

# 3D Log Viewer

*Based on version 1.9.6*

## Reference Manual

*Revision 1.5*



**Novell.**

# 3D Log Viewer

## Table of Contents

Introduction.....	3
Getting Started.....	5
Using Bookmarks.....	6
Creating a New Profile.....	7
Creating a New Rule.....	8
Locating text in the log viewer which matches rules .....	9
Locating text in the Log Viewer, without using the rule list.....	10
Writing Log Viewer results to file.....	11
Changing Rule Levels.....	12
Reordering Rules.....	13
Refining your searches.....	14
Creating Rules Which Work.....	15
Regular Expression Syntax.....	16
Installation.....	18
Using on Linux.....	18

# Introduction

The viewer provides the ability to structure text based on text matching relationships. The viewer will work best with structured files, in which each text line is surrounded by lines which add context to the line being viewed. These file types will almost always be chronological and single thread based. Where more than one thread of activity can occur in parallel, then context of a line of text could be lost.

Examples of structured files are Novell Identify Manager / DirXML driver traces, NetWare Abend logs, NetWare system or console logs and DSBrowse high value count output. There are many, many other logs files which would be suite for viewing using this utility.

An example of a log file which is not suited to any viewer is DSTRACE.LOG when using multiple eDirectory trace filters. (IDM/DirXML output is chronological and single threaded.)

*Aside:* The file DSTRACE.LOG trace contains multi-threaded activity, with no thread number. Often a line is surrounded by text which has no bearing on the line being viewed. To view a file like DSTARCE.LOG, the lines of text would first need grouped on thread number, since there aren't any, it's not realy a useful log. If this file contained IDM/DirXML output the log would be difficult to make sense of the output.

To get thread numbers use SET DSTRACE or iMontor (HTML). The developers denied the request to add thread numbers to DSTRACE.LOG, even though the filters have an option for this.

To use the 3D Log Viewer a rule in the form of a Regular Expression (Regex) is created. This is a simple task as pure text alpha-numeric is a valid Regex. Things get more interesting when using special characters like `[]\.^*()`. The use these characters as literal text they need to be escaped, using a backslash `\`.

Example:

Errors reported (1)

Would be represented in Regex as:

Errors reported `\(1\)`

This is all a lot simpler using the viewer as special characters are automatically escaped for you.

Rules are associated with Profiles, to groups them. With a Regex rule configured, read in a text file which has this text in the file. When the line of text is read in it is formatted based on the rule configuration of colour, Level (indentation) and bold display.

**Important:** The power of this viewer lays in the fact that all text which does **not** match a rule, is automatically displayed at a lower level (as a subordinate) of the **previous rule** which **did** match.

This makes the viewer useful as only the text we know or care about has a rule defined. All other text fits around the framework of these rules.

If a rule is created for display at Level 1 and no Level 0 rule has matched, then the text is still displayed in the viewer, but the orphaned text:

- Will be displayed at Level 0
- Will be pre-pended by the message "Parent Missing:"

The end result of matching rules which relate to the start or end of a block of text is that the text below the matched rule can be hidden until required.

No text is actually removed from the file (unless expressly configured) it is available below

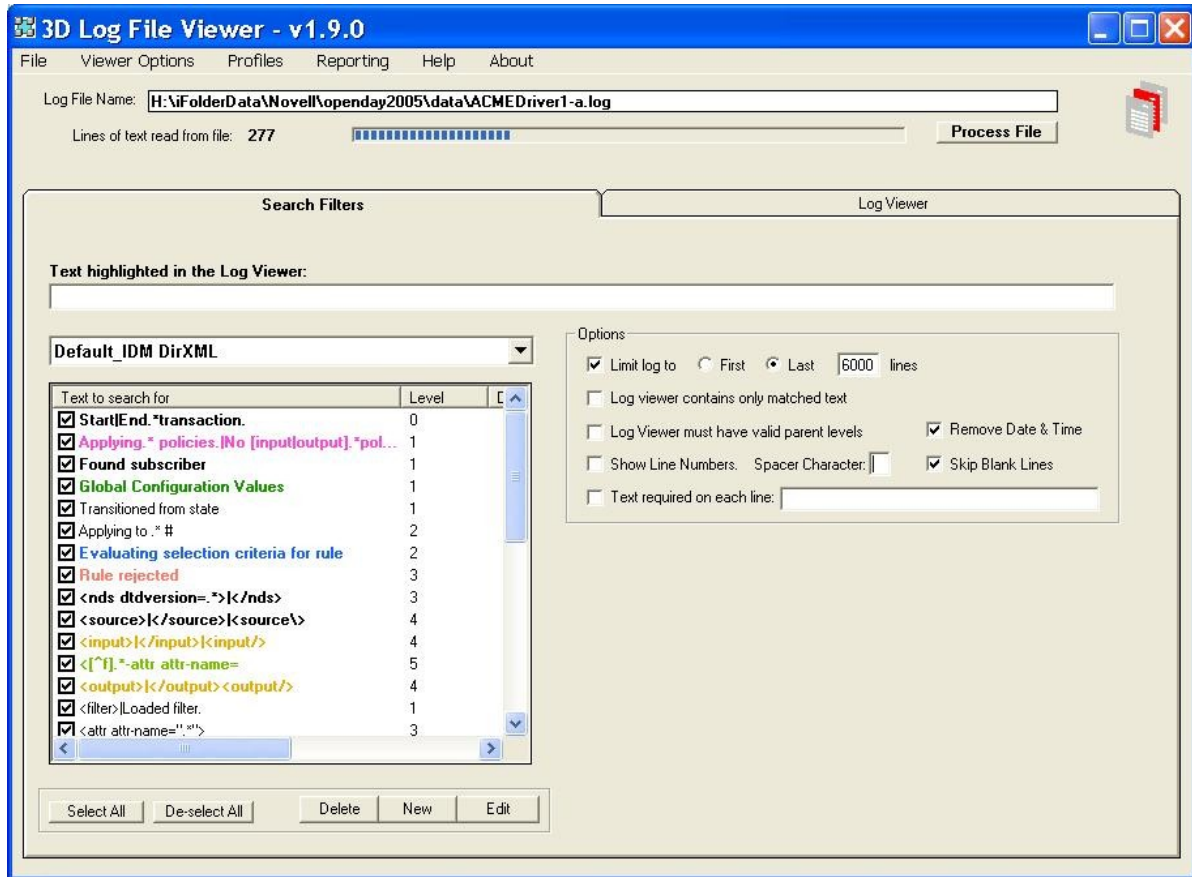
expandable text in a tree structure view.

