the latest Oracle 10g driver. If you are connecting to an Oracle 9i database, use the Oracle 9i driver.

To define the JDBC driver for a database type:

1. If you do not have the JDBC driver for a specific database platform, you can usually obtain one from the database vendor. If that fails, you can find a directory of databases drivers on Sun's web site [http://developers.sun.com/product/jdbc/drivers] . There is also a permanent thread in the SQL Power user support forum [http://www.sqlpower.ca/forum/posts/list/401.page] where you can share information with other users about finding and configuring drivers for a particular database platform.

2. Decide on a permanent location to store your JDBC drivers. A good strategy is to create a JDBC folder under your Documents folder and collect all of you JDBC driver files there.

3. Save the JDBC driver (it will usually be one or more JAR files) in the location you've chosen.

4. If the JDBC Drivers dialog box is not already open, select Database » Manage Database Connections, then click JDBC Drivers.

5. Select a database type from the list on the left side of the dialog box.

6. Click Add JAR.

7. Locate the JAR file and click Open. If there is a valid driver class in the JAR file, a file tree will appear showing the JDBC driver classes within the JAR file.

8. Select the driver you want to use.

9. Click OK.

Setting up Database Connections

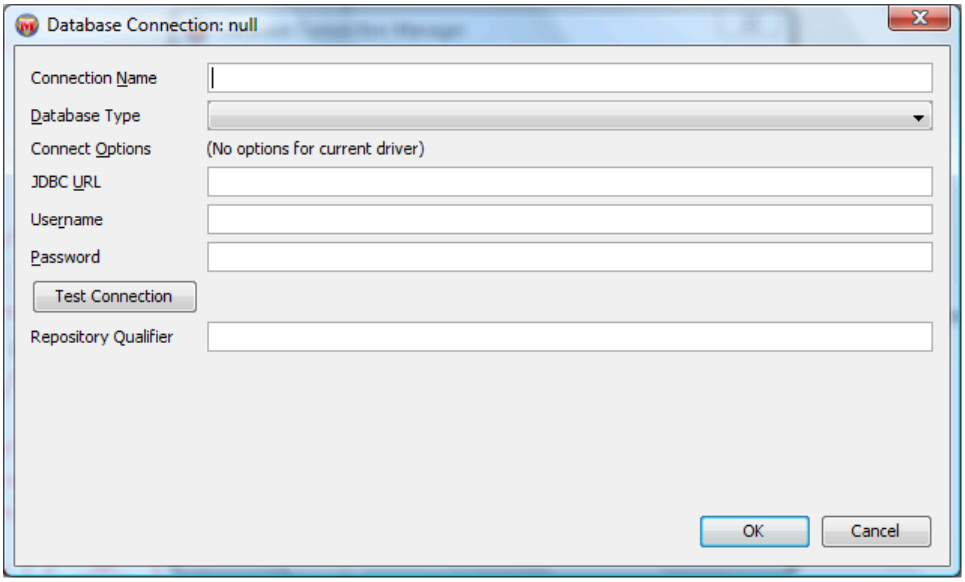
You must set up a database connection for each database DQguru will need to connect to.

Before creating a connection, you must define the general settings for the database platform. For more information, see the section called “Setting up Database Types” .

Creating a New Database Connection

To create a new database connection:

1. Select Database » Manage Database Connections, then click New. The Database Connection dialog box appears.



2. Enter the following information:

Connection Name	Enter a name for the database connection.
Database Type	Select the database platform used by the database you want to connect to. Note: This list contains the database types you defined previously. For more information, see the section called “Setting up Database Types” .
Connect Options and JDBC URL	Enter the connection options for the database driver. These options are based on the database type you select. If you are using one of the fully-supported drivers, the connection option parameters are added into the JDBC URL field in the order that the Java driver expects to see them (this string is sometimes called a "db URL" in Java terminology). In the following example: • The default port number from the database type has been entered automatically in the Connect Options. (You would not usually change a default value unless the database server you're connecting to has been configured to use a different value.)

In this field ...	Do this ...
	• The hostname and database name have been entered manually in the Connect Options. • The PostgreSQL driver is being used.
Username and Password	Enter the username and password to connect to the database.
Repository Qualifier	If the database connection is for a repository, enter the repository qualifier. This information will automatically be prefixed to the repository tables. The value is dependent on the database platform. • In Oracle, this would be the schema owner. • In DB2, this would be the schema name. • In SQL Server 2000, this would be the database name, followed by "." and then the owner. • In SQL Server 2005, this would be the database name, followed by "." and then the schema name. • In MySQL, this would be the database name. • In PostgreSQL, this would be the database name, followed by "." and then the schema name.

3. Click OK. The new connection is added to the Database Connection Manager.

Modifying or Deleting Database Connections

You can modify a database connection's properties or permanently delete it.

To modify a database connection:

1. Select Database » Manage Database Connections. Select a database connection, then click Edit. The Database Connection dialog box appears.

2. Modify the connection settings. For information on the settings, see the section called “Creating a New Database Connection” .

