Genesis Software Corp. - GS Log Telemetry Toolkit

EZ Log - Viewer

Feature Guide

Contents

Introduction

Most text log files are easily read by the myriad of text file readers available. Database table logs are easily accessible as well. So why would you need a Log Viewer tool?

The EZ Log Viewer offers in a single software package, the ability to view a text file and a database recordset, side by side at the same time. In addition to that, a few utility functions to help when working with result sets and log files. I use this one tool to debug and monitor multiple applications from one normal PC Workstation. With or without the internet! I also use this tool to control the telemetry output level my applications produce in real time.

While the EZ Log Viewer can be used at the enterprise level, the intent is for the EZ Log Viewer is to be a personal tool. Even with Splunk[*] as one of the data analysis tools I use, the EZ Log Viewer is the tool I use **most** to find and debug software issues quickly. *(Splunk is an amazing enterprise tool I am using in addition to this tool at the time I am writing this doc in 2021).*

Over the years, I have added cool features I needed but did not document them well. I feel this product is ready to share with my professional associates. So… during this re-write of the document my goal is to familiarize you with the features I use most frequently. I hope to provide a video or two demonstrating a feature when I can in the Google Group associated with this product as well as answer questions.
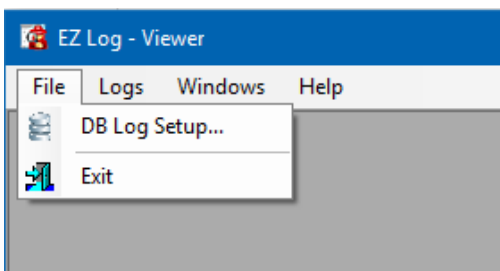
Environment

The EZ Log Viewer is a Microsoft Windows desktop app. Latest compile was with .Net Frameworks 4.7.2

The Main Screen

The main screen is designed so you can simultaneously examine multiple logs.
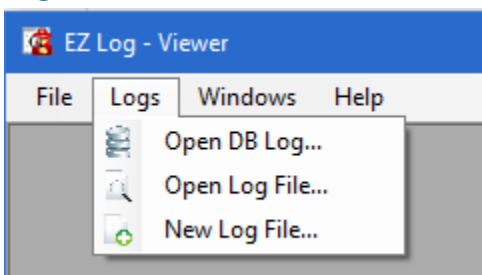
Menus:

File Menu

The file menu is where you perform your Database Connection Library setup (described later) or exit the app.

Logs Menu

The Log Menu is where you open one or more log windows.

You can optionally create a New Text Log file to store text copied from a larger pre-existing log file or simply to use the text search features of the Text File log window.

Database Logs

Logging to a database provides the most flexibility for an organization to perform analysis of the log by multiple people simultaneously as well as from remote locations.