The default profile is a **blank name**. The Profile Options set when this profile is selected are applied to application at start up.

**Important:** Beside this reference guide, the utility Help menu has an option for enabling Tool Tips. Allow the mouse to hover over an option and a short description is displayed. This feature is **off** by **default**. It must be manually enabled each time you need it.

# Getting Started

There are a number of default rule profiles installed along with the utility, this list will be expanded as the requirement for new log rules arises. A few sample logs are provided. For the purposes of this explanation we'll use an IDM log.



## Processing a file:

1. Select an IDM/DirXML text log file:
  - **Menu "File" | "Open Log File"**
  - Or by using drag and drop onto the "Log File Name" field
  - Or by using drag and drop onto the log viewer tree itself
2. From the drop-down list **select a the profile** "Default IDM DirXML"
3. Click **"Process File"**
4. When the file has completed loading the Log Viewer itself will be displayed.
5. Use the **horizontal scrollbar** to adjust the viewer to the left, if required
6. Try using the Menu: "Viewer Options" | "Expand First Level"
7. When a line of text is highlighted in the viewer, it is also copied to the "Text highlighted in the Log Viewer" field, seen above. This is used later to create Regex rules.

# Using Bookmarks

The bookmark feature provides a list of text lines in the log viewer for quick recall. The list is sorted by line number.

## The add lines to the bookmark list:

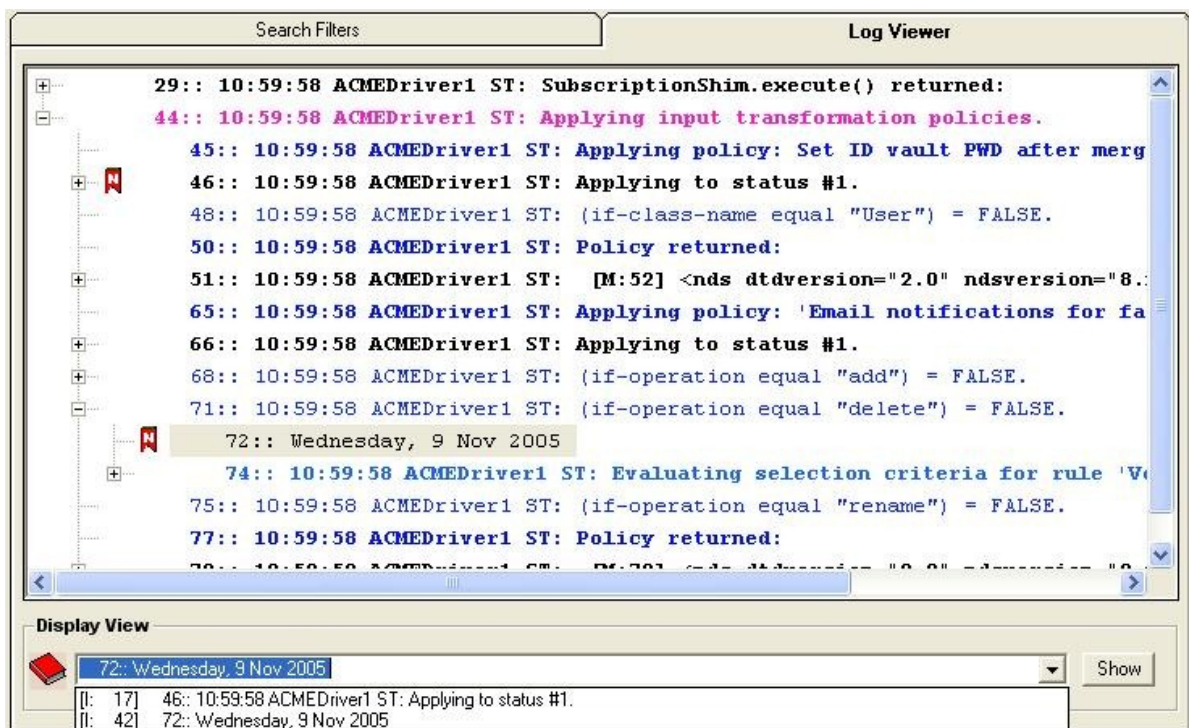
- Select a line in the Log Viewer.
- Double-click the book image located in the “Display View” or “Search View” frame.
- A bookmark image will be displayed next to the line in the Log Viewer.

## To remove a line from the bookmark list:

- The Log Viewer must be in “Display View” mode.
- Use the drop-down list located next to the book image.
- Select a line from the drop-down list, the one which should be removed.
- Double-click the book image located in the “Display View” frame.