3. Click OK.

To permanently delete a database connection:

1. Select Database » Manage Database Connections.

2. Select a database connection, then click Remove.

Chapter 5. Working with Repositories

About Repositories

Your DQguru projects are saved in a special data store called a repository. The first time you run DQguru, a local repository is created for you automatically, called the DQguru Default Repository. The repository itself is actually a relational database schema and is stored in an embedded database, located in a directory called .mm (notice the leading dot) within your home directory. On Unix systems, you can find your home directory by issuing a cd command with no arguments. On Windows systems, your home directory is usually in C:\Documents and Settings\ *Your Name* . Be sure to backup this repository regularly. Simply make a copy of the .mm directory when DQguru is not running and save this copy in a safe location.

There are many advantages to using this local repository: you don't require a network connection, you can easily backup and restore the repository, it's fast, and it's already been set up for you. However, there are times you may want to save your work in a remote, shared repository. For instance, if you are working on a large De-duping project as part of a team, you may want to divide up the work and tackle it in parallel. Or if your cleansing process is part of a nightly maintenance routine that runs on a server, you may want to point the routine to the same repository you work in. Or perhaps there is a DBA at your organization who makes backups nightly and your project data would be safer in her system than it would be in a local database on your workstation or laptop.

If you decide to use a remote repository, you must create the repository, then connect to it using DQguru. See the following sections for details.

Creating a Remote Repository

Create a Database and User

Before creating a remote repository, you must decide where to store it. Storing a shared repository on a workstation is not recommended for several reasons. The entire team will rely on one person's workstation, and that person will be responsible for keeping their system up and running and performing regular backups of the repository database. They will also be inconvenienced when the database software needs to be upgraded. Hosting the repository on a dedicated database server is a much better choice.

Once you've chosen a location, you must choose the database software you will use to create the repository. You can create a repository using PostgreSQL, MySQL, Oracle, Microsoft SQL Server, or HSQLDB. If available, you can use a database server that already exists on your network. Otherwise, there is a wide range of free and commercial database platforms to choose from. If you have to install a new database and you're unsure which to choose, the DQguru development team recommends PostgreSQL. It's robust, flexible, performs well, is easy to set up and administer, and it's free.

Note: If you'd like to experiment with keeping the repository in a different type of database, contact the DQguru developers on the web support forum [http://www.sqlpower.ca/forum] .

Use the database software's vendor-supplied tools to create the database you will use to house your repository and to create a new database user for DQguru. This user is typically called mm but you can use any name.

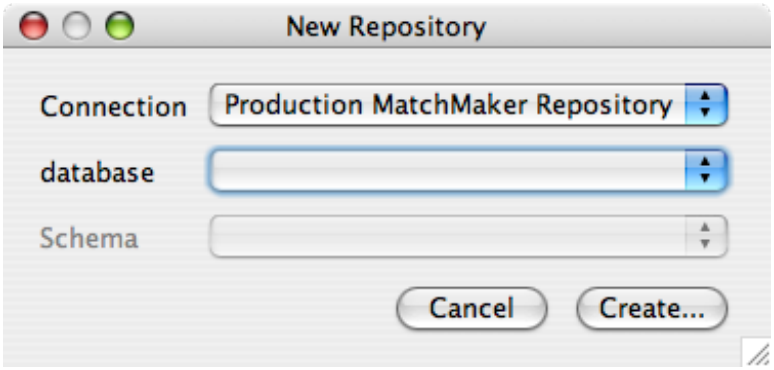
Create a New Data Source

You must create the data source that DQguru will use to connect to your new repository. For details, see ??? .

Create the Repository Tables

Once you've created your repository database and data source, you must create the tables in the repository.

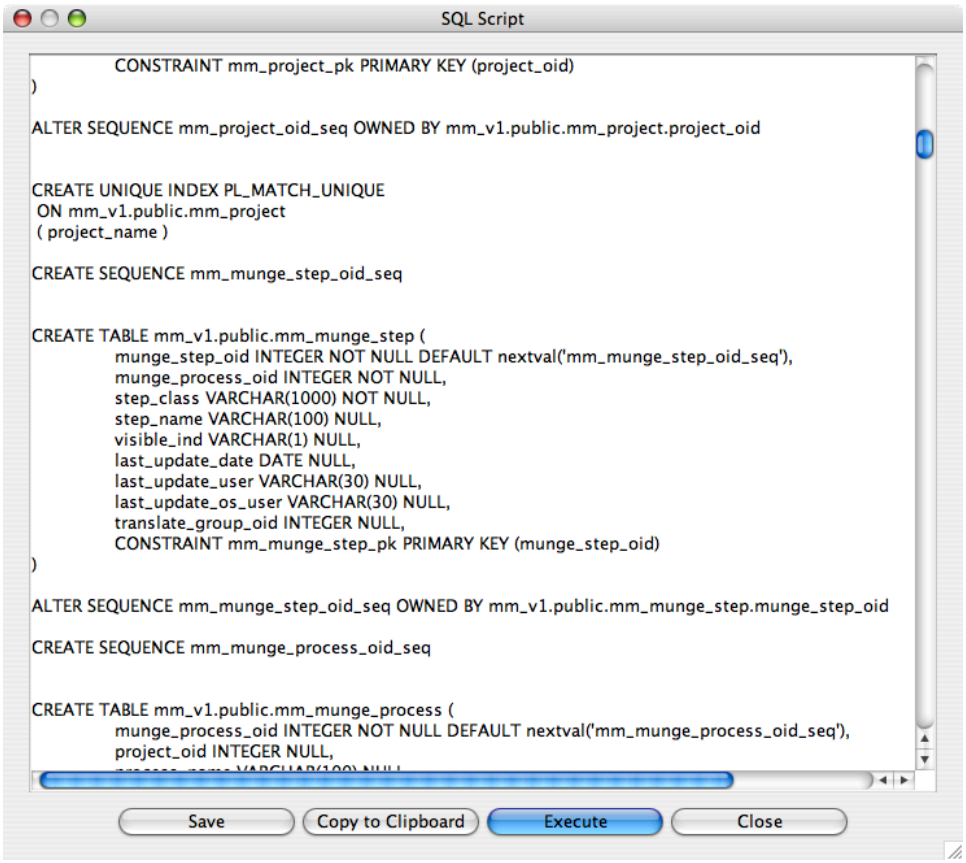
1. Select Database » Create Repository. The New Repository dialog box appears.



