One Pager: GlassFish 3.1 Logging

1. Introduction

1.1. Project/Component Working Name

Logging

1.2. Name(s) and e-mail address of Document Author(s)/Supplier

Naman Mehta, naman.mehta@sun.com

1.3. Date of This Document

Initial Draft: May 13 2010

Updated Draft:
Friday June 14 2010. Based on the comment from Anissa and Vince (http://wiki.glassfish.java.net/Wiki.jsp?page=3.1loggingReviewAnissa).
Friday August 06 2010. Based on AS ARCH comment and Updated some of the missing Section.

Review complete: August 18 2010

2. Project Summary

2.1. Project Description

Enhance a logging facility available for GlassFish Application Server 3.1.

2.2. Risks and Assumptions

The logging facility for GlassFish 3.1 will leverage the JDK logging utility available in v3.0 and continue to upgrade existing logging functionality.

3. Problem Summary

3.1. Problem Area

The logging facility for GlassFish 3.1 will enhance the logging functionality. This means integrating logging for all instances. The logging.p available for all instances and allows user to configure different logging level for different instances. Planning to upgrade existing logging comi –target option.

Adding diagnostic info for all SEVERE, WARNING and INFO messages and also provide wiki page to reference all diagnostic info.

Planning to support CLI commands available in v2.1 for backward compatibility.

Provide CLI to collects all log files from cluster and create as a zip file.

Expand search criteria for reviewing log messages from Log viewer.

Provide support for Rotated Log Files in Log Viewer.

Provide REST support for Log Viewer.

3.2. Justification

Carry forward existing v3.0 functionality and enhance support for cluster environment.

4. Technical Description

4.1. Details

Logging has long been an important part of GlassFish. In v3 the logging facility has been rewritten to address some performance issues and to l logging utility as much as possible. In 3.1 logging is going to integrate to support remote instances, the implementation does include a small amo is present in previous releases of GlassFish v3.0. A significant change in the new code base is that all logger level configuration is now hand logging.properties file available for all instances. The log service is notified when there is a change to the logging.properties file (usually file has will lookup the known loggers in the file and set the log level to the level specifiedglassfish/admin-core/jsr77/src/java/com/sun/enterprise/managem file. All GlassFish loggers are set the level INFO by default. This logging.properties file is different for different instances and available under conf command that uses to set log level, list log levels is going to upgraded to support –target option. The description of properties found in the loggin is found in section <u>4.5 Interfaces</u> which is similar to v3.0. The default logger uses those properties to configure the logging facility. Only the l loggers can changed dynamically. That is no server restart is required for the new values to take affect.

Provide CLI that collects all the logs files for a given target as a zip file. Organize the files under labeled dir - example: node123 nodeXYZ/instance1/server.log, nodeABC/instance2/server.log etc.

As part of this release all SEVER, INFO and WARNING messages should contain messaged ids. All SEVERE and WARNING messages contains d part of LogString.properties file. There should be wiki page maintained for all diagnostic info and which could be accessed from external machi helpful to developers and communities to get more diagnostic info about the error. To verify the log messages planning to come up with build s check for proper messages in server.log file after quicklook run.

Need to provide RESTful API to support log viewing and setting the log levels in logging.properties file.

Povide support for rotated log files in the Log Viewer. And to improve logging facility new search options is going to provided as part of Log Vie upgrade the GUI for the same.

To give backward compatibility CLI logging commands need to be added from GlassFish v2.1.

4.2. Bug/RFE Number(s)

N/A

4.3. In Scope

Software needed to produce logs files that are suitable for analysis by end users, system administrators, developers and ro-field service engineers. capture information such as security failures, configuration errors, diagnostic messages and/or bugs in the server.