Database Connection Setup

DB Log List

The diagram below shows a list of database logs I am working with at the time. Each log has its own connection properties associated with it. At the bottom of the diagram, a sample Connection Dialog is shown.
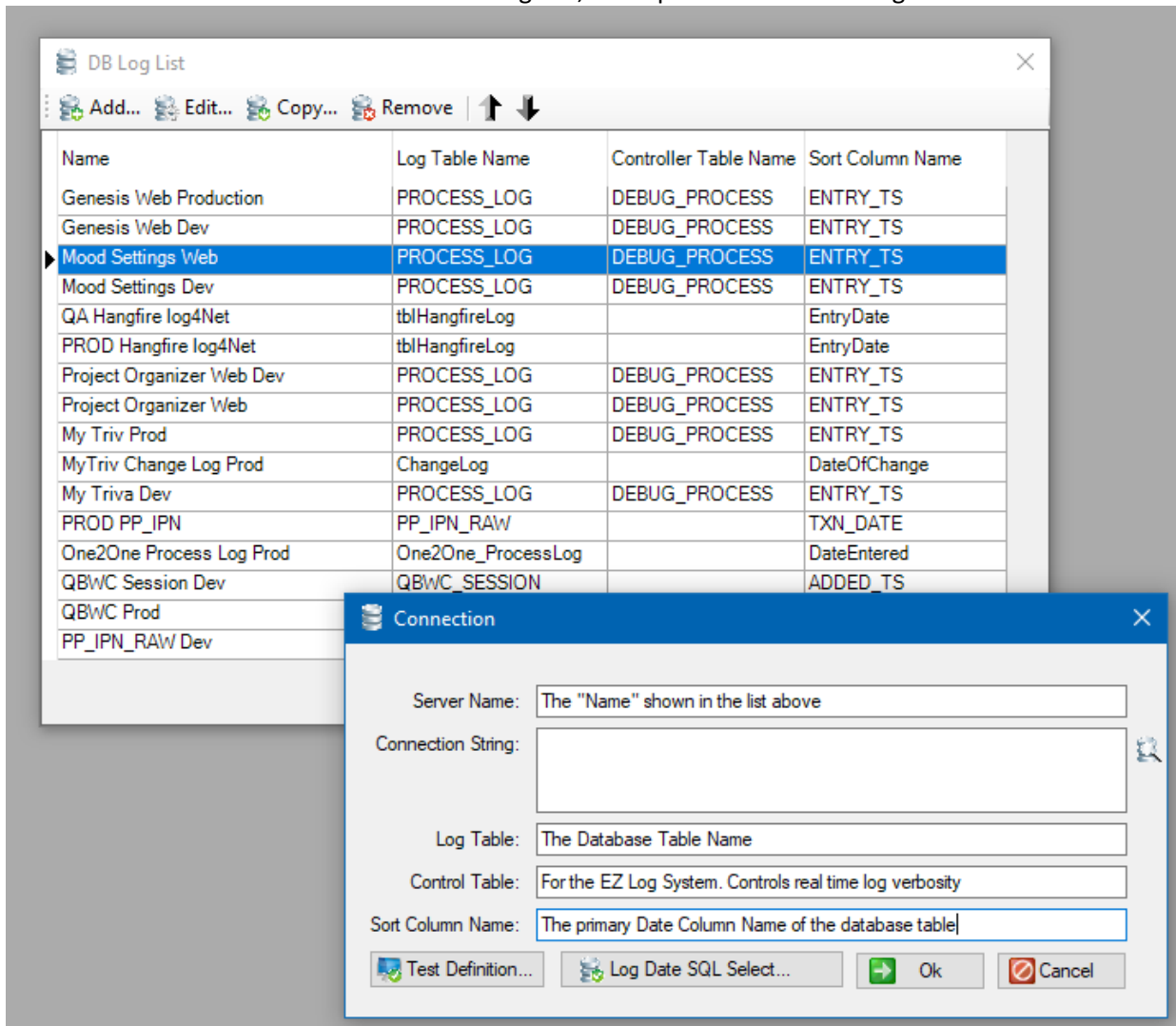
The description for the Connection Dialog continues on the next page.

Connection Definition Dialog

A single definition describes a connection. Let's go over each attribute and button in this window.
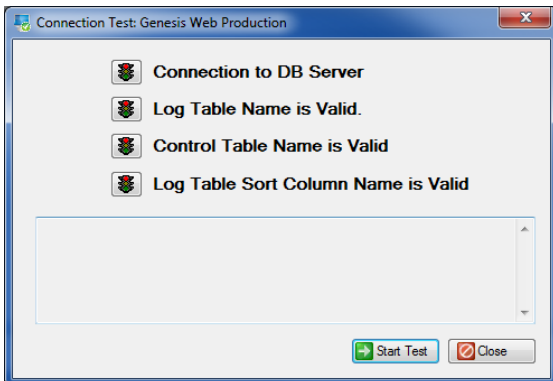
Attributes