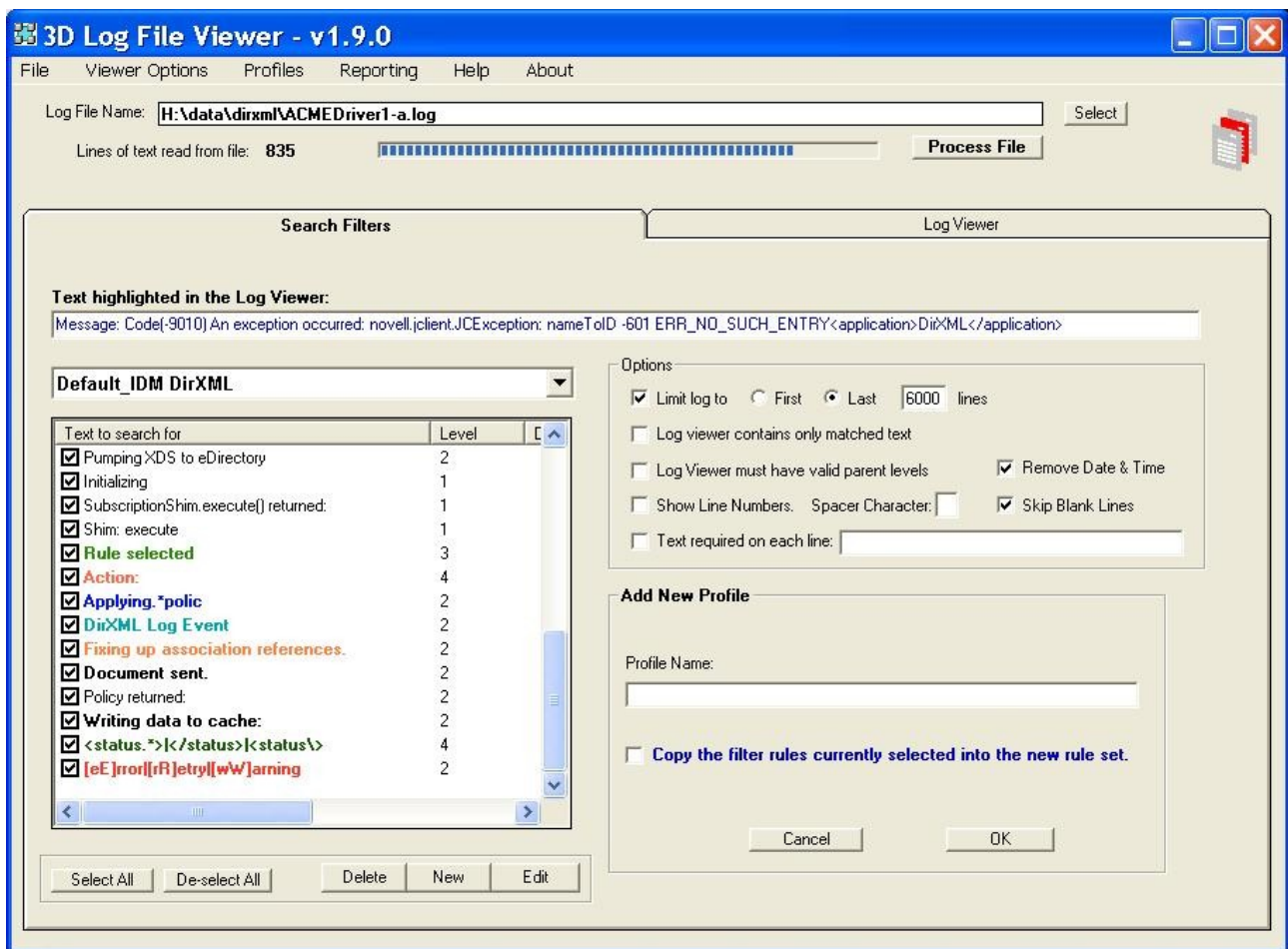
## To recall a bookmarked line:

- The Log Viewer must be in “Display View” mode.
- Use the drop-down list located next to the book image.
- Select a line from the drop-down list.
- The bookmarked line will be displayed in the Log Viewer.



# Creating a New Profile

1. Select the menu option "Profiles", then "Add New".
2. Choose a name for your profile. Duplicate names are detected.  
The profile name is actually a file name. The rules are located in the application installation directory, below a sub-directory called "Rules".  
The profile name may therefore not contain characters which are illegal as a file name. The Profile name will **automatically be corrected** to make it a valid file name.
3. Before clicking OK, decide which rules you would like to copy from an existing Profile.  
Only one profile may be used. (Perhaps a future version will provide an option to select rules from multiple profiles.)
4. Check the box "Copy the filter rules currently selected into the new rule set." to copy **highlighted** rules to this new profile. Multiple rules may be selected.



## Creating a New Rule

Before getting serious about creating rules, consider this. The result from these rules should be an organised hierarchy in terms of text viewed in a tree structure. To achieve this it's easier if you start with the high level text, then work your way to the lower levels. Or start at the lower levels and work up. Randomly selecting text will result in a tree view which obviously highlights the text you wanted, but can make the subordinate text appear out of context.

**Important:** Rules are evaluated from the top of the list to the bottom. As soon as a rule is matched on a line of text, **no other rules will be evaluated for that line**. You need to ensure that lines of text that are similar are carefully matched using a reliable Regex pattern.

1. Select an existing Profile name from the drop-down menu.
2. Click the "New" button located below the rules.
3. If there is text displayed in the "Text highlighted in the Log Viewer", it will be copied into "Text Rule: (Regular Expression)" text field.
4. Decide at which level the text should be displayed.

If you choose Level 2 and there is no Level 1 text matched previously, a message "Parent Missing:" will be displayed in the viewer. Either move this text up a level or ensure text at Level 1 is matched before the Level 2 text is matched.

5. Choose a colour and Bold settings.
6. Merge next line: When this text rule is matched, merge the following line with this one.
7. The Description is optional. (A future version may allow for this to be written to the Log Viewer, with the matched next displayed as a subordinate to this description)
8. Remember to use the menu "File", then "SAVE" before testing your rules against a file. If the program crashes your rules will be gone if you don't explicitly save them.

