

# ISQL-VIEWER 3.0.0

## *Installation and User Guide*



Mark A. Kobold

2007

<b>Introduction</b>	<b>2</b>
<b>Installation</b>	<b>3</b>
Requirements	3
Installing iSQL-Viewer	3
<b>Getting Started</b>	<b>3</b>
<b>User-Interface Basics</b>	<b>4</b>
<i>Tabbed Views</i>	
<i>Keyboard Shortcuts</i>	
<i>Drag-n-Drop Operations</i>	
<i>File System Information</i>	
<b>Creating a Database Connection</b>	<b>5</b>
<b>Browsing the Database Schema</b>	<b>7</b>

<b>Schema Browsing Tips</b>	<b>7</b>
Variable Substitutions	8
<b>Variable Substitution Tips</b>	<b>8</b>
Making Database Queries	9
<b>SQL Command Editor Tips</b>	<b>10</b>
Viewing Query Results (Result-Sets)	10
<b>Result-set Viewer Tips</b>	<b>11</b>
Using SQL Bookmarks	11
<b>SQL Bookmark Tips</b>	<b>12</b>
Using SQL History	13
<b>SQL History Tips</b>	<b>13</b>
<b>Frequently Asked Questions</b>	<b>14</b>
<b>Credits</b>	<b>15</b>

## **Introduction**

Thank you for supporting open-source software by choosing to use the iSQL-Viewer program. This document will cover all the basic functionality that is needed to use iSQL-Viewer on a daily basis. This document will also cover some of the not so obvious things you can do to make working with iSQL-Viewer that much more enjoyable.

iSQL-Viewer is an open-source (free as in beer and speech) SQL tool written in Java to provide an uninhibited environment (no hibernate or spring) where JDBC drivers can function. iSQL-Viewer strives to provide a set of tools that would complement any JDBC driver, even those drivers that have their own specific tools.

The iSQL-Viewer program has a variety of uses and is targeted primarily towards developers and possibly DBA who need to work in one or more database platforms on a daily basis. iSQL-Viewer can also provide a way for JDBC driver developers to test their driver within.

## Installation

### Requirements

The iSQL-Viewer program only requires that the Java Runtime Environment 1.5.0\_06+ (JRE) to be installed prior to installing the application using the installer provided by iSQL-Viewer. Failing to have Java properly installed will likely cause the application to not run properly.

Proceed to this address for information on how to install and acquire the correct JRE for your environment. <http://java.sun.com/javase/downloads/index.jsp>

### Installing iSQL-Viewer

iSQL-Viewer can be installed using the installer programs downloadable from the Sourceforge.net file servers. Select the right installer for your operating system, and proceed with the installer prompts.

All application installers can be found at the following Internet address: [http://sourceforge.net/project/showfiles.php?group\\_id=11233](http://sourceforge.net/project/showfiles.php?group_id=11233)



**Note:** The iSQL-Viewer installers are not very complex at this time. At the very most any decision beyond 'next' will be to choose whether or not to install the source code of the application.

## Getting Started

Once iSQL-Viewer has been successfully installed. It is now time to start up iSQL-Viewer for the first time. iSQL-Viewer can be started from the "Start Menu" (Windows, KDE, Gnome) or the Applications directory (Mac OS X). Once the application has been started you should see an empty screen.

If iSQL-Viewer is being started for the first time, there will most likely be a notable pause for the first run of the application. This pause is normal, if this pause lasts longer than a few minutes then there might be a problem with the application. See the troubleshooting section for more information.



**Note:** If data from iSQL-Viewer 2.1.x is detected the program will make an attempt to import the existing data (history, bookmarks, and services) into the new system structures.

## User-Interface Basics

Before proceeding any further, this section will cover some basics of how to understand iSQL-Viewer's user-interface (UI).

### Tabbed Views

All views that are contained within a tab will have two distinct properties. All views that have a beveled X  icon allow that tab to be removed from view. Also, tabs can be double-clicked to perform a *zoom* effect. This zooming effect causes the selected tab to take up all of the available screen real-estate so that the contents of that tab can be viewed more easily. To revert back to the normal view, simply double click the tab once more to *un-zoom* the view. Users of the Eclipse Java IDE will be very familiar with this functionality.

### Keyboard Shortcuts

All keyboard short-cuts (*accelerators*) are converted wherever possible to fit in with the target platform. For example, most accelerators are updated to use the  key on Mac OS X, and on Windows and Linux platforms the primary accelerator key is the Control key.