Server Name:	This is a name you use to recognize a pre-defined connection in the DB Log List.
Connection String:	This input is the connection string to use when connecting to the database. To the right of the connection string input is an image that when clicked will bring up my Data Connection String builder dialog. *(GSDataUI Available separately for you to use in your apps)*
Log Table:	Input for the name of the Table that holds the data you're interested in viewing for this connection.
Control Table:	**This is an optional property directly associated with the EZ Log Telemetry toolkit.** If you are not using the **EZ Log Writer - GSLog** assembly to write your DB Logs, you can leave this input blank. The Control table is a very cool thing that is not so obvious. I will try to put together a tutorial on it. (See the Writer portion documentation of the EZ Log toolkit for details.)
Sort Column Name:	You can leave this input blank and the result sets will NOT be sorted by default. Most software process logs have a date time column associated with the data. Put the Name of the column you want the data result set to sort on by default.

Utility Buttons on the Database Connection Setup dialog.

I added these so I can test the connection and move on with confidence.

Test Definition Button

This button opens a window that validates the attributes you put into the Connection definition. (Seen on the left)

Each individual test is performed and reported in the textbox that appears above the Start Test button.

The Test Definition window is a convenience to make sure your Log definition is correct and that other items that are used by the DB Log Window are valid.

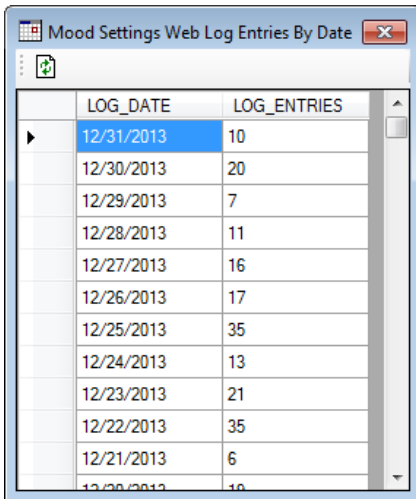When all tests are successful the traffic lights turn green.

Log Date SQL Select Button
This button is not obvious at all and needs explaining. Something perhaps you may want to skip over now if you are new to the EZ Log Viewer.

Here is the story behind the feature.

Many of my small business clients use rented database resources that have size and database feature limitations. Sometimes Log Tables can hold more information than you want to hold on to. In the DB Log Window. I wanted a quick way to see how many log entries are stored for a particular date so I could delete unwanted records from the database and thus stay within the limitations of my client's rental arrangement.

The DB Log window has a "Log Entries by Date" button. *(See the Log Entry Count button description later in this document for more details.)*

When the button is clicked, a popup window appears showing the results of a query that would break down the count of entries in the Log table by the day in a pop-up window. *(Shown to the left)*

The SQL query to support that pop-up window is associated with the Connection definition because the SQL Query uses the name of the Table and the Sort Column.

The window seen below shows the pop-up window that appears when you click the Log Date SQL Select button in the Connection Dialog.
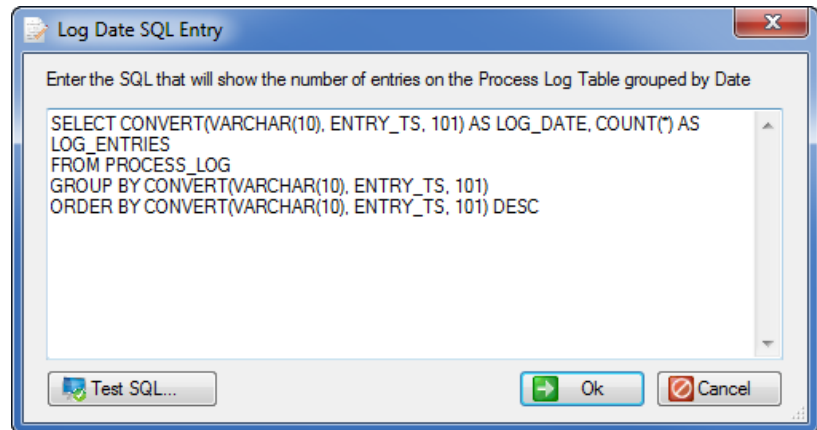
The default SQL Query, as shown, works for log table structure of the **EZ Log Writer** assembly. Adapt the query SQL to suit your table and column names. If you do not use the feature, don't worry about it. (Just have to smile 😊)

On the next page we will examine the DB Log window that appears when you open a "Connection" from the DB Log List.