**Text highlighted in the Log Viewer:**

102: Message: Code(-9010) An exception occurred: novell.jclient.JCEException: nameToID -601 ERR\_NO\_SUCH\_ENTRY<application>DirXM

Default\_IDM2

**Edit Rule**

Level: 2 Description:

**Text Rule: (Regular Expression)**

[eE]rror[rR]etry[wW]arning[[F]atal

Bold 2701823 Colour  Merge Next Line

Test Regex Total Occurrences: - Cancel OK

Select All De-select All Delete New Edit

# Locating text in the log viewer which matches rules

The rules not only serve to structure the Log Viewer content, they are also a shortcut for locating text in the viewer. The entire viewer tree could be very large, simply scrolling through it will take too long.

1. Double-click the rule you want to locate.
2. This will locate all instances of the rule in the viewer and display references to them in a search view.
3. Each instance of the matched rule will be expanded in the tree view.
4. When you double-click a reference in the search view that text is displayed in the tree view.
5. Note that the "Search View" text field contains the rule you selected from the rules list.

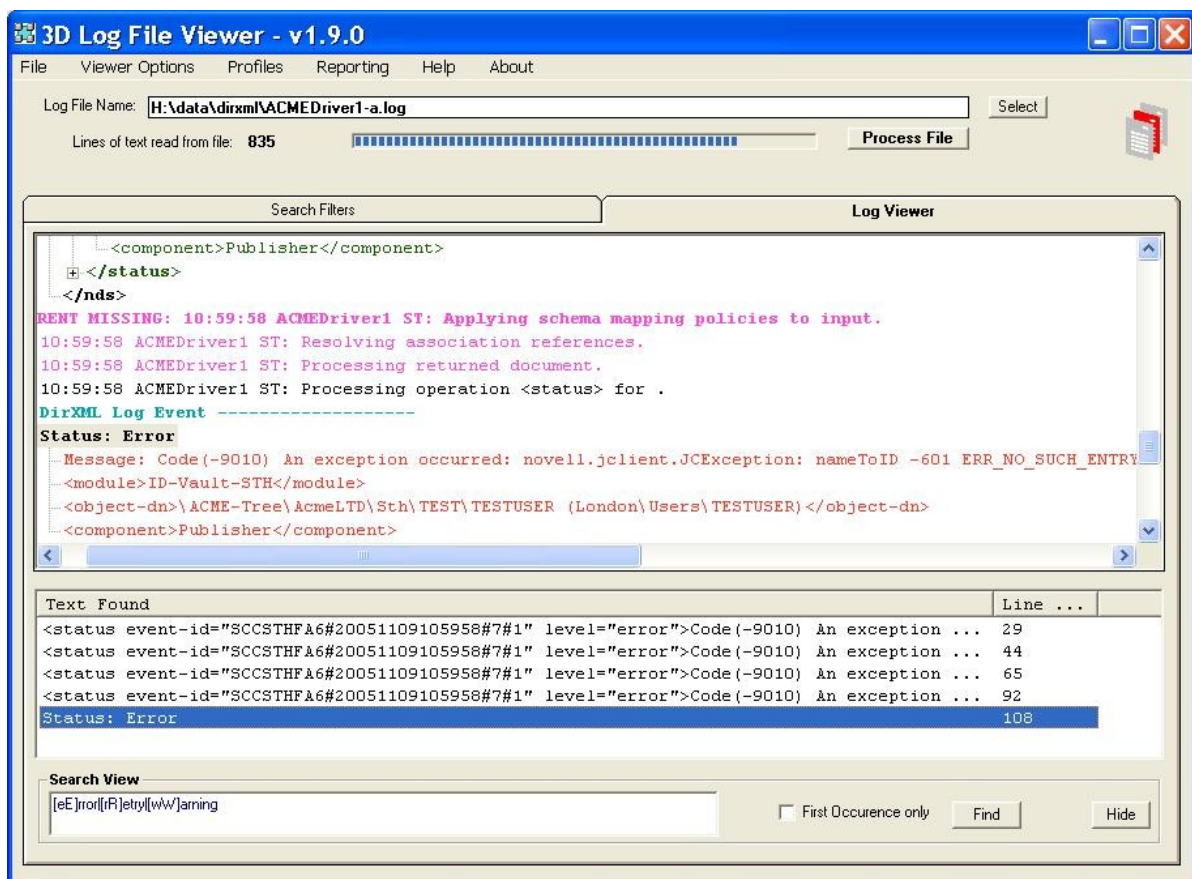
The screenshot shows the 3D Log File Viewer - v1.9.0 interface. The Log File Name is H:\data\dirxml\ACMEDriver1-a.log, and 835 lines of text have been read. The Search Filters panel shows two rules: <status>|</status>|<status> (4 matches) and [eE]rror|[R]etry|[wW]arning (2 matches). The Log Viewer shows a DirXML Log Event with a Status: Error. The Text Found table lists four instances of the error message, with the first instance on line 108 highlighted. The Search View panel shows the rule [eE]rror|[R]etry|[wW]arning.

Text Found	Line ...
<status event-id="SCCSTHFA6#20051109105958#7#1" level="error">Code (-9010) An exception ...	29
<status event-id="SCCSTHFA6#20051109105958#7#1" level="error">Code (-9010) An exception ...	44
<status event-id="SCCSTHFA6#20051109105958#7#1" level="error">Code (-9010) An exception ...	65
<status event-id="SCCSTHFA6#20051109105958#7#1" level="error">Code (-9010) An exception ...	92
Status: Error	108

# Locating text in the Log Viewer, without using the rule list

When the “Display View” is active, use the “Show” button at the bottom right of the screen to display the “Search View”.

