

SAP Log Viewer

Administrator & User Guide



SAP J2EE 6.20



© Copyright 2003 SAP AG. All rights reserved

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors. Microsoft®, WINDOWS®, NT®, EXCEL®, Word®, PowerPoint® and SQL Server® are registered trademarks of Microsoft Corporation. IBM®, DB2®, DB2 Universal Database, OS/2®, Parallel Sysplex®, MVS/ESA, AIX®, S/390®, AS/400®, OS/390®, OS/400®, iSeries, pSeries, xSeries, zSeries, z/OS, AFP, Intelligent Miner, WebSphere®, Netfinity®, Tivoli®, Informix and Informix® Dynamic Server™ are trademarks of IBM Corp. in USA and/or other countries.

ORACLE® is a registered trademark of ORACLE Corporation.

UNIX®, X/Open®, OSF/1®, and Motif® are registered trademarks of the Open Group.

Citrix®, the Citrix logo, ICA®, Program Neighborhood®, MetaFrame®, WinFrame®, VideoFrame®, MultiWin® and other Citrix product names referenced herein are trademarks of Citrix Systems, Inc.

HTML, DHTML, XML, XHTML are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

JAVA® is a registered trademark of Sun Microsystems, Inc.

J2EE™ is a registered trademark of Sun Microsystems, Inc.

JAVASCRIPT® is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

SAP, SAP Logo, R/2, RIVA, R/3, SAP ArchiveLink, SAP Business Workflow, WebFlow, SAP EarlyWatch, BAPI, SAPPHIRE, Management Cockpit, mySAP, mySAP.com, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. MarketSet and Enterprise Buyer are jointly owned trademarks of SAP Markets and Commerce One. All other product and service names mentioned are the trademarks of their respective owners.

Disclaimer

Some components of this product are based on Java™. Any code change in these components may cause unpredictable and severe malfunctions and is therefore expressly prohibited, as is any decompilation of these components.

Any Java™ Source Code delivered with this product is only to be used by SAP's Support Services and may not be modified or altered in any way.

Documentation in the SAP Service Marketplace

You can find this documentation at the following address:

<http://service.sap.com/instguides>

Typographic Conventions

Type Style	Represents
<i>Example Text</i>	Words or characters that appear on the screen. These include field names, screen titles, pushbuttons as well as menu names, paths and options. Cross-references to other documentation
Example text	Emphasized words or phrases in body text, titles of graphics and tables
EXAMPLE TEXT	Names of elements in the system. These include report names, program names, transaction codes, table names, and individual key words of a programming language, when surrounded by body text, for example, SELECT and INCLUDE.
Example text	Screen output. This includes file and directory names and their paths, messages, names of variables and parameters, source code as well as names of installation, upgrade and database tools.
Example text	Exact user entry. These are words or characters that you enter in the system exactly as they appear in the documentation.
<Example text>	Variable user entry. Pointed brackets indicate that you replace these words and characters with appropriate entries.
EXAMPLE TEXT	Keys on the keyboard, for example, function keys (such as F2) or the ENTER key.

Icons

Icon	Meaning
	Caution
	Example
	Note
	Recommendation
	Syntax

Contents

- Introduction5
- Architecture.....5
- Purpose5
- Context6
- Prerequisites for Running the Log Viewer6
- Related Documentation6
- Running Log Viewer7
- Icons7
- Installing and Running the Log Viewer Client7
- Installing and Running the Standalone Log Viewer8
- Registering a Log for Viewing9
- Managing Log Viewer Server Properties10
- Viewing Logs13
- General Log Viewer13
- SAT Log Viewer13
- Attribute Setting15
- Docking / Undocking the Application Log15
- Search Strings15
- Show / Hide Columns16
- Generation of the CCMS Monitoring Template17
- Flow17
- Description of the CCMS Agent Template File18

Introduction

This guide discusses features and navigation for the Log Viewer application. The Log Viewer runs as a service on the J2EE server. A Log Viewer client displays log data.

Architecture

The server monitors the system for new logs and makes every log file available as it is created. Clients access the log files as `JMX MBeans` using `RMI`. Currently `Inqmy p4` and the `JRMP` protocol are being used to connect to the server. The `MBean` access allows a client to get a reference to a log-descriptor that allows the creation of queries against the log file.

Logs can be written in any format. The Log Viewer parses logs into columns to provide the merging feature. The records have timestamps which are used to impose sequential order.