**Tip:** Mac OS X users will need to make sure that full-keyboard access is enabled in the system preferences in order use any keyboard short-cut that requires the use of any of the standard function keys.

### Drag-n-Drop Operations

All drag and drop (DnD) operations are implemented where ever it is appropriate. Users are encouraged to explore the iSQL-Viewer application by dragging items to and fro. Users should also be the local operating system operations such as move, copy, and link. iSQL-Viewer will act differently (accordingly) with these different Drag-n-Drop actions.

## File System Information

iSQL-Viewer will use a directory known as **isql.home**. This directory is the base directory where iSQL-Viewer will store its application, service, and log files. This directory is based on the users home directory. Windows this is C:\Documents and Settings\<\$user\_name>. UNIX users home directory is most commonly /home/\$user\_name. On Mac OS X the isql.home directory is located in /Users/\$user\_name/Library/Application Support/iSQL-Viewer.

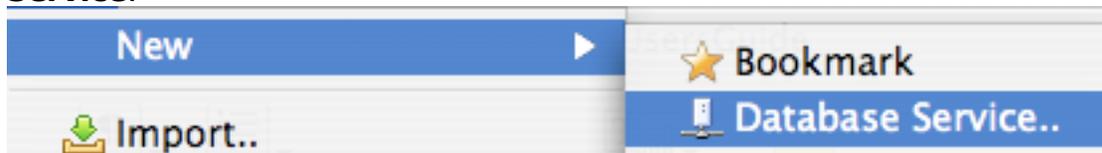
## Creating a Database Connection

Now that iSQL-Viewer is started, it will now be required to defined a service in order to use iSQL-Viewer against a database. Follow these simple steps in order to get the connection defined correctly.

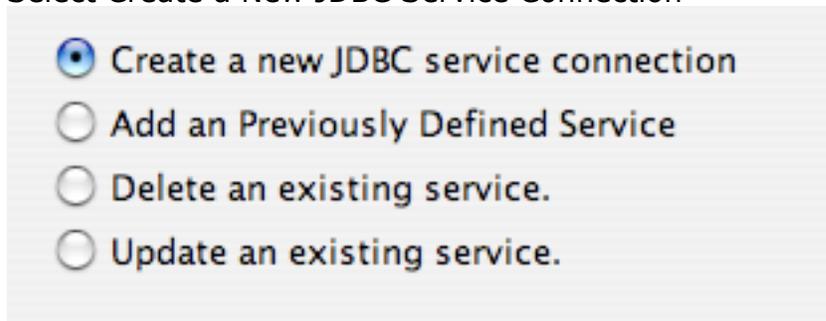


**Note:** Creating an iSQL-Viewer service was probably the biggest complaint I was never told about. However, from my own research this process has been hopefully simplified a great deal from the 2.1.8 series of iSQL-Viewer.

1. Invoke the service wizard from the tools menu or **File >> New >> Database Service**.



2. Select Create a New JDBC Service Connection



3. Fill out the following basic connection information including URL, and the JDBC driver class-name. You may need to refer to the JDBC driver documentation for information on the URL and class-name to use.
4. Configure the class-path for this service by adding the Java archive (JAR), ZIP File, or local file directory that contains the driver class specified from the previous step.
5. **Optional:** Configure Optional iSQL-Viewer preferences for this service. All options specified at this point are specific to the iSQL-Viewer application.

6. **Optional:** Configure Driver Properties for this connection. This step may require you to refer to the driver documentation as far as what to set. This step allows platform specific option; for example using iSQL-Viewer to connect to PostgreSQL via SSL. If new properties are to be provided based on those configured you can always refresh to get options as defined by the JDBC API.
7. Confirm your settings and choose the compatibility that will be required for this service.
8. Check the **Connect to Service** if you would like to connect to this service once the service has been created.



**Tip:** If it is not planned to use this service within 2.1.8 it is strongly recommended to save it in version 3.0+ of the service file format.

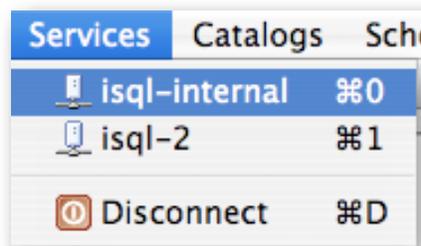
## Password Security Information

iSQL-Viewer password security in versions prior to 3.0 was next to non-existent. However in this version more standards have been adopted and the passwords that are saved within the service definition files are far more secure than before; and here is how.