1. Enter a Regular Expression or plain text into the “Search View” text field.
2. This can be done either by editing the field directly or by right-clicking on a line in the log viewer.
3. When the right-click option is used, the text is automatically corrected to make it a valid Regex.
4. If only the first instance of that text should be shown in the reference list, select the check box “First occurrence only”
5. Use the Find button



## Writing Log Viewer results to file

It does not make sense to write out the entire log file as text, as it would look exactly like the input file. The menu option "Reporting" | "Export Visible Text Only" writes out text exactly as it's seen in the viewer (Except for formatting colour and bold.) Text formatting in the export will be added later.

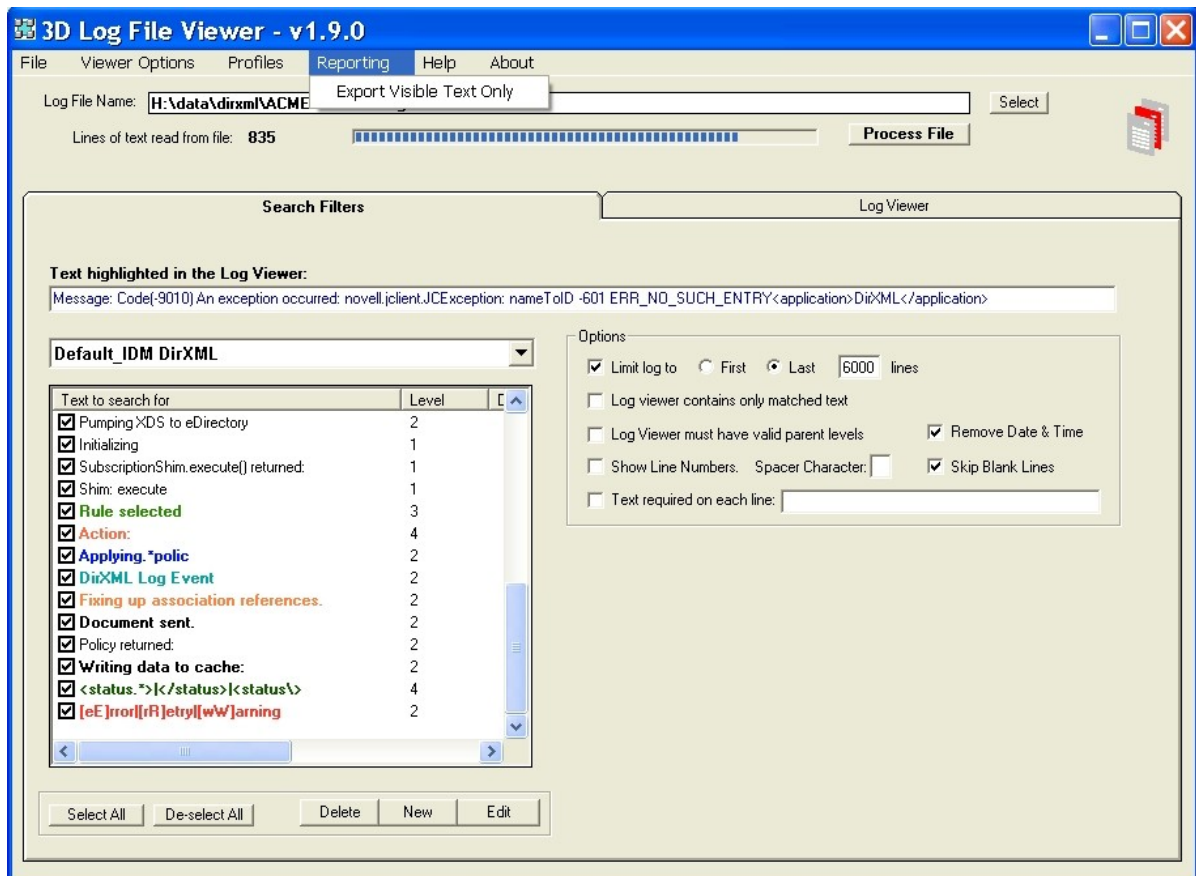
The output is only text, but is reported as indented. So it's still easier to read than the original log file.

Also, be aware that text has actually been omitted, because only the text visible (expanded) in the log view is reported on.

If rules are poorly written to parse the log file, then an incorrect view of the file can be generated adding confusion rather than making things clearer.

But for the purposes of extracting specific data and its subordinates the current export option works.

(A future version should provide formatted output and may support a tree view in a browser.)