The DB Log Window

A DB Log Window is associated with a Connection from the DB Log List you learned about in the previous section. Shown below is a screen shot of the DB Log Window after a query has been made.

At the top of the window is the tool bar providing access to functionality. Under the tool bar is a textbox where you can enter and issue WHERE Clause SQL Stub Statements. Below the textbox is a Data Grid where the results of the log table select query is displayed.

You refine the resultset by entering a where clause in the text box and clicking the "Select Rows" button on the tool bar.

The next section goes over the tool bar functions for database logs

The Tool Bar Functions

The image above shows the tool bar. Some of the functions need no more explaining than the tooltip provides. I'll provide details for the less obvious ones in the order I think you'll want to learn them.

Select Rows

This function works in conjunction with the textbox below the Tool Bar. You put the SQL WHERE clause in the text box and click the Select Rows button to submit the SQL statement to the DB. A SQL Select statement is generated using the values of the associated Connection properties we learned about earlier.
The SQL Template is *(SELECT * FROM <table name> … ORDER BY <sort column>).*
 The results of the "Select" are displayed on the grid.

Limit Resultset Length

I simply inject a TOP nnn declarative to the SQL Select statement. Sends less data over the wire from the DB.
Enter the number of rows you want and press the Select Rows button.

Query Timeout Setting

Opens a dialog where you can set the Timeout period of the SELECT Statement database query.

View Cell Contents

A very convenient way to review all of the data in the grid cell without manually cutting and pasting the cell data into a separate text editor.

Simply click on the cell you want to see, then click this button to expose the text in a popup window.

Consider the diagram below. Specifically, the contents of the grid cell currently highlighted.

	MEDIA PLAYER	20130401074835.820000	70.193.193.214:MedaiPlayer.aspx:Load:	IsPostBack=False
▶	MEDIA PLAYER	20130401074835.820000	70.193.193.214:MedaiPlayer.aspx:Load:	sUserAgent=Mozilla/5.0 (Linux; Android 4.1.2; DROID RAZR Build/9.8.2O-72_VZ\
	MEDIA PLAYER	20130401074835.823000	70.193.193.214:MedaiPlayer.aspx:Load:	iApple=False
	MEDIA PLAYER	20130401074835.823000	70.193.193.214:MedaiPlayer.aspx:Load:	Getting Publications

Clicking the 🔍 button on the toolbar results in the following popup which displays the total contents of the cell

Cell Contents

sUserAgent=Mozilla/5.0 (Linux; Android 4.1.2; DROID RAZR Build/9.8.2O-72_VZW-16) AppleWebKit/535.19 (KHTML, like Gecko) Chrome/18.0.1025.166 Mobile Safari/535.19

The popup window can be re-sized to convenient dimensions.

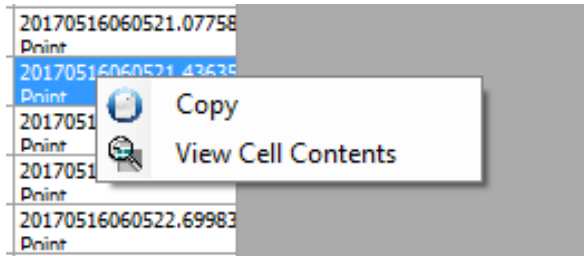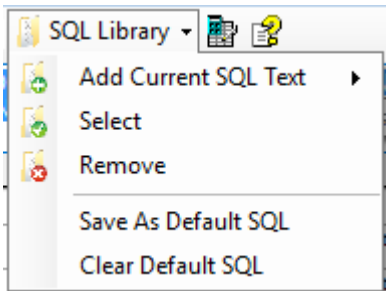Alternatively, mouse or stylus users can right click the desired grid cell and the context menu to the right will appear.