Purpose

The J2EE server listens to logging activity and makes every log file available as it is created. The user can open log files and parse data so that columns of data can be accessed. The most recent log messages are displayed first. It is possible to open and view many log files at one time.

The Log Viewer:

- Displays application and system logs
- Displays different logs side by side
- Gives insight into system problems
- Helps with debugging
- Improves performance by providing insights into the inner workings of applications
- Displays information in a useful form, because Log Viewer understands the information in a log
- Brings logs together from the J2EE server and related systems in one place
- Searches logs
- Merges logs that have compatible sequence information across servers, based on time stamps in each record
- Allows control of the amount of log data created with log severity thresholds

Context

Monitoring

You can run a query across many logs with a specific severity level selected, while searching for information in specific logs. This is an easy way to keep track of problems. The Log Viewer also can generate template files for the CCMS agent. Alerts are then generated in the CCMS.

Problem Diagnostics

The messages in log files provide hints to the cause of problematic application behavior.

Performance Tuning

The SAT logs show execution times for user transactions. The SQL trace provides details on the database interactions through the OpenSQL interface.

Prerequisites for Running the Log Viewer

- You must have Java runtime 1.3 or greater installed
- The Log Viewer client is a Swing application. The Log Viewer Service on the SAP J2EE Engine is required.

Related Documentation

For the latest information on the use and administration of Log Viewer, see SAP Note Number 621466.

Running Log Viewer

A Log Viewer server is installed and operational with the installation of the J2EE engine. The Log Viewer client, however, must be unzipped and installed from its installation directory. Once installed, the client is run using a batch file. The client can connect the Log Viewer user via the P4 port to the J2EE engine. The same client can also be used to connect the user via the RMI port to a stand-alone version of the Log Viewer server.

The stand-alone version of the Log Viewer server provides the ability to monitor logs in the system when the J2EE engine is not running properly, does not start up at all, or is not available on the system to monitored.

When the J2EE engine is installed, two .zip files are also written into the installation directory. They are stored in directory, <install>/tools/lib, where <install> is the name of the directory into which the J2EE engine is installed.

- logviewer_Client.zip holds files associated with the dual-purpose client that can connect the user to the Log Viewer service on the J2EE engine, or to the stand-alone Log Viewer.
- logviewer_Server.zip holds files associated with the standalone Log Viewer.

Icons

Use the following icons as you operate the Log Viewer.

Icons	Meanings
	Configure Log Viewer Client
	Set the attributes of the selected log file (if it is controlled by the logging.jar service)
	Dock
	Toggle displaying the search results pane
	Get the latest logs from the server

Installing and Running the Log Viewer Client

To run the client:

1. Unzip logviewer_Client.zip.
2. Run the logviewer.bat batch file to start up the client.
The Log Viewer client user interface window is displayed in the left pane.
3. To configure additional servers, click the *Configure Log Viewer Client* icon, 
The *Configuration and Connect* pane is displayed.
4. Click on *Add* to add additional servers.
The *Add an application server* pane is displayed:
5. Enter the values for *Name*, *Host Name*, *Port Number*, *Server Type*, *User*, and *Password*.

- a. For standalone operation the RMI port is defined in the `LogviewerServer.properties` file under `Logviewer_java.naming.provider.url`.



```
Logviewer_java.naming.provider.url = rmi://localhost:1099
```

- b. For the J2EE engine the port is defined in the P4 properties. The properties can be viewed via the Config Tool under the dispatcher node.
6. When all the values have been entered into the *Add an application server* pane, click *OK*. Control focus returns to the *Configuration and Connect* pane.
 7. Click *OK* again to return to the *Log Viewer* window. The *Log Viewer* client window appears with all the defined servers displayed in the left pane.

See section, *Viewing Logs*, for a description of how to use the Log Viewer user interface.

Installing and Running the Standalone Log Viewer

To install the standalone Log Viewer:

1. Unzip `logviewer_Server.zip`.

The following files are created:

- `server_startup.bat`
- `server_shutdown.bat`
- `register_log.bat`
- `client_startup.bat`

2. Run the `server_startup.bat` batch file to start up the standalone server.
3. Run the `client_startup.bat` batch file to start up the client. The Log Viewer client user interface window is displayed.
4. To register a log for viewing, run the `register_log.bat` batch file. See *Registering a Log for Viewing* for a description of the batch command arguments.
5. In the left pane select the machine to be monitored. A list of application servers is presented in the left pane.
6. Select the application server to be monitored. A list of application logs is presented.
7. Select the specific log to be monitored. The log messages are listed.
8. Run the `server_shutdown.bat` batch file to gracefully shutdown the Standalone Log Viewer server.