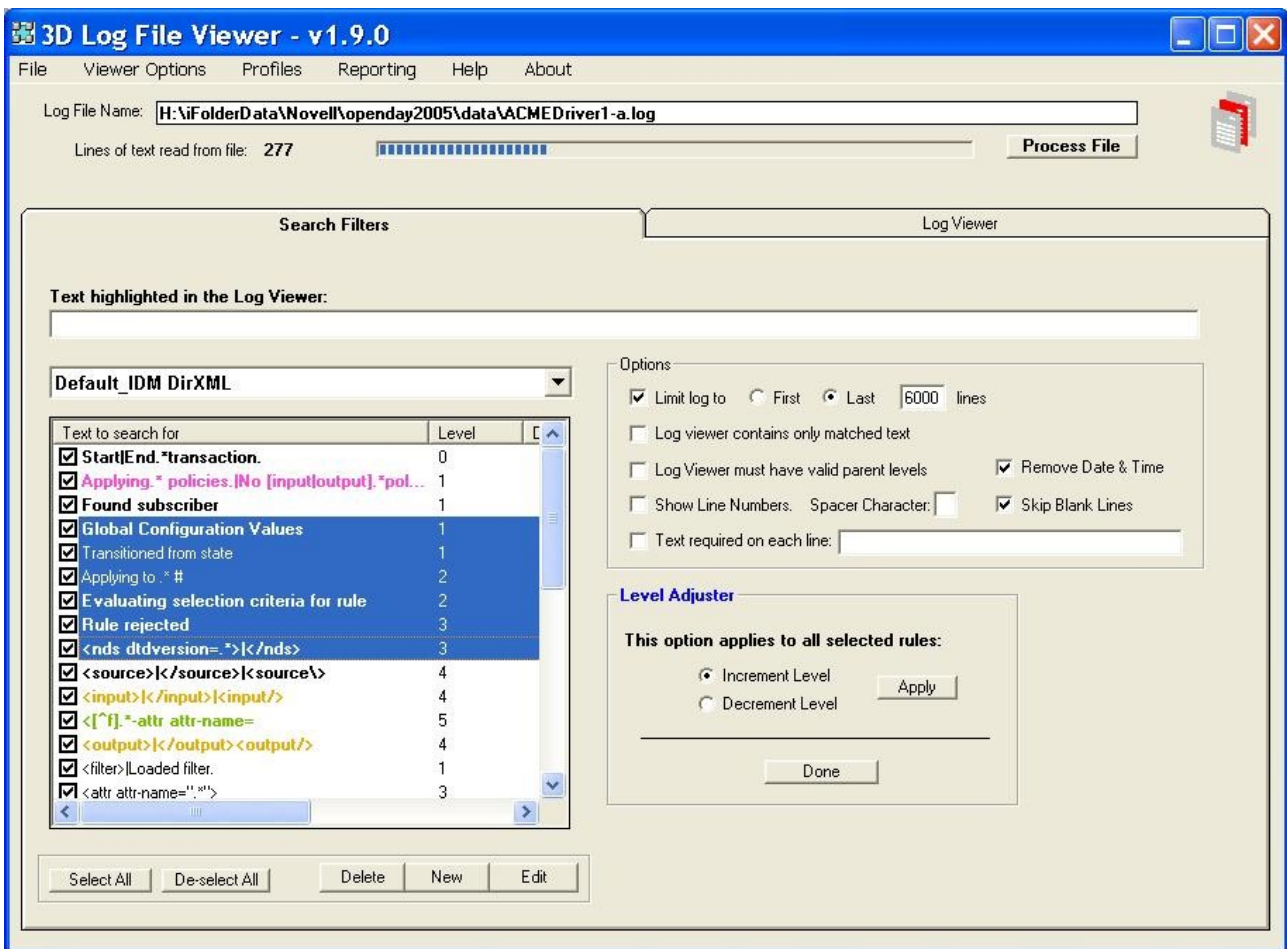
# Changing Rule Levels

You may need to adjust the level of a whole lot of rules at the same time, while keeping them in the same level relationships.

When selecting a Level 1 and a Level 2 rule at the same time, then the Level 1 increments to Level 2 and Level 2 increments to Level 3.

1. Use menu "Profiles", then "Level Adjuster".
2. Highlight the rules which need changing
3. Choose the direction, increment or decrement
4. Click the "Apply" button

If only one rule needs to be changed, then highlight the rule and use the "Edit" button. (The "Level Adjuster" will also work.)