2. In the Connection box, select the data source you created for the repository.

3. Select the catalog and schema for the database, if applicable.

4. Click Create. In a few moments, a SQL scripts appears. This script will be used to create the repository tables.



5. Click Execute to run the script. Alternatively, you can save or copy the script and then use third-party tool to run it.

If any of the statements in the script fail, an error message will appear showing the SQL statement that failed and the database vendor's error message. If the statement failed due to permissions problems, you could use a vendor-supplied tool to grant the necessary permissions, then try the failed statement in an external tool. Once the statement executes properly, you can come back to DQguru and resume the script. If the statement fails for other reasons which don't appear to be recoverable, you will have to abort the script and visit our support forum for assistance.

Once the script finishes executing, a message will appear showing how many statements were attempted and how many succeeded. Assuming the script executed successfully, your remote repository has now been created.

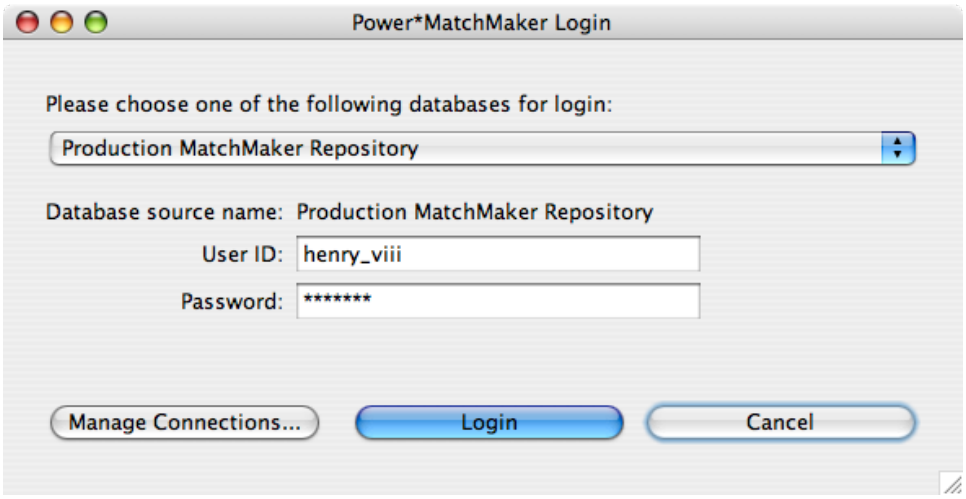
Connecting to a Remote Repository

Once you've created a remote repository, you must connect to it in DQguru. You can then start saving your projects in the repository.

DQguru allows you to connect to any number of repositories at the same time. Each active repository connection is referred to as a session. Each session is shown in a separate DQguru window and the window's title bar identifies the repository. The name of the user you're connected as, followed by an @ sign, followed by the repository's data source name, is shown in the title bar.

To start a new repository session:

1. Select Database » Connect to Remote Repository. The DQguru Login dialog box appears.



2. Select the data source for the repository you want to connect to. For more information, see ??? .

3. Enter the user ID and password for the repository.

4. Click Login. A window appears containing the new repository session.

Note: When you start a new repository session, any other active repository sessions remain open as well. To close a repository session, simply close the session's window.

Using Concurrent Sessions Within the Same Repository

One of the main advantages of using a shared repository is that multiple DQguru users can work within it at the same time. However, users must be careful not to overwrite changes made by others. To prevent this, each user should work on a different part of the project. For example, if you are working on one transformation and a co-worker is working on a different transformation within the same project, you will not interfere with each other. However, if you are both working on the same transformation, whoever saves last will "win," trampling the other person's version of the same transformation.

Real time collaboration is planned for a future version of DQguru, so users within the same repository can see all changes immediately. Until this feature is implemented, ensure that all users working on the same project are aware which part of the project each user is modifying.