See section, *Viewing Logs*, for a description of how to use the Log Viewer user interface.

Registering a Log for Viewing

To register a log for viewing, run the `register_log.bat` batch command file with parameters that identify the log.

The format of the command is the following

```
register_log <hostname> <port> <logfile name> <logfile_type>
```

where:

`<hostname>` is the name of the host system to be monitored.

`<port>` is the number of the port on the system where the server is listening for new logs.

The default port is 5465.



The port for the standalone server can be found in the `LogviewerServer.properties` file. The property is called `socketPort`.

On the J2EE engine you find the port using the *Config Tool* and view the `logviewer` service on the server. The name of the key property is `Logviewer_socketPort`. Note also that the J2EE port has a lifetime only as long as the lifetime of the J2EE engine.

`<logfile_name>` is the full path name of log file to be monitored. The physical file must be on the server.

`<logfile_type>` is the log type to be monitored – a specific log reader is invoked by Log Viewer to view the log type.

Possible log type readers are: SAPJLog, SATLog, and ASCIILog.



To monitor log file `c:\test\test.log` of type `SAPJLog` on local host port 4444, run the `register_log` batch command as follows:

```
register_log localhost 4444 c:\test\test.log SAPJLog
```

Since log files that get registered via the J2EE engine socket (`Logviewer_socketPort`) have the same lifespan as the J2EE Engine, you may want to use some way to register logs persistently. Two options are provided:

- Automatic registration of logs by the logging framework.
- Configuration of ASCII logs in properties. See section *Configuring the Log Viewer Properties*.

Managing Log Viewer Server Properties

Log Viewer server properties can be managed using the J2EE Engine Configuration Tool. While the initial parameter values are adequate to run the Log Viewer, you may need to change values to better suit your logging needs.

To view or change the Log Viewer properties, do the following:

1. Select the J2EE Engine Config Tool.

The user interface window is displayed:

2. In the left pane select the `logviewer`.

The right pane displays the Log Viewer parameter key names and the corresponding current values.

3. View or edit the properties, as required. See the following sections for descriptions of the properties.



All key names that start with “`Logviewer_`” are passed to the Logging Framework. They can be edited there. See the *Logging* administration document.

Configuring the Log Viewer Properties

The description of the purpose, format, and values of key properties follows.

Logviewer_socketPort

This property defines the port where the server listens for new logs. For a description see *section Registering a Log for Viewing*.

Logviewer_java.naming.provider.url

Defines the connection string and port for the standalone server. Set the port value here, when you want to run the standalone server on a port other than 1099.

Logviewer_ASCIILogs

This parameter holds the full path to additional ASCII Logs to be displayed. If you want to add more than one ASCII Log, separate each by a SEMICOLON (;).



```
Logviewer_ASCIILogs=c:\temp\test.txt;c:\temp\jjj.txt
```

Output_directory

The `output_directory` key parameter specifies where in the directory structure the template files are stored. It has the format:

```
Logviewer_output_directory = <directory file path>
```

where `<directory file path>` defines the location where the template files are stored.



MS Windows-based system:

```
Logviewer_output_directory = C:\\usr\\sap\\PRFCLOG\\logmon
```

Double backslashes (\\) are required as directory separators, because the backslash has special meaning in Microsoft Windows. The first backslash serves as the escape character and indicates that the following backslash is actually part of the text string that identifies the directory.



Unix-based system:

```
Logviewer_output_directory = /usr/sap/PRFCLOG/logmon
```

Configuring the Log File Produced by the Log Viewer Server

The following key properties specify Log Viewer logging properties.

```
com.inqmy.services.logviewer.severity = <severity level>
```

The logging framework only writes those log records whose severity is equal to or above the severity level you set. The settings are applied as new log records are written by the application running on the application server. This does not impact the display of records written before the setting was changed.