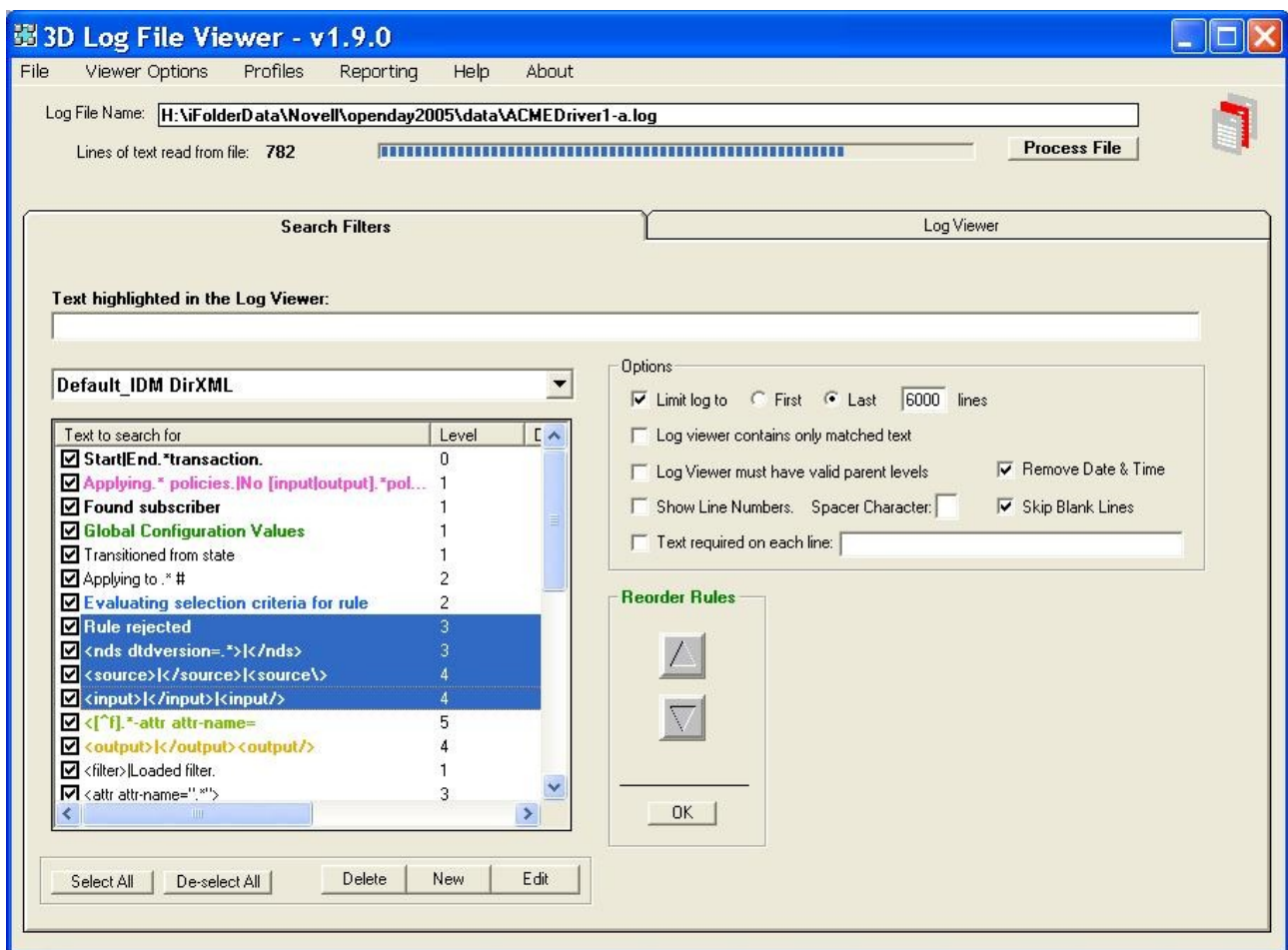


# Reordering Rules

Once you have a few rules created you may find that you would like to keep them together **or** more importantly ensure they execute before other rules.

As you add rules you may discover you've missed some and need to go back afterwards to add them. They will be added to the bottom of the list.

1. Select the menu option "Profile", then "Reorder Rule List".
2. The Reorder Rules option has two buttons: Up and Down.
3. Highlight all the rules you would like to move in one direction (not using the check boxes).
4. Then use the Up/Down button to move all the highlighted rules.
5. Remember to save this when you're done ..."File|Save".



## Refining your searches

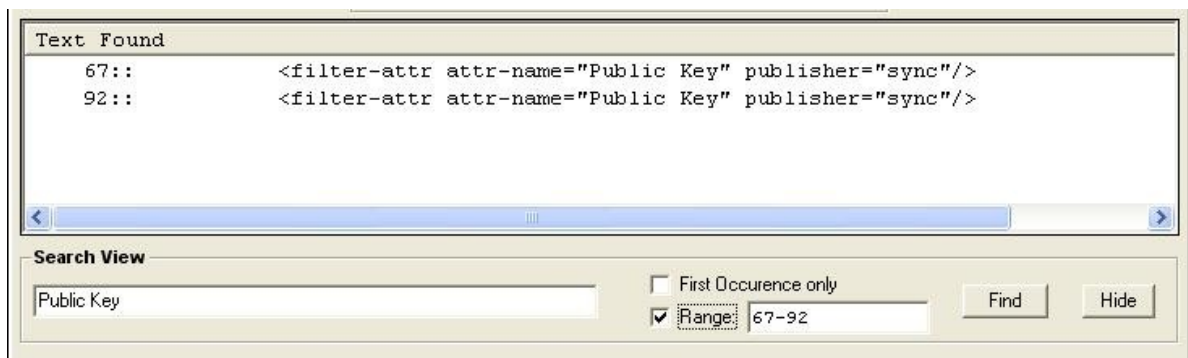
The "Search View" has a "Range" option. Whenever a search is performed the first and last line where text matched the search is recorded in the range.

By checking the box next to the "Range" all further searches are limited to this lines between and including the lines shown in the range field.

This option is very useful for refining searches. Search for user name in an IDM log, for example. Then check the range option. Then use the Profile rules or search field to search for data which may be of interest in connection with this user. Only the lines which have bearing on that user will be searched.

To refine the search further, you would need to manually edit the range at this point. The range field is **only updated** when the range check box is **not** checked.

The range option is unchecked when a file is selected or processed. The range option only have bearing on searches, not on file processing (reading in a file).



# Creating Rules Which Work

In the case of IDM/DirXML creating rules which work in every situation is not an exact science.

One of the reasons why rules don't always do what you expect, is that a rule at a particular Level has no parent level in the tree, it has not been created yet. This situation is most likely to occur at the beginning of the file or if only a part of the file is read in. (Using "Limit Log to".)

Unless you know about every single text variation or have a perfect trace/log in which every line of text that could ever possibly be displayed is in the log, you won't know what rules to write until you see that text you're missing. So the process is one of rule evolution.

A good result is dependent on solid rules firstly and secondly being aware of possible deviations in the log viewer, due to imperfect rules. You can use the check box next to the rule to turn off rules. This is a good way of checking each rule until you see an unexpected result, then examine that rules **text pattern** or it's **level** to determine if they are correct.

To ensure you create valid rules, which work in the real world, do the following:

## DO's

1. Show line numbers
2. Use a spacer character, like "."
3. Try to understand the structure of the file you are processing.  
*This will help you assign the correct parent-child relationship to text*
4. If you have a log/trace taken from startup, then **do** read the file from the beginning.

## DONT's

1. **Do not** skip blank lines.
2. **Uncheck** "Log Viewer must have valid parent levels".
3. **Uncheck** "Log Viewer contains only matched text".
4. **Uncheck** "Text required on each line".

## Testing your rules

1. To test your rules highlight the a Level 0 line, (which is the parent for branch).
2. Use the menu "Viewer Options" | "Expand Branch".
3. Check that all nodes in that branch have been expanded.
4. Check that the line numbers are in sequence.

The two points below indicate that a level is set lower in the tree than it should be. There may be a sub branch which is at the correct level in terms of itself and it's subordinates. Select all the rules which relate to that sub-branch and decrease the level. This will move them all closer to Level 0 and may resolve the issue.

- Closed branches after an "Expand Branch"
- Line numbers out of sequence

As boring as it may seem to create rules for text, if a user of the text is unable to create rules which describe the relationship of the text, then that user is unlikely to be any good at using that text unaided. Rather create too few rules, then go overboard and create rules which don't work in all situations.