When the iSQL-Viewer application starts up it attempts to detect a private key file in the iSQL-Viewer home directory. If that file exists that then becomes the system or application key for encrypting and decrypting information. If the file doesn't exist a new one is generated and saved for future use. This key is used to encrypt using standard and strong encryption algorithms. If the generated key is deleted any passwords will not be recoverable.

## Connecting to the Database

Now that you have created your service, it is now time to actually connect to that service. All services are accessed via the **Service** menu, as shown below.



*Service Menu*

All services will be shown with their given name and possibly and keyboard short-cut. Only the first 10 services will be assigned a keyboard short-cut of ⌘ (Mac) or Control (Windows, Linux) Key plus a number zero through nine. Once you select your service from the menu or use the keyboard shortcut. iSQL-Viewer will proceed to connect using the settings configured for that service. You may be prompted to

for a user-name and password if your service was configured to use an authentication prompt.

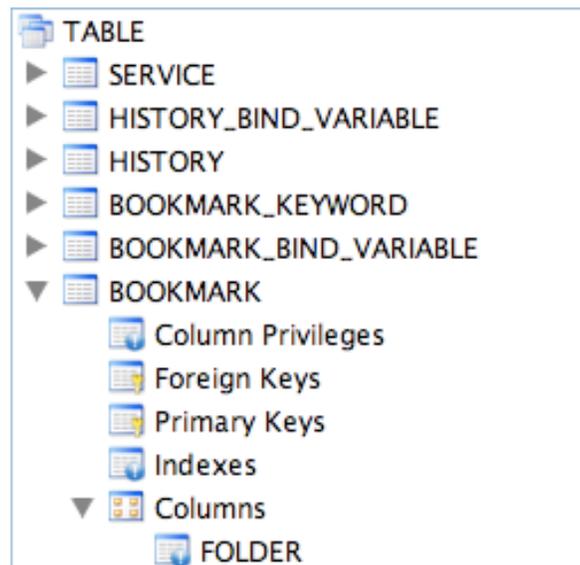
If the service fails to connect to the database, the error will be reflected in the lower console screen in **red** text. Please verify your URL and settings along with the actual error message.



**Tip:** All errors are logged to the local disk for every database connection. So if for some reason erratic behavior is occurring there is a log file in the **logs** directory in your **isql.home** location

## Browsing the Database Schema

Once you have successfully connected to your iSQL-Viewer service. One of the common things that people like to do with iSQL-Viewer is browse the database schema. iSQL-Viewer has a JDBC schema viewer. This view reflects all of the schema information that can be only be queried using the JDBC API. As can be seen below the schema browser will show all table types; which includes views, synonyms, and normal tables.



*Schema Viewer*

The JDBC Schema Viewer can be access by using the  $\uparrow \text{⌘} S$  for Mac OS X, or Ctrl-Shift-S for Windows and Linux environments. This view can also be opened from the following menu sequence; **Window >> Show View >> JDBC Schema**.

## Schema Browsing Tips

★ Along with tables, many other things can be viewed in the database schema tree. The following items can also be seen through the JDBC explorer.

- Stored Procedures
- Stored Procedure Columns
- Data Types
- User Data Types
- Primary Keys
- Foreign Keys
- Indexes

★ If a warning sign appears on any of the database objects. That means there was a SQL error that occurred while trying to query the information. Most common causes for this icon appearing is due to privilege restrictions.

★ Most items from the database schema view can be dragged over in to the main query result area. If a **move** (*shift key down*) is done then iSQL-Viewer will render the JDBC schema information for that node. Otherwise if a *copy* (*control key down*) is done iSQL-Viewer will make an attempt to create an SQL query based on the item being dropped.

★ Sometimes the schema browser will not show the expected information make sure you have the correct schema and catalog from the main menu selected.

## Variable Substitutions

It is now appropriate to discuss a general functionality, which is referred to as variables or string substitution within the rest of this document. These substitution allows for queries to be written a bit more generically where a particular value is only slightly different; for example an id, or other grouping value. Variable substitutions are exposed in the typical Java fashion like **`${variable name}`**. When appropriate you will be prompted to enter text to replace this placeholder with whatever value you choose to enter.

### Variable Substitution Tips

★ Variable substitutions can be used in both bookmarks and any query you wish to execute with iSQL-Viewer.

★ iSQL-Viewer will remember your previous inputs for each variable saving you time when entering the same substitutions over and over again. So it advantageous to use the same variable names whenever appropriate.