I occasionally copy the cell contents into the clip board and open a new Log File Viewer to search deeper.

SQL Where Clause Library

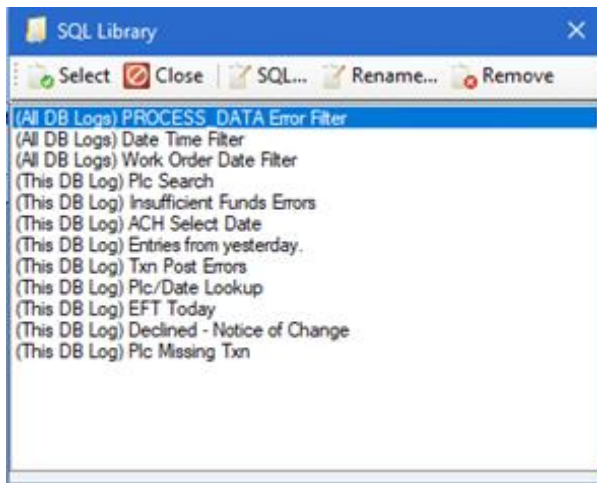I think this is one of the coolest features of this application. I use it all the time.

The SQL Library is used to store frequent SQL Statement Where Clause stubs you enter in the text area for selecting records for later recall.

The "Add Current SQL Text" menu functions takes the contents of the "WHERE Clause" textbox and stores it for future recall.

When you Add a SQL stub, you can add it to all of the DB Logs defined in the DB Log List or for the current DB Log you are looking at.

In addition, you can set up the SQL stub to be the default SQL the next time you open that DB log. This is useful if the log table you are examining has a large number of rows, you review the log frequently, and you typically only want to see a specific subset of the log table.

The "Select" or "Remove" menu options shown above, open the SQL Library manager pop up shown on the left.

This pop-up window is where you access functions to edit the SQL clause, rename the snippet or remove it.

To use, highlight the desired entry and click or tap the "Select" option.

Next, I will talk about the feature the EZ Log Toolkit was designed for. Real time changes to the level of telemetry reported by the application employing the EZ Log writer assembly.

This feature is available ONLY when using the **EZ Log Writer** to write your database log entries.

When writing your database log entries, you set the debug level requirement for your software to output the log entry in real time. You use the "Control" table to set the "DEBUG_LEVEL" for one or more Processes in one or more applications.

PROCESS_NAME	DEBUG_LEVEL
MAIN_PAGE	0
PUB DETAILS	0
MEDIA SERVICES	0
RATING	0
SCHEDULING	9
PERFORMANCE_REQUEST	9
ADMIN	9
PAYPAL	9
PHOTOS	9
MEDIA PLAYER	0
SCHEDULE	0
VENUES	0
BIO	0
USR_SYSTEM	0
WebRequests	0
VenueServices	0

17 Rows — DEBUG_PROCESS

PROCESS_NAME	DEBUG_LEVEL
MAIN_PAGE	3
PUB DETAILS	0
MEDIA SERVICES	0
RATING	0
SCHEDULING	9
PERFORMANCE_REQUEST	9
ADMIN	9
PAYPAL	9
PHOTOS	9
MEDIA PLAYER	0
SCHEDULE	0
VENUES	0
BIO	0
USR_SYSTEM	0
WebRequests	0
VenueServices	0

17 Rows — DEBUG_PROCESS

You change the DEBUG_LEVEL value associated with a process by changing the number in the display grid and pressing Enter.

The EZ Log Writer adds the PROCESS_NAME automatically. (See the EZ Log Writer Programmer's Guide for details.)