# Regular Expression Syntax

## Description

Special characters and sequences are used in writing patterns for regular expressions. The following table describes these characters and includes short examples showing how the characters are used.

Char.	Description
\	Marks the next character as special. <b>n</b> matches the character "n". The sequence <b>\n</b> matches a linefeed or newline character. To use <b>\</b> literally, escape with a <b>\\</b>
^	Matches the beginning of input or line. <b>(Each Log Viewer line is a new line)</b> To use <b>^</b> literally, escape with a <b>\\^</b>
\$	Matches the end of input or line. <b>(Each Log Viewer line is a new line)</b> To use <b>\$</b> literally, escape with a <b>\\\$</b>
*	Matches the preceding character zero or more times. <b>z*</b> matches either <b>z</b> or <b>zoo</b> To use <b>*</b> literally, escape with a <b>\\*</b>
+	Matches the preceding character one or more times. <b>zo+</b> matches <b>zoo</b> but not <b>z</b> To use <b>+</b> literally, escape with a <b>\\+</b>
?	Matches the preceding character <b>zero</b> or <b>one</b> time. <b>a?ve?</b> matches the <b>ve</b> in <b>never</b> To use <b>?</b> literally, escape with a <b>\\?</b>
.	Matches any single character except a newline character. <b>To.</b> matches <b>Top Total</b> To use <b>.</b> literally, escape with a <b>\\. </b>
( <i>pattern</i> )	Matches <i>pattern</i> and remembers the match. The matched substring can be retrieved from the result <b>Array</b> object elements <b>[1]...[n]</b> or the <b>RegExp</b> object's <b>\$1...\$9</b> properties. To use <b>( or )</b> literally, escape with a <b>\\( \\)</b>
<b>x y</b>	Matches either <b>x</b> or <b>y</b> . <b>z.* food</b> matches <b>zoo</b> or <b>food</b> To use <b> </b> literally, escape with a <b>\\ </b>
<b>{n}</b>	<i>n</i> is a non-negative integer. Matches exactly <i>n</i> times. <b>o{2}</b> does not match the <b>o</b> in <b>Bob</b> but matches the first two <b>o's</b> in <b>food</b> To use <b>{ or }</b> literally, escape with a <b>\\{ \\}</b>
<b>{n,}</b>	<i>n</i> is a non-negative integer. Matches at least <i>n</i> times. <b>o{2,}</b> does not match the <b>o</b> in <b>Bob</b> and matches all the <b>o's</b> in <b>food</b> <b>o{1,}</b> is equivalent to <b>o+</b>
<b>{n,m}</b>	<i>m</i> and <i>n</i> are non-negative integers. Matches at least <i>n</i> and at most <i>m</i> times. <b>o{1,3}</b> matches the first three <b>o's</b> in <b>food</b>
<b>[xyz]</b>	A character set. Matches any one of the enclosed characters. <b>[abc]</b> matches the <b>a</b> in <b>plank</b> . To use <b>[ or ]</b> literally, escape with a <b>\\[ \\]</b>
<b>[^xyz]</b>	A negative character set. Matches any character not enclosed. <b>[^abc]</b> matches the <b>p</b> in <b>plain</b>
<b>\\b</b>	Matches a word boundary, such as a space. <b>ea*r\\b</b> matches the <b>er</b> in <b>never early</b>
<b>\\B</b>	Matches a non-word boundary. <b>ea*r\\B</b> matches the <b>ear</b> in <b>never early</b>
<b>\\d</b>	Matches a digit character. Equivalent to <b>[0-9]</b> . <b>File3.txt</b> matches <b>3</b>
<b>\\D</b>	Matches a non-digit character. Equivalent to <b>[^0-9]</b> . <b>File3.txt</b> matches <b>File</b> and <b>.txt</b>

<code>\f</code>	Matches a form-feed character.
<code>\n</code>	Matches a linefeed character.
<code>\r</code>	Matches a carriage return character.
<code>\s</code>	Matches any white space including space, tab, form-feed, and so on. Equivalent to <code>[\f\n\r\t\v]</code>
<code>\S</code>	Matches any nonwhite space character. Equivalent to <code>[^\f\n\r\t\v]</code>
<code>\t</code>	Matches a tab character.
<code>\v</code>	Matches a vertical tab character.
<code>\w</code>	Matches any word character including underscore. Equivalent to <code>[A-Za-z0-9_]</code>
<code>\W</code>	Matches any nonword character. Equivalent to <code>[^A-Za-z0-9_]</code>
<code>\num</code>	Matches <i>num</i> , where <i>num</i> is a positive integer. A <b>reference back</b> to remembered matches. If the text is <b>File3.txt</b> . To access the results in <code>()</code> for pattern <b>File(<i>d</i>).txt</b> it's <code>\1</code> result is <b>3</b>
<code>/n/</code>	Matches <i>n</i> , where <i>n</i> is an octal, hexadecimal, or decimal escape value. Allows embedding of ASCII codes into regular expressions.

## Installation

The file 3DLOGVIEWER.EXE installs and overwrites 3D Log Viewer profile rule files called "Default\_\*.3D.lst". For this reason, if you create your own profile rules, it is important to never name them "Default\_".

During the installation a few DLL's and OCX files are installed. It is possible that some of these files are already present on the machine.

In some cases the files to be installed are **protected by Windows**. If presented with a message similar to the one below, click OK to continue. The software should install and work correctly.



## Using on Linux

The utility should work on Linux using Wine. The code has been tested to the point where the code works. If a problem occurs it should be related to unregistered DLL's or OCX's.

Depending on how Wine is configured or what files are already installed, then various problem scenarios could arise.

What may occur is that files do not get installed (copied to the windows\system directory) – first troubleshoot this by looking at the Wine debug output.

The installer should contain all the required files. If DLL's or OCX's are missing from the installer please notify Novell Support.