★ If the text is a variable declaration and should be taken literally simply select the **Do Not Substitute**. This will cause no modification to the text as it stands in the query.

★ String substitutions can also be applied to service parameter including parts the JDBC URL and user-name fields.

## Making Database Queries

The most common task that will be carried out by iSQL-Viewer is executing database queries and viewing the results. The SQL command editor supports syntax highlighting and various keyboard short-cuts for executing any query. The SQL command editor can be seen below

```
SELECT
  COUNT(*),
  trunc(EVENT_TIME) AS "EVENT_DATE"
FROM
  HISTORY;
GROUP BY
  TRUNC(EVENT_TIME);
```

*SQL Command Editor*

Once a query needs to be executed there are a few ways that the SQL commands can be invoked. Here are the few ways that a query can be executed.

- I. Simply press ⌘ Return or Ctrl-Enter to execute all of the text currently in the editor.
- II. Select specific text (like a sub-query) and press ⌘ Return or Ctrl-Enter to execute only the selected text.
- III. Simply move your cursor somewhere within the query you would like to execute and press ⇧F9 or Shift-F9 to execute a context-sensitive query. Such that if multiple queries are present in the editor and separated by a blank line the this will execute. As seen below using this short-cut will only execute the middle query.

```
SELECT COUNT(*) FROM HISTORY;

SELECT COUNT(*) FROM BOOKMARK;

SELECT COUNT(*) FROM SERVICE;
```

### Context Sensitive Query Example

Once the command is passed to the database for execution a new tab will be created where the table data will be rendered (*if available*). Once the table data is rendered then the query is finished. Information regarding the process of the query will show in the console and tab views.

## SQL Command Editor Tips

- ★ Text from just about any application can be dragged into or out of this editor.
- ★ Multiple SQL commands can be executed at once by separating each command with an unquoted semi-colon (;).
- ★ Tables object that are in the currently selected schema and catalog will only be highlighted and colored; this is mainly an effort to be 'nice' to the database as some metadata queries can be quite taxing.
- ★ Bookmarks and Historical items can be dropped into the editor from their respective views to insert or replace the query text of those items.
- ★ When dragging and dropping text into this editor if a copy action is used it will insert the text at the cursor position where the text is dropped. Otherwise if a move action is used it will replace the text with what ever text is being dropped.
- ★ If there are some basic query errors like unclosed quotations or variables names; then that text will show up as **red** in the editor.
- ★ Sometimes the color highlighting will get confused; Often just adding and removing a space at the beginning of the text will correct the coloring.
- ★ The SQL command editor will remember the text that existed when the application shuts down; and this text will be restored back into the editor when iSQL-Viewer is restarted.

## Viewing Query Results (*Result-Sets*)

For most SQL commands executed there will be a result set that is returned. Otherwise there will be a **rows affected** value. For result-sets iSQL-Viewer will show results in tabular form as can be seen below.

HISTORY_ID	SERVICE_ID	QUERY_TIME	TRANSACTION_ID	COMMAND_TYPE	COMMAND_TEXT
154	2	2007-05-08 21:59:27.692	-1	QUERY	SELECT * FROM BOOKMARK
155	2	2007-05-09 17:13:27.161	-1	QUERY	SELECT * FROM BOOKMARK
156	2	2007-05-09 17:13:52.925	-1	QUERY	SELECT * FROM BOOKMARK
157	2	2007-05-09 17:14:26.946	1178748866946	BATCH_QUERY	
158	2	2007-05-09 17:14:26.946	1178748866946	QUERY	SELECT * FROM BOOKMARK
159	2	2007-05-09 17:14:51.996	1178748866946	QUERY	SELECT * FROM HISTORY
160	2	2007-05-09 17:17:22.16	1178749042160	BATCH_QUERY	
161	2	2007-05-09 17:17:22.16	1178749042160	QUERY	SELECT * FROM BOOKMARK
162	2	2007-05-09 17:17:37.244	1178749042160	QUERY	SELECT * FROM HISTORY
163	2	2007-05-09 17:22:29.938	-1	QUERY	SELECT * FROM BOOKMARK
164	2	2007-05-09 17:22:34.871	-1	QUERY	SELECT * FROM BOOKMARK
165	2	2007-05-16 18:44:01.023	-1	OUERY	SELECT * FROM BOOKMARK