Changing Your Repository Startup Preferences

By default, DQguru will login to the DQguru Default Repository at startup. You can change your preferences so that you login to a remote repository automatically when DQguru starts or so that you are prompted to choose a repository at startup. For example, if you typically work in the same remote repository all the time, you may want to login to that repository each time you start DQguru. Or, if you typically work on projects in several different repositories, you may want to be able to choose a repository each time you start DQguru.

1. Select File » User Preferences. On Macintosh systems, select the DQguru application menu, then select User Preferences (you can also use the standard keyboard shortcut Command-comma). The User Preferences dialog box appears.

2. If you want to login to a repository automatically at startup, select Automatically connect to this repository and choose a repository.

3. If you want to select a repository at startup, select Show the Login Dialog.

4. Click OK. Your changes will take effect the next time you start DQguru.

Chapter 6. Engine Runner Tool

The Engine Runner Tool is a command-line tool recently added to the DQguru for running a DQguru project's engine without having to use the GUI. It should be located in the directory where the DQguru was installed as the file named `dqguru-engine-runner.jar`. This is an executable JAR file, so you will need to use the `java` command to run it on the command-line.

IMPORTANT: You need to setup your DQguru projects through the GUI *before running this tool* . This tool will only be able to run on already pre-defined transformations in projects. It is also important that you set your engine settings exactly as you want them in the GUI before running the tool, because you cannot override them with the command-line interface at current.

Here's an example usage of the tool:

```
java -Xmx1024m -jar dqguru-engine-runner.jar --repository "DQguru Default Repository" --username "sa" --password "" --project "My Project"
```

Here's a piece-by-piece explanation of the command:

- java - The Engine Runner is a Java program, so you will need to use Java to run it.
- -Xmx1024m - You can optionally specify Java Virtual Machine arguments after the java command. For example, this sets the maximum memory the program may use. Increase the number if the Engine Runner keeps running out of memory.
- -jar dqguru-engine-runner.jar - Specifies the executable JAR file that actually contains the command-line program.
- --repository "DQguru Default Repository" - This is where you specify the name of the datasource where you have stored your DQguru repository. Note that if it has spaces in it, you need to put it in quotes (" ").
- --username "sa" - This is where you specify the username for the datasource where your repository is stored.
- --password "" - This is where you specify the password for the datasource and username that you're logging in with. If the password is nothing, you will still have to enter empty quotes ("").
- --project "My Project" - This is the project name with the engine you want to run. Note that if it has spaces in it, you need to put it in quotes (" ").

You may also run the program with a "--help" flag to get a list of all possible arguments that you can give to the program.

```
java -jar dqguru-engine-runner.jar --help
```

Chapter 7. Glossary

This section lists some database-related terms and their meanings.

Some of these terms are from FolDoc, "The Free On-line Dictionary of Computing", http://www.foldoc.org/, Editor Denis Howe.

Column	The set of all instances of a given field from all records in a table [http://foldoc.org/foldoc/foldoc.cgi?table] .
Database	One or more large structured sets of persistent data, usually associated with software to update and query [http://foldoc.org/foldoc/foldoc.cgi?query] the data. A simple database might be a single file containing many records [http://foldoc.org/foldoc/foldoc.cgi?records] , each of which contains the same set of fields [http://foldoc.org/foldoc/foldoc.cgi?fields] where each field is a certain fixed width.
Data Modeling	The product of the database design process which aims to identify and organize the required data logically and physically.
Data Warehouse	A database, often remote, containing recent snapshots of corporate data. Planners and researchers can use this database freely without worrying about slowing down day-to-day operations of the production database.
ETL	Extract, Transform and Load, the process of maintaining and transforming data into and out of a relational database.
Foreign key	A column [http://foldoc.org/foldoc/foldoc.cgi?column] in a database table [http://foldoc.org/foldoc/foldoc.cgi?table] containing values that are also found in some primary key [http://foldoc.org/foldoc/foldoc.cgi?primary+key] column (of a different table). By extension, any reference to entities of a different type. Some RDBMSs [http://foldoc.org/foldoc/foldoc.cgi?RDBMSs] allow a column to be explicitly labelled as a foreign key and only allow values to be inserted if they already exist in the relevant primary key column.
Identifying Relationship	Where the key of the parent table is a subset of the key of the child table.
JDBC	Java Database Connectivity, an unofficial acronym for the "java.sql" package of functionality used to access relational databases from programs written in the Java programming language.

Key	A value used to identify a record [http://foldoc.org/foldoc/foldoc.cgi?record] in a database, derived by applying some fixed function to the record. The key is often simply one of the fields [http://foldoc.org/foldoc/foldoc.cgi?fields] (a column [http://foldoc.org/foldoc/foldoc.cgi?column] if the database is considered as a table with records being rows - see key field [http://foldoc.org/foldoc/foldoc.cgi?key+field]). Alternatively the key may be obtained by applying some function, such as a hash function [http://foldoc.org/foldoc/foldoc.cgi?hash+function] , to one or more of the fields. The set of keys for all records forms an index [http://foldoc.org/foldoc/foldoc.cgi?index] . Multiple indices may be built for one database depending on how it is to be searched.
Primary key	The candidate key [http://foldoc.org/foldoc/foldoc.cgi?candidate+key] selected as being most important for identifying a body of information (an entity, object or record [http://foldoc.org/foldoc/foldoc.cgi?record]).
Record (row)	One or more structured sets of persistent data, usually associated with software to update and query [http://foldoc.org/foldoc/foldoc.cgi?query] the data. A simple database might be a single file containing many records [http://foldoc.org/foldoc/foldoc.cgi?records] , each of which contains the same set of fields [http://foldoc.org/foldoc/foldoc.cgi?fields] where each field is a certain fixed width.
SQL	Originally SEQUEL [http://en.wikipedia.org/wiki/SQL#History] and still pronounced that way by many practitioners, SQL is the Standard Query Language; a unified language for creating queries that is accepted (with some variations) by all modern relational databases.
Table	A collection of records [http://foldoc.org/foldoc/foldoc.cgi?records] in a relational database [http://foldoc.org/foldoc/foldoc.cgi?relational+database] .