In the diagram to the left, the processes with a DEBUG_LEVEL of zero (0) effectively have logging turned off.

I have used the control table to focus on portions of the application. For example, if I only wanted to see the networking activity of my application, I would have the application write to the log with a specific "Process Name" when performing networking activity. In addition to the process name, I can use the Debug Level to signal how much information is to be sent to the database. The control table software mechanism for lack of a better term, can be employed to suit whatever strategy you wish to use. For me, at the time that I designed the telemetry system, it was simply to focus on a specific area of the application and limit the number of records being sent to the database table.

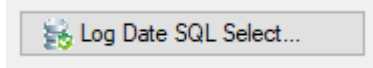The next section explains the Log Entry Count by Date feature

Log Entry Count by Date

Many of my logs are on rented space and I need to make sure the size of the log is limited. I added this feature just because I needed to see where and how much to delete from the log tables…

LOG_DATE	LOG_ENTRIES
12/31/2013	10
12/30/2013	20
12/29/2013	7
12/28/2013	11
12/27/2013	16
12/26/2013	17
12/25/2013	35
12/24/2013	13
12/23/2013	21
12/22/2013	35
12/21/2013	6

The pop-up window shown to the left, displays the results of the SQL Query entered in the Connection Definition Dialog.
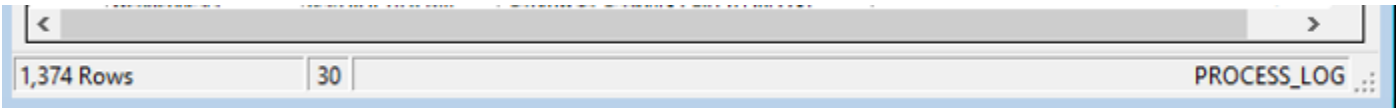
Log Date SQL Select…

Seemed like a good way to find records to clean up when I wrote it.

Something like this can show heavy dates for web usage as well.

I bet if you alter the SQL defined in the Connection Definition you could show something that may be more meaningful for you. However, be mindful that the popup grid may not co-operate if you go way outside of the intentions for this popup.

Status Bar Indicators

1,374 Rows	30	PROCESS_LOG

Nothing too complicated but it does have to be said: Row Count | Select Command Timeout | Log Table Name