*iSQL-Viewer Result-set view example.*

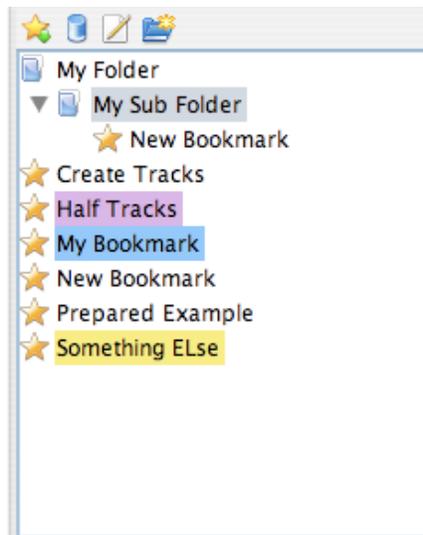
The result-set viewer provides a simple manner to browse the results of your query as the database provides them. This view provides client side searching and sorting. So that the data can be organized more appropriately for the task at hand.

## Result-set Viewer Tips

- ★ Text used to search the result set can be in the form of regular expression for complex pattern searching.
- ★ By selecting any table cells and using the copy command (⌘C/Ctrl-C) will copy the table text as tab-delimited data for pasting.
- ★ Selected table cells can also be dragged into other applications as tab-delimited text. This allows a very smooth transition of data between iSQL-Viewer and spreadsheet applications such as Microsoft Excel.
- ★ Double-clicking the column separators will cause all the columns to automatically adjust their size based on all of their column data
- ★ Columns can be manually re-arranged by dragging them as appropriate for the situation.

## Using SQL Bookmarks

SQL Bookmarks are much like any other kind of bookmark; except that these are SQL queries that one does not wish to type over and over again. Bookmarks can be organized into folders and sub-folders, and even highlighted with certain colors (*ala Mac OS X Finder*) to assist with organizing and accessing your bookmarks. Bookmarks are often dragged into the SQL Command editor and executed.



*iSQL-Viewer Bookmark Viewer*

The SQL Bookmark Viewer can be accessed by using the  $\text{⌘}B$  for Mac OS X, or Ctrl-Shift-S for Windows and Linux environments. This view can also be opened from the following menu sequence; **Window >> Show View >> Bookmarks**.

## SQL Bookmark Tips

- ★ Empty folder structures are not saved, so if a complex folder structure is built and no bookmarks are actually added that folder structure will not be saved. Only the deepest folder that contains a bookmark will be saved to disk.
- ★ Bookmarks can be dragged into the query result area along with the text editor. If the bookmark is dragged into the results area that will cause the bookmark to be automatically executed.
- ★ Bookmarks and folders can be organized by dragging and dropping them within the main bookmark view. Bookmarks can be copied or moved by using the appropriate action. Folders can only be moved at this time.
- ★ It is recommended to use variable declarations within bookmarks in order to make them more re-usable so that the same bookmark does not have to be defined for minor differences within the query itself.
- ★ Folders can only be renamed by editing 'in-place' within the view. Such click the folder and briefly wait before clicking the folder again and it will allow you to re-name the folder. This also works for quickly renaming bookmark.
- ★ Bookmark color assignments are made in the information tab of the bookmark editor.

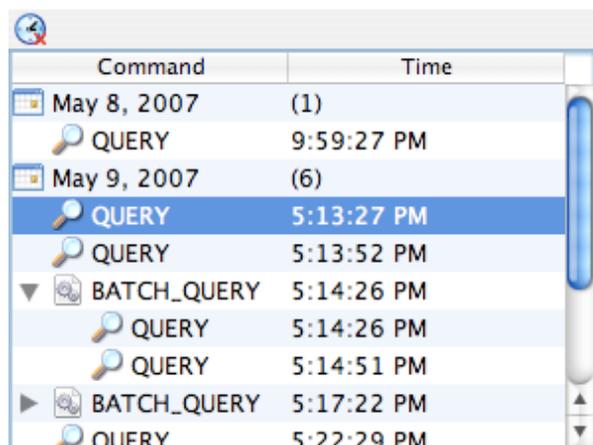
★ Any plain text can be dropped into the main bookmark view to create a new bookmark with the dropped text.

★ Sometimes the color highlighting will get confused; Often just adding and removing a space at the beginning of the text will correct the coloring.

★ The SQL command editor will remember the text that existed when the application shuts down; and this text will be restored back into the editor when iSQL-Viewer is restarted.