Chapter 8. Appendices

Appendix A: GNU GPL Version 3

The DQguru is distributed under the terms of the GNU General Public License, version 3 or later. Here is the text of that license:

GNU General Public License version 3

Version 3, 29 June 2007

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program—to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL , as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

“This License” refers to version 3 of the GNU General Public License.

“Copyright” also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

“The Program” refers to any copyrightable work licensed under this License. Each licensee is addressed as “you”. “Licensees” and “recipients” may be individuals or organizations.

To “modify” a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a “modified version” of the earlier work or a work “based on” the earlier work.

A “covered work” means either the unmodified Program or a work based on the Program.

To “propagate” a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To “convey” a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays “Appropriate Legal Notices” to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The “source code” for a work means the preferred form of the work for making modifications to it. “Object code” means any non-source form of a work.

A “Standard Interface” means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The “System Libraries” of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A “Major Component”, in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The “Corresponding Source” for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control

those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a. The work must carry prominent notices stating that you modified it, and giving a relevant date.

b. The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to “keep intact all notices”.

c. You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d. If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an “aggregate” if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation’s users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a. Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b. Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c. Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d. Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e. Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A “User Product” is either (1) a “consumer product”, which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, “normally used” refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

“Installation Information” for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

“Additional permissions” are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a. Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b. Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c. Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d. Limiting the use for publicity purposes of names of licensors or authors of the material; or

e. Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f. Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered “further restrictions” within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you

permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An “entity transaction” is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party’s predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A “contributor” is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor’s “contributor version”.

A contributor’s “essential patent claims” are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, “control” includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor’s essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a “patent license” is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To “grant” such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. “Knowingly relying” means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient’s use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the

covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is “discriminatory” if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others’ Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License “or any later version” applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy’s public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the “copyright” line and a pointer to where the full notice is found.

one line to give the program’s name and a brief idea of what it does.
Copyright (C) *year name of author*

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see http://www.gnu.org/licenses/ [http://www.gnu.org/licenses/] .

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

program Copyright (C) *year name of author* This program comes with ABSOLUTELY NO WARRANTY; for details type ‘ `show w` ’. This is free software, and you are welcome to redistribute it under certain conditions; type ‘ `show c` ’ for details.

The hypothetical commands ‘ `show w` ’ and ‘ `show c` ’ should show the appropriate parts of the General Public License. Of course, your program’s commands might be different; for a GUI interface, you would use an “about box”.

You should also get your employer (if you work as a programmer) or school, if any, to sign a “copyright disclaimer” for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL , see http://www.gnu.org/licenses/ [http://www.gnu.org/licenses/] .

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read http://www.gnu.org/philosophy/why-not-lgpl.html [http://www.gnu.org/philosophy/why-not-lgpl.html] .

Appendix B: Third Party Licenses

The FAMFAMFAM Silk Icon Set

The DQguru development team is grateful to Mark James for the beautiful Silk icon set which is used extensively throughout DQguru's user interface.

The Silk icons are freely available under the Creative Commons Attribution 2.5 License [http://creativecommons.org/licenses/by/2.5/] . If you want to use these icons in your own work (and why wouldn't you?), you can obtain the full set from the FAMFAMFAM web site [http://www.famfamfam.com/lab/icons/silk/] .

The Apache Software Foundation

The DQguru development team is grateful to the Apache Software Foundation and their contributors; their high-quality reusable Java libraries have been invaluable in the development of DQguru. The text of the Apache License follows, because we are redistributing several Apache libraries upon which DQguru depends.

The following license applies to these library jar files, which are distributed as part of the DQguru download:

- commons-beanutils.jar
- commons-digester.jar
- commons-logging.jar
- commons-beanutils-bean-collections.jar
- commons-beanutils-core.jar
- jakarta-regexp-1.2.jar
- commons-collections-3.1.jar
- commons-dbcp-1.2.1.jar

- commons-pool-1.3.jar
- commons-vfs-1.0.jar
- log4j.jar

Apache License Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions .

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License . Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License . Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution . You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- You must give any other recipients of the Work or Derivative Works a copy of this License; and
- You must cause any modified files to carry prominent notices stating that You changed the files; and
- You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions . Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks . This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty . Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without

limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability . In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability . While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0
(the "License"); you may not use this
file except in compliance with the License. You
may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to
in writing, software distributed under the
License is distributed on an "AS IS"
BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
KIND, either express or implied. See the License
for the specific language governing permissions
and limitations under the License.