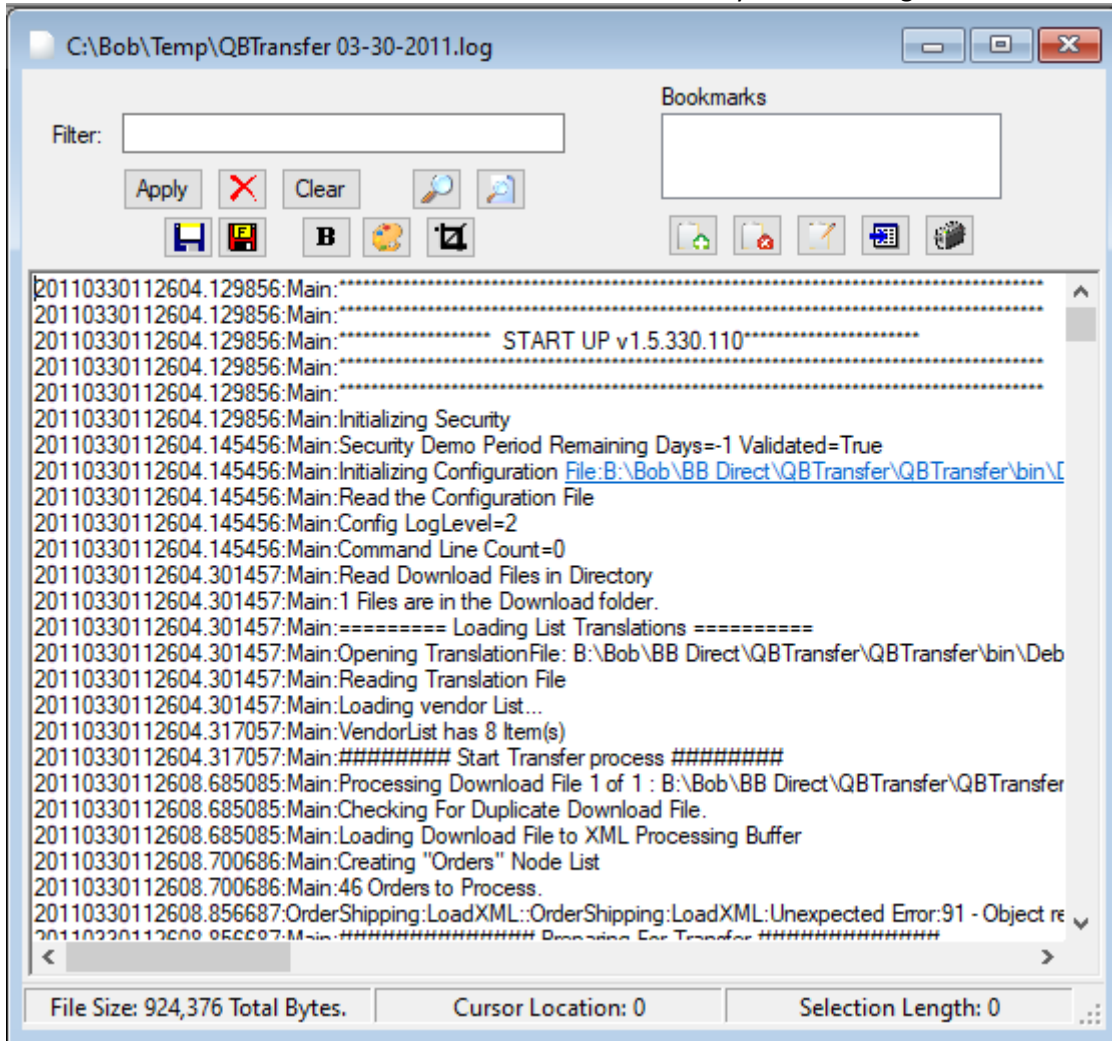The next section covers the text file viewer.

Text File Logs

The next most popular logging by software is writing to a text file.

The Text Log Window

This image below shows the Text File viewer. This is a window used to analyze text file logs.
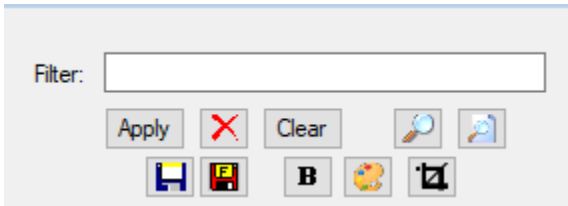

There are 3 sections to the Log File window I'll discuss in the order of importance.

Log Text Display Section

This section is the majority of the display area. The log text is editable in this window so you can add notes to assist in your analysis as well as use cut and paste to manipulate the contents. This is the main portion of the window.

Filtering Section

Basically, you key in your search term and click Apply. Then each line in the file is scanned for the search term and included in the displayed result.

The "X" button in the filtering section acts as a logical NOT for the search.
For example, if you put the text "Error" in the Search Text textbox and clicked the "X" button, the result would return all lines in the log file that do NOT contain the text "Error".

Find

This is your typical text search function. The cursor position is moved to the location (or the next location in the log) where the text pattern you search for is located. Like some other text editors, pressing the F3 key will search for the next occurrence of the search text and move the display to the next occurrence when found.