## Using SQL History

For every successful command executed by iSQL-Viewer, those commands will get logged into a history for future viewing and retrieval. iSQL-Viewer will show each command based on the type of command executed (*Insert, Batch, Update, Delete, and Select*). The SQL History very simple, and will primarily be used by dragging history items



Command	Time
May 8, 2007 (1)	
QUERY	9:59:27 PM
May 9, 2007 (6)	
QUERY	5:13:27 PM
QUERY	5:13:52 PM
BATCH_QUERY	5:14:26 PM
QUERY	5:14:26 PM
QUERY	5:14:51 PM
BATCH_QUERY	5:17:22 PM
QUERY	5:22:29 PM

*iSQL-Viewer Command History Viewer*

The Command History Viewer can be access by using the  $\uparrow\text{⌘}H$  for Mac OS X, or  $\text{Ctrl-Shift-H}$  for Windows and Linux environments. This view can also be opened from the following menu sequence; **Window >> Show View >> Query History**.

## SQL History Tips

★ History Commands can be dragged into the bookmark view and dropped into making a new bookmark with the command text of the history item.

★ Batch queries can be dragged all at once or only a single sub-query of that batch command.

# Frequently Asked Questions

Here are some frequently asked questions. Most of these don't really have much to do about using iSQL-Viewer program however it sets the story straight.

## 1. What does iSQL-Viewer stand for?

iSQL-Viewer stands for Independent SQL Viewer. My goal of this application is to create nice a way to look at data that is usually pretty boring. However, the independence stems both from database and operating system requirements.

## 2. Why do you work on the iSQL-Viewer project?

Well I primarily use iSQL-Viewer to hone a different skill set than that of which I use for my job. I primarily do a lot of back-end work, no UI or anything. iSQL-Viewer is a very rewarding project for me personally to make something that actually works and I hope that it can stand as a good example of good open-source software in such a tough niche category of software.

## 3. What do you plan on getting out iSQL-Viewer?

Mainly prestige, money can never buy it. It would be great if I could get a well paying gig to write developer tools like iSQL-Viewer. However, time will tell.

## 4. Will iSQL-Viewer always remain free (speech and beer)?

As long as this project is under my belt, most certainly yes.

## 5. Does iSQL-Viewer ride the iPod, iTunes, iEverything bandwagon?

Nope, iSQL-Viewer was started in mid 1999. This predates a lot of the I applications and products produced by Apple as well as others. So technically when I started this project there was no bandwagon of the sort to jump on.

## 6. Is iSQL-Viewer related to isql from Sybase or Microsoft?

No, iSQL-Viewer is almost always referred to by its full name, and should always be as this project is not endorsed nor is it affiliated with Sybase or Microsoft. Though iSQL-Viewer does work with their products. However the short name for iSQL-Viewer is isql; merely a convenience then if anything.

## 6. Is iSQL-Viewer related to ISQL\*PLUS?

No, iSQL-Viewer is affiliated with ISQL\*PLUS nor is iSQL-Viewer endorsed by the makers of that product.

## 7. Why was there a large gap time for development between 2.1.8 and 3.0?

This not an easy question to answer. However this question has many answers but it boils down to two key points. 1. personal irresponsibility with dealing with my own problems, mainly unhealthy compulsions to do other things then to

deal with my issues. 2. Simply burnt out; from what seemed as a lack of community support and other factors along those lines i figured i needed a break anyways.

## Credits

Here is where I would like to give credit where credit is due for all those people and corporations that have provided the support for this project when and where they could.

- Sourceforge.net (<http://www.sourceforge.net>) -- Well without these guys i would have to spend a lot of my own money hosting my project. Many many thanks
- Anitra Pavka (<http://www.anitrapavka.com>) -- Personal friend and supporter of this project since its early days. She was responsible for a lot of the web footwork in the beginning.
- Bernd Stolle -- German Localizer for the 2.1.8 series.
- Parallels -- Makers of virtualization software for Mac OS X. Though they have not directly contributed to iSQL-Viewer. Their virtualization software has saved me tons of time testing in different environments easily with out having a bunch of physical machines
- ej-technologies (<http://www.ej-technologies.com>) -- Company that graciously provided me an open source license of the installer programs used for windows and linux systems. I will also say they have a very nice profiler as well.
- Cenqua -- (<http://www.cenqua.com/clover/>) -- Company that has provided an open source license to use the clover product in creating lean code through testing and figuring out where i might have missed a situation; or in most cases where I have over analyzed a particular function.