JGoodies Karsten Lentzsch

The DQguru team is also grateful to JGoodies for their excellent forms layout manager for Swing. JGoodies forms is released under the BSD license, reproduced below.

The following license applies to these library jar files, which are distributed as part of the DQguru download:

- forms-1.1.0.jar

The BSD License for the JGoodies Forms

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of JGoodies Karsten Lentzsch nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

PostgreSQL JDBC Driver

The DQguru team would like to thank the PostgreSQL JDBC Driver team for their JDBC driver.

The following license applies to these library jar files, which are distributed as part of the DQguru download:

- postgresql_8.jar

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the PostgreSQL Global Development Group nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

iText

The DQguru team is also grateful to Bruno Lowagie and Paulo Soares for their excellent Java PDF library. iText is released under the MPL license, reproduced below.

The following license applies to these library jar files, which are distributed as part of the DQguru download:

- itext-1.4.8.jar

MOZILLA PUBLIC LICENSE Version 1.1

1. Definitions.

1.0.1. "Commercial Use" means distribution or otherwise making the Covered Code available to a third party.

1.1. "Contributor" means each entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Code, prior Modifications used by a Contributor, and the Modifications made by that particular Contributor.

1.3. "Covered Code" means the Original Code or Modifications or the combination of the Original Code and Modifications, in each case including portions thereof.

1.4. "Electronic Distribution Mechanism" means a mechanism generally accepted in the software development community for the electronic transfer of data.

1.5. "Executable" means Covered Code in any form other than Source Code.

1.6. "Initial Developer" means the individual or entity identified as the Initial Developer in the Source Code notice required by Exhibit A.

1.7. "Larger Work" means a work which combines Covered Code or portions thereof with code not governed by the terms of this License.

1.8. "License" means this document.

1.8.1. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means any addition to or deletion from the substance or structure of either the Original Code or any previous Modifications. When Covered Code is released as a series of files, a Modification is: A. Any addition to or deletion from the contents of a file containing Original Code or previous Modifications.

B. Any new file that contains any part of the Original Code or previous Modifications.

1.10. "Original Code" means Source Code of computer software code which is described in the Source Code notice required by Exhibit A as Original Code, and which, at the time of its release under this License is not already Covered Code governed by this License.

1.10.1. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.11. "Source Code" means the preferred form of the Covered Code for making modifications to it, including all modules it contains, plus any associated interface definition files, scripts used to control compilation and installation of an Executable, or source code differential comparisons against either the Original Code or another well known, available Covered Code of the Contributor's choice. The Source Code can be in a compressed or archival form, provided the appropriate decompression or de-archiving software is widely available for no charge.

1.12. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License or a future version of this License issued under Section 6.1. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. Source Code License.

2.1. The Initial Developer Grant. The Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license, subject to third party intellectual property claims: (a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer to use, reproduce, modify, display, perform, sublicense and distribute the Original Code (or portions thereof) with or without Modifications, and/or as part of a Larger Work; and

(b) under Patents Claims infringed by the making, using or selling of Original Code, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Code (or portions thereof).

(c) the licenses granted in this Section 2.1(a) and (b) are effective on the date Initial Developer first distributes Original Code under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: 1) for code that You delete from the Original Code; 2) separate from the Original Code; or 3) for infringements caused by: i) the modification of the Original Code or ii) the combination of the Original Code with other software or devices.

2.2. Contributor Grant. Subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor, to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof) either on an unmodified basis, with other Modifications, as Covered Code and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: 1) Modifications made by that Contributor (or portions thereof); and 2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) the licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first makes Commercial Use of the Covered Code.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: 1) for any code that Contributor has deleted from the Contributor Version; 2) separate from the Contributor Version; 3) for infringements caused by: i) third party modifications of Contributor Version or ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or 4) under Patent Claims infringed by Covered Code in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Application of License. The Modifications which You create or to which You contribute are governed by the terms of this License, including without limitation Section 2.2. The Source Code version of Covered Code may be distributed only under the terms of this License or a future version of this License released under Section 6.1, and You must include a copy of this License with every copy of the Source Code You distribute. You may not offer or impose any terms on any Source Code version that alters or restricts the applicable version of this License or the recipients' rights hereunder. However, You may include an additional document offering the additional rights described in Section 3.5.

3.2. Availability of Source Code. Any Modification which You create or to which You contribute must be made available in Source Code form under the terms of this License either on the same media as an Executable version or via an accepted Electronic Distribution Mechanism to anyone to whom you made an Executable version available; and if made available via Electronic Distribution Mechanism, must remain available for at least twelve (12) months after the date it initially became available, or at least six (6) months after a subsequent version of that particular Modification has been made available to such recipients. You are responsible for ensuring that the Source Code version remains available even if the Electronic Distribution Mechanism is maintained by a third party.