The following list shows the possible values for <severity level>:

- ALL means that all severity levels are used
- INFO is used for information-only logs
- WARNING is used for minor processing issues
- ERROR is used for more serious process issues
- FATAL indicates a most serious processing issue, beyond which the user should take no action without assistance from the system administrator



If you select the value `ERROR`, then all the values of severity below `ERROR` are not visible in the log file updates. Only all log records with severity settings equal to or above `ERROR` appear under the column *Severity* in the log.

The following values are initially configured and do not need to be changed:

```
log[logviewer] = FileLog
log[logviewer].desc = Logviewer server side
log[logviewer].pattern = logviewer_server.log
log[logviewer].formatter = ListFormatter
```

Parameters for Generating the CCMS Template File

This section describes the parameters that must be set to support the generation of a Computing Center Management System (CCMS) template file. For a description of the CCMS template file generation process, see section .

Creation Flag

This property gives you the option to create a CCMS Template file or not.

It has the format:

```
Logviewer_FlagForGeneratingCCMSTemplate = <binary digit>
```

where <binary digit> is:

0, to specify that no template file is to be created

1, to specify that the template file is to be created, as specified by properties below.



```
Logviewer_FlagForGeneratingCCMSTemplate = 1
```

Search Patterns and CCMS Display Color

These properties define the text string pattern that CCMS is to search for, the log type, and the color in which the corresponding line in the log is to be displayed.

It has the format:

```
Logviewer_<TypeOfFormatter>_<Pattern> = <Color Value>
```

This property is composed of two parts, separated by the UNDERSCORE (_) character:

The <TypeOfFormatter> part specifies the type of log to monitored, and can have one of the following values:

Specify `SAPJLog` for the List Format log type.

Specify `ASCIILog` for the ASCII logs log type.

Specify `SATLog` for the SATTraces log type.

The <Pattern> part specifies the text string pattern to be searched for. Examples are:

Error

Warning

Information

The <Color Value> property can have one of three values:

GREEN

YELLOW

RED



`Logviewer_SAPJLog_Fatal = RED`, indicating a List Format log is to be searched for text "Fatal" and will be displayed on the CCMS monitor screen with color red.

`Logviewer_ASCIIILog_Warning = YELLOW`, indicating an ASCII log is to be searched for text "Warni ng" and will be displayed on the CCMS monitor screen with color yellow.

`Logviewer_ASCIIILog_Debug = GREEN`, indicating an ASCII log is to be searched for text "Debug" and will be displayed on the CCMS monitor screen with color green.

`Logviewer_SATLog_Warning = YELLOW`, indicating an SATTraces log is to be searched for text "Warni ng" and will be displayed on the CCMS monitor screen with color yellow.

Viewing Logs

There are two log viewing formats available:

- General logs except SAT logs
- SAT logs

General Log Viewer

When a log is displayed in the Log Viewer window, the most recent section is initially displayed.

You have the following controls to view the log:

- Move up and down the log using the up and down arrows, or the scroll bar.
- Use the Refresh Selected Log File icon  to show the most current part of the log.
- View the most recent part of the log (the top) using the CONTROL-HOME keys.
- View the oldest part of the log (the bottom) using the CONTROL-END keys.
- Sort the log lines presented by the column headings using the buttons on the column tops.

SAT Log Viewer

SAT log files are displayed in a format different from general logs. When displaying an SAT file the whole file gets read. This is necessary because the SAT records have to appear in a certain order.

Turn on Traces for Users

In SAT Administration section in the portal, turn on traces for the required users, specifying users by name and user ID. See *Portal Platform → System Administration → Monitoring Technology and Logging → Activity Tracing in the SAP Enterprise Portal Administration Guide*. This guide is located on SAP Help Portal at <http://help.sap.com/ep>.

Choose and View an SAT Log

After activation for a certain user, the SAT trace file is displayed by the Log Viewer. Three different record types are shown:

- request
- component
- action

These records are displayed in hierarchical fashion, as seen in the following example:

```

1 Request_X                23:12:20.200 ...
2   Component_X1           23:12:20.200 ...
3     Component_X11        23:12:20.400 ...
4       Component_X12      23:12:20.500 ...
5         Action_X121      23:12:20.550 ...
6   Component_X2           23:12:20.600 ...