Replace

The replace function is useful for when I want to remove or replace text in a single bulk operation. This is a fast and convenient way to remove information rom the log you are not interested in at the moment.
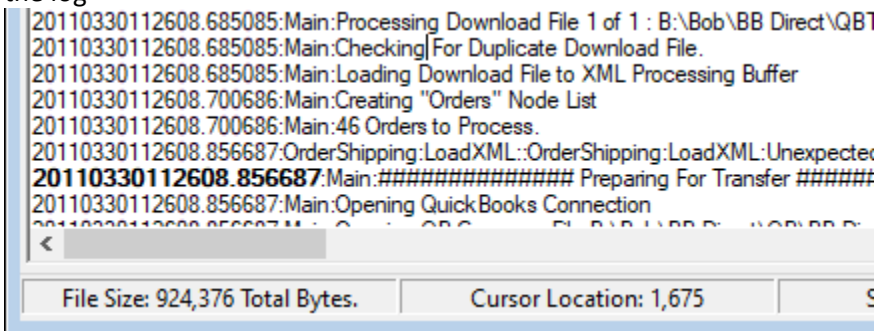
Crop

Sometimes when I import data from another application like splunk, there is simply too much data in the text. This gives me a convenient way to simply remove a block of text from the whole log.

Bold

Clicking this button will "Bold" the "Selected Text" in the log display. I use this type of markup to easily locate sections of the log
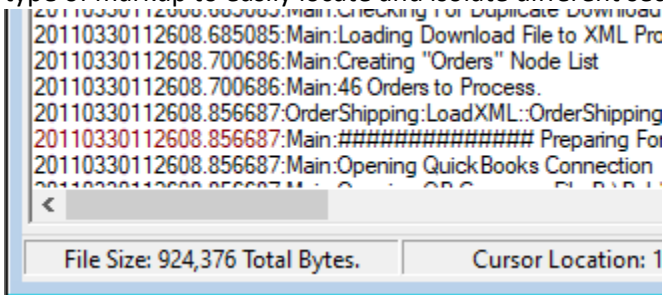
Color

Clicking this button will open a color picker where you set the "Color" of the "Selected Text" in the log display. I use this type of markup to easily locate and isolate different sections of the log.

Save

After you have marked up the log or added notes to the text display area you can save your text changes by clicking the "Save" button. Any formatting, (Bold, Color, Font Adjustments, …), is not saved.
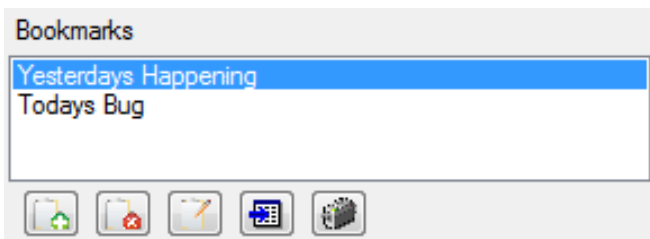
Save Formatted

The text displayed file is saved in an RTF format so as to keep any font formatting you applied to the log text.

Bookmarks

Sometimes logs are huge (20 Meg++) holding entries over many days. Supporting an app that had this type log, a user called notifying me of an error and told me that "the same thing worked yesterday". So… I would find the time/location in the log where the situation worked yesterday for that user and also where it did not work today. Thus "Bookmarks" was added to the Text Log viewer.
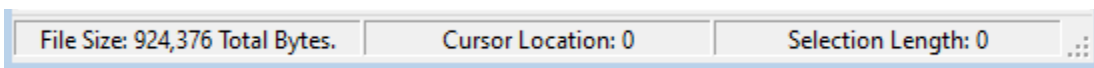

The bookmarks area of the text log window in the diagram to the left. I have added 2 bookmarks to go along with the story I used to present the bookmark feature.

Use buttons below the bookmark list to work with

The first three buttons are used to Add, Remove, and Rename individual bookmarks. Clicking the fourth button will move the cursor position in the text to the location of the highlighted bookmark. The fifth button is used to "Select the Text" between two bookmarks in preparation to copy or cut the text to the Window's clipboard. This was useful when distance (in text) between the bookmarks is large and manually selecting the text is cumbersome.

The Status Bar Section

Anymore need be said?