3.3. Description of Modifications. You must cause all Covered Code to which You contribute to contain a file documenting the changes You made to create that Covered Code and the date of any change. You must include a prominent statement that the Modification is derived, directly or indirectly, from Original Code provided by the Initial Developer and including the name of the Initial Developer in (a) the Source Code, and (b) in any notice in an Executable version or related documentation in which You describe the origin or ownership of the Covered Code.

3.4. Intellectual Property Matters (a) Third Party Claims. If Contributor has knowledge that a license under a third party's intellectual property rights is required to exercise the rights granted by such Contributor under Sections 2.1 or 2.2, Contributor must include a text file with the Source Code distribution titled "LEGAL" which describes the claim and the party making the claim in sufficient detail that a recipient will know whom to contact. If Contributor obtains such knowledge after the Modification is made available as described in Section 3.2, Contributor shall promptly modify the LEGAL file in all copies Contributor makes available thereafter and shall take other steps (such as notifying appropriate mailing lists or newsgroups) reasonably calculated to inform those who received the Covered Code that new knowledge has been obtained.

(b) Contributor APIs. If Contributor's Modifications include an application programming interface and Contributor has knowledge of patent licenses which are reasonably necessary to implement that API, Contributor must also include this information in the LEGAL file.

(c) Representations. Contributor represents that, except as disclosed pursuant to Section 3.4(a) above, Contributor believes that Contributor's Modifications are Contributor's original creation(s) and/or Contributor has sufficient rights to grant the rights conveyed by this License.

3.5. Required Notices. You must duplicate the notice in Exhibit A in each file of the Source Code. If it is not possible to put such notice in a particular Source Code file due to its structure, then You must include such notice in a location (such as a relevant directory) where a user would be likely to look for such a notice. If You created one or more Modification(s) You may add your name as a Contributor to the notice described in Exhibit A. You must also duplicate this License in any documentation for the Source Code where You

describe recipients' rights or ownership rights relating to Covered Code. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Code. However, You may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear than any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.6. Distribution of Executable Versions. You may distribute Covered Code in Executable form only if the requirements of Section 3.1-3.5 have been met for that Covered Code, and if You include a notice stating that the Source Code version of the Covered Code is available under the terms of this License, including a description of how and where You have fulfilled the obligations of Section 3.2. The notice must be conspicuously included in any notice in an Executable version, related documentation or collateral in which You describe recipients' rights relating to the Covered Code. You may distribute the Executable version of Covered Code or ownership rights under a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable version does not attempt to limit or alter the recipient's rights in the Source Code version from the rights set forth in this License. If You distribute the Executable version under a different license You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or any Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.7. Larger Works. You may create a Larger Work by combining Covered Code with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Code.

4. Inability to Comply Due to Statute or Regulation.

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Code due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be included in the LEGAL file described in Section 3.4 and must be included with all distributions of the Source Code. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Application of this License.

This License applies to code to which the Initial Developer has attached the notice in Exhibit A and to related Covered Code.

6. Versions of the License.

6.1. New Versions. Netscape Communications Corporation ("Netscape") may publish revised and/or new versions of the License from time to time. Each version will be given a distinguishing version number.

6.2. Effect of New Versions. Once Covered Code has been published under a particular version of the License, You may always continue to use it under the terms of that version. You may also choose to use such Covered Code under the terms of any subsequent version of the License published by Netscape. No one other than Netscape has the right to modify the terms applicable to Covered Code created under this License.

6.3. Derivative Works. If You create or use a modified version of this License (which you may only do in order to apply it to code which is not already Covered Code governed by this License), You must (a) rename Your license so that the phrases "Mozilla", "MOZILLAPL", "MOZPL", "Netscape", "MPL",

"NPL" or any confusingly similar phrase do not appear in your license (except to note that your license differs from this License) and (b) otherwise make it clear that Your version of the license contains terms which differ from the Mozilla Public License and Netscape Public License. (Filling in the name of the Initial Developer, Original Code or Contributor in the notice described in Exhibit A shall not of themselves be deemed to be modifications of this License.)

7. DISCLAIMER OF WARRANTY.

COVERED CODE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED CODE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGING. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED CODE IS WITH YOU. SHOULD ANY COVERED CODE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED CODE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

8. TERMINATION.

8.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. All sublicenses to the Covered Code which are properly granted shall survive any termination of this License. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

8.2. If You initiate litigation by asserting a patent infringement claim (excluding declatory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You file such action is referred to as "Participant") alleging that:

(a) such Participant's Contributor Version directly or indirectly infringes any patent, then any and all rights granted by such Participant to You under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively, unless if within 60 days after receipt of notice You either: (i) agree in writing to pay Participant a mutually agreeable reasonable royalty for Your past and future use of Modifications made by such Participant, or (ii) withdraw Your litigation claim with respect to the Contributor Version against such Participant. If within 60 days of notice, a reasonable royalty and payment arrangement are not mutually agreed upon in writing by the parties or the litigation claim is not withdrawn, the rights granted by Participant to You under Sections 2.1 and/or 2.2 automatically terminate at the expiration of the 60 day notice period specified above.

(b) any software, hardware, or device, other than such Participant's Contributor Version, directly or indirectly infringes any patent, then any rights granted to You by such Participant under Sections 2.1(b) and 2.2(b) are revoked effective as of the date You first made, used, sold, distributed, or had made, Modifications made by that Participant.