4.4. Out of Scope

N/A

4.5. Interfaces

4.5.1 Exported Interfaces

Class Details	Stability	Former Stability (if changing)	Class Name	Co
String setLoggingProperty(String propertyName, String propertyValue, **String target**)		Committed	com.sun.logging.LoggingConfigImpl	Set log nar spe giv
Map updateLoggingProperties(Map properties, **String target**)		Committed	com.sun.logging.LoggingConfigImpl	Set log in t wit con val tar
Map getLoggingProperties(**String target**)		Committed	com.sun.logging.LoggingConfigImpl	Ge pro log for

void removeLoggingProperties(Set properties, **String target**)		Committed	com.sun.logging.LoggingConfigImpl	Re: pro in the log file tar
boolean **createLoggingZip**(String target)			com.sun.logging.LoggingConfigImpl	Ge in : giv
AttributeList fetchRecordsUsingQuery(LogFile logFile, long startingRecord, boolean next, boolean forward, long requestedCount, Date fromDate, Date toDate, String logLevel, boolean onlyLevel, List listOfModules, Properties nameValueMap, **String anySearch**)		Committed	com.sun.enterprise.server.logging.logviewer.backend.LogFilter	Fet log dif opt use Vie Ad opt any of Ad arg any par fur
public AttributeList getLogRecordsUsingQuery(String logFileName, Long fromRecord, Boolean next, Boolean forward, Integer requestedCount, Date fromDate, Date toDate, String logLevel, Boolean onlyLevel, List listOfModules, Properties nameValueMap, **String anySearch, String instanceName**)			com.sun.enterprise.server.logging.logviewer.backend.LogFilter	Thi use giv ins Thi use Vie log ins clu env
public File **downloadGivenInstanceLogFile**(Habitat habitat, Server targetServer, Domain domain, Logger logger, String instanceName, String domainRoot, String logFileName) throws IOException			com.sun.enterprise.server.logging.logviewer.backend.LogFilterForInstance	Thi use coc ins rer Thi SS SF the rer for nar
public void **downloadAllInstanceLogFiles**(Habitat habitat, Server targetServer, Domain domain, Logger logger, String instanceName, String tempDirectoryOnServer) throws IOException			com.sun.enterprise.server.logging.logviewer.backend.LogFilterForInstance	Thi use log to log rer Thi SS: SC the rer for nar
public Vector **getInstanceLogFileNames**(Habitat habitat, Server targetServer, Domain domain, Logger logger, String instanceName) throws IOException			com.sun.enterprise.server.logging.logviewer.backend.LogFilterForInstance	Thi use coc ins nar rer Thi SS: SF the

				fro... ma... ins...
boolean allChecks(LogFile.LogEntry entry, Date fromDate, Date toDate, String queryLevel, boolean onlyLevel, List listOfModules, Properties nameValueMap, **String anySearch**)		Committed		Ac... lin... Re... che... ma... que... tha... tru...
boolean **messageDataCheck**(String message, String anySearch)				Thi... allC... fun... che... con... any... not... tha... tru...
public void execute(AdminCommandContext context)			com.sun.enterprise.server.logging.commands.CollectLogFiles	Thi... get... file... tar... It ... file... ren... and... unc... dir... use...

Important Note/Dependencies:

- Planning to use Cluster Dynamic Reconfig as it provides the infrastructure for all CLIs which uses –target option.

- Plan to use SSH, SCP and FTP functionality to get log files from remote instances .

- Plan to convert existing code to support REST API for setting log levels and backend code to get Log details. Refer to item http://wiki.glassfish.java.net/Wiki.jsp?page=GuiRestRequirements .

    - https://glassfish.dev.java.net/issues/show_bug.cgi?id=12147

    - https://glassfish.dev.java.net/issues/show_bug.cgi?id=12136

How the logging.properties file would be handled in Cluster Scenario?

- If user will pass –target option as Server (DAS) it should update logging.properties file under glassfish/domains/domain1/config/logging.prope...

- If user will pass –target option as Cluster Name it should update logging. properties file under glassfish/domains/domain1/config/<cluster's-co... name>-config/logging.properties.

- If user will pass –target option as Instance Name it should update logging. properties file under glassfish/domains/domain1/config/<instance's... name>-config/logging.properties.

    As