```

The following fields are displayed by default:

- `Level` – Nesting level
Necessary because components can be nested and this is interesting information
- `Name` – Name of the request or component
For JARM framework internal reasons, it is not possible to provide action names, but this is requested
- `StartTS` – Start time stamp
Format `HH:MM:SS:MMM` is requested. The date information needs to be considered for sorting purposes but not displayed.
- `GrossTime` – Duration in milliseconds of the activity
Elapsed time
- `NetTime` – Net time in milliseconds of activity (elapsed)
This is the gross time minus the time which was spent in subcalls
- `OutData` – Amount of transferred data in bytes
- `Descr` – Short text description of activity
- `Username` – User name

Attribute Setting

Log file attributes include the formatter used, and the log severity level. Only the log severity level can be set through the log viewer user interface. The attribute settings change the logging level on the log, not the location.

The settings are applied as new log records are written by the application running on the application server. This does not impact the display of records written before the setting was changed.

Typical logging levels are: *Info*, *Warning*, *Error*, *Fatal*, where *Fatal* is the highest severity, and *Info* is least. To explicitly write all levels of severity, specify *All*.

Procedure

1. Select (highlight) the log file for which you want to set the attribute settings. Right click on the log file name and select *Attribute Setting* (alternatively click *Tools* → *Attribute Setting..* from the top menu).
2. On the *Attribute Setting* window, select *Severity*. The possible values appear under the *Values* column. Select the severity value (logging level) that you want to display in the Log Viewer. The severity is an attribute that can be set for the SAP Java Logging framework. To adjust the logging level, the logging framework only writes those log records whose severity is higher than the severity level you set.



If log records are initially being written with the severity level set to *All*, and you set the severity attribute value to *Error*, then all the records with values of severity equal to or greater than *Error* will be visible in the log file updates.

Docking / Undocking the Application Log

You can view the log file within the Log Viewer application window (dock) or as a separate window (undock).

This may be helpful for the case where you want to compare two or more log files simultaneously in separate windows.

Procedure

1. Right click on the log file tab in the Log Viewer window, and select `Undock<file_name>`. This action launches a new log file window.
2. To dock the file, click on the dock icon .

Search Strings

Searches for the first (and next) log messages that contain the search string. The messages of each row in the log can be searched with PERL regular expressions.

Procedure

1. Right click anywhere on the row or the column (not on the column header). The highlighted text indicates the name of the column.
2. On the dialog window enter a case-sensitive PERL search string as the input value and click *Ok*. All lines with that text are selected. The search results appear in the split window.



Examples of PERL search strings.

The search string: `select`

would match:

```
deselect
reselect
selection.
```

The search string: `Select .`

would match:

```
Selects
Select .
```

but *not* match:

```
Selection
```

For a tutorial on PERL regular expressions, see

<http://www.english.uga.edu/humcomp/perl/regex2a.htm>

3. Double click on one row inside the search result window. The upper log file window automatically scrolls to the particular position and highlight the line you have clicked in the log file.

Show / Hide Columns

You can show or hide columns in the Log Viewer. Right click on the column header and select / deselect the column name which you want to show / hide.

It is also possible to move columns. Drag and drop them at the new position.

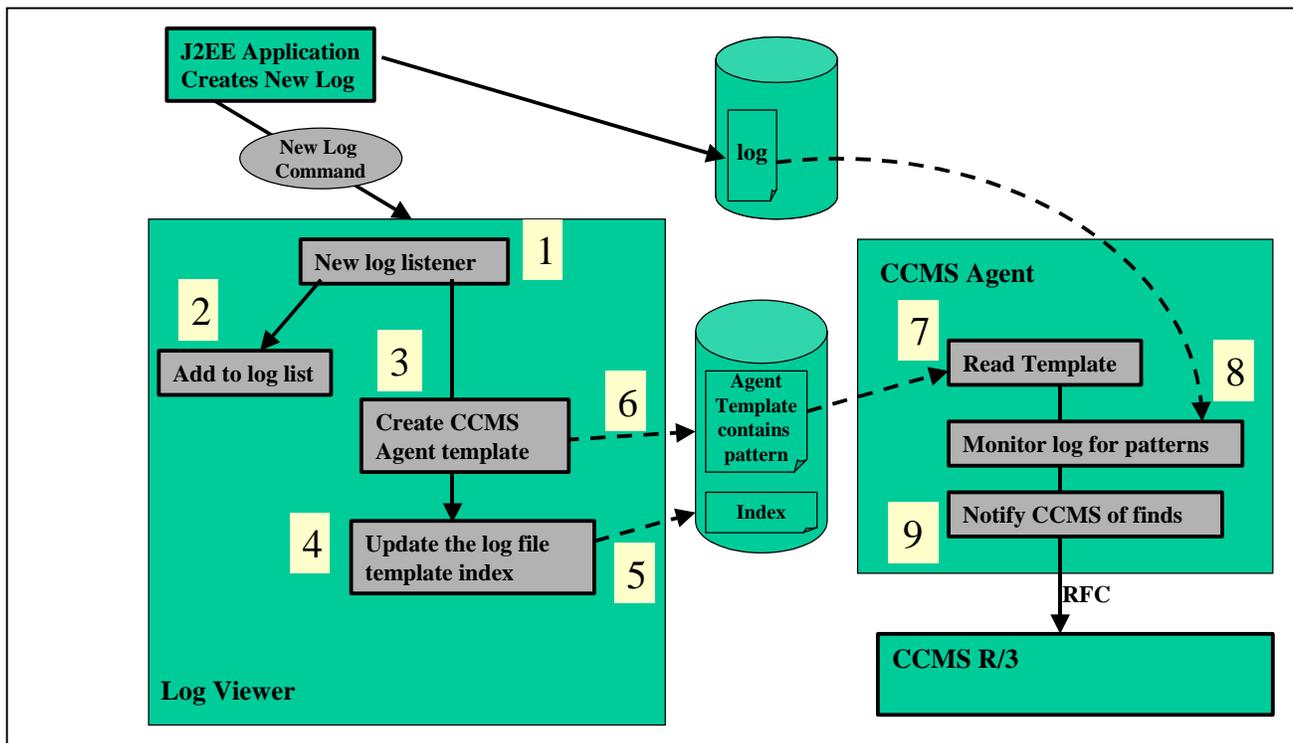
Generation of the CCMS Monitoring Template

Computing Center Management System (CCMS) is an application running on an SAP R/3 system that monitors the state and quality of SAP execution, both within the R/3 System, and in external servers. CCMS receives information from CCMS Agent applications running within the R/3 system and on external servers.

This section describes the format and creation of CCMS Agent Template files (templates) used by CCMS Agents within servers external to R/3. These templates define to the CCMS Agent what log files to monitor, where they are located in the directory structure, and what log output patterns to look for. A CCMS Agent Template is created whenever the Log Viewer receives notification that a new log has been created. The contents of a template file are created according to the values specified in a properties file.

Flow

As illustrated in the figure below, the Log Viewer receives a New Log command (from a J2EE application, for example) to initiate the creation of the CCMS Agent Template file.



The sequence of events is shown by the numbers in the figure above. The Log Viewer gets the location of the log in the file system (1), and adds the log file to the list of log files to be monitored by the Log Viewer (2). The Log Viewer then reads the Log Viewer properties file to determine the text pattern to search for, and in what color CCMS is to display each such line found. Log Viewer then creates the template file (3), inserting the log file path name, the search pattern, and the color value. The log file template index is updated (4) and stored (5). The Log Viewer then stores the template file in the template output directory (6), as specified in the properties file.

When the CCMS Agent finds a new template, it reads the template (7) that corresponds to the new log, from the template output directory. The CCMS Agent begins monitoring the log (8) for text patterns specified in the

template. When the agent finds one of the specified text patterns, it notifies CCMS on the R/3 system via an RFC (9).

Description of the CCMS Agent Template File

This file is created by the Log Viewer and stored in the template output directory, `output_directory`, as specified by the Log Viewer properties. The contents are created, based on the example properties specified in section *Configuring the Log Viewer Properties*.

The file name is of the form: `<log_timestamp>logmon.ini`, where the file name is composed of two parts:

- A unique log time stamp prefix, `<log_timestamp>`



1054170026516

- The required end characters, `logmon.ini`, that indicate this is a log monitoring file.



1054170026516logmon.ini

- The following is an example of a CCMS Agent template file.



```
LOGFILE_TEMPLATE
FILENAME=LogFile0.log
DIRECTORY=C:\Documents and Settings\i803727\properties\CombinedLogMergeLogrecordCounterTest2

PATTERN_0=ERROR
VALUE_0=Pink
PATTERN_1=Warning
VALUE_1=Yellow
PATTERN_2=Info
VALUE_2=White
PATTERN_3=FATAL
VALUE_3=Red
PATTERN_4=Debug
VALUE_4=Blue
.
```