8.3. If You assert a patent infringement claim against Participant alleging that such Participant's Contributor Version directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

8.4. In the event of termination under Sections 8.1 or 8.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or any distributor hereunder prior to termination shall survive termination.

9. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED CODE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

10. U.S. GOVERNMENT END USERS.

The Covered Code is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" and "commercial computer software documentation," as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Code with only those rights set forth herein.

11. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by California law provisions (except to the extent applicable law, if any, provides otherwise), excluding its conflict-of-law provisions. With respect to disputes in which at least one party is a citizen of, or an entity chartered or registered to do business in the United States of America, any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California, with venue lying in Santa Clara County, California, with the losing party responsible for costs, including without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License.

12. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

13. MULTIPLE-LICENSED CODE.

Initial Developer may designate portions of the Covered Code as "Multiple-Licensed". "Multiple-Licensed" means that the Initial Developer permits you to utilize portions of the Covered Code under Your choice of the NPL or the alternative licenses, if any, specified by the Initial Developer in the file described in Exhibit A.

EXHIBIT A -Mozilla Public License.

``The contents of this file are subject to the Mozilla Public License Version 1.1 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.mozilla.org/MPL/

Software distributed under the License is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the License.

The Original Code is _____.

The Initial Developer of the Original Code is _____. Portions created by _____ are Copyright (C) _____ _____. All Rights Reserved.

Contributor(s): _____.

Alternatively, the contents of this file may be used under the terms of the _____ license (the "[___] License"), in which case the provisions of [_____] License are applicable instead of those above. If you wish to allow use of your version of this file only under the terms of the [____] License and not to allow others to use your version of this file under the MPL, indicate your decision by deleting the provisions above and replace them with the notice and other provisions required by the [___] License. If you do not delete the provisions above, a recipient may use your version of this file under either the MPL or the [___] License."

[NOTE: The text of this Exhibit A may differ slightly from the text of the notices in the Source Code files of the Original Code. You should use the text of this Exhibit A rather than the text found in the Original Code Source Code for Your Modifications.]

JFree

The DQguru team is also grateful to the JFree team for their top-notch charting library, which has a nice API as well as nice-looking output.

The following license applies to these library jar files, which are distributed as part of the DQguru download:

- jcommon-1.0.0.jar
- jfreechart-1.0.1.jar

GNU Lesser General Public License version 3

Version 3, 29 June 2007

Copyright © 2007 Free Software Foundation, Inc. http://fsf.org/

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, “this License” refers to version 3 of the GNU Lesser General Public License, and the “ GNU GPL ” refers to version 3 of the GNU General Public License.

“The Library” refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An “Application” is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A “Combined Work” is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the “Linked Version”.

The “Minimal Corresponding Source” for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The “Corresponding Application Code” for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL .

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL .

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a. under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b. under the GNU GPL , with none of the additional permissions of this License applicable to that copy.

3. Object Code Incorporating Material from Library Header Files.

The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a. Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b. Accompany the object code with a copy of the GNU GPL and this license document.

4. Combined Works.

You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a. Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b. Accompany the Combined Work with a copy of the GNU GPL and this license document.

c. For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d. Do one of the following:

1. Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

2. Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e. Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL , and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)

5. Combined Libraries.

You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a. Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b. Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

Darwin Systems

Thanks to Ian Darwin of Darwin Systems for his many contributions to DQguru.

The following license applies to:

- darwinsys.jar

JUnit

The DQguru team would also like to extend our sincere thanks to the JUnit.org team. JUnit forms an invaluable part of our development process, but it is not redistributed as part of the DQguru download so its license is not reproduced here.

If you develop software, you should become test infected too! Learn about JUnit at http://www.junit.org/ [http://www.junit.org/] .

The Eclipse Foundation

The DQguru was primarily developed and tested using the Eclipse [http://www.eclipse.org/] Java Development Tools, one of the more productive Java environments around.

Sun Microsystems

Last but not least, many thanks to Sun Microsystems [http://java.sun.com/] and their various Java development teams for creating, extending, bugfixing, documenting, and supporting the Java platform over the past *N* years!

We redistribute the JavaHelp runtime library with DQguru. Although the JavaHelp website says that the system will be redistributable royalty-free, it does not actually link to the specific license terms. If someone can point us to the license text for JavaHelp redistributions, we would be grateful!

The portion of JavaHelp that we redistribute is in the following JAR file:

- jhall.jar

We also redistribute the JavaMail library and the JavaBeans Activation Framework, on which it depends

The following JAR files are covered by the CDDL, reproduced below:

- mail.jar
- activation.jar

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable means the Covered Software in any form other than Source Code.

1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License means this document.

1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications means the Source Code and Executable form of any of the following: A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications; B. Any new file that contains any part of the Original Software or previous Modification; or C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof);

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License;

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code. Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications. The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices. You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms. You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions. You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipients rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works. You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions. Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions. You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions. When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY. COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGING. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE

PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY. UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTYS NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS. The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS. This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdictions conflict-of-law provisions. Any litigation relating to this

License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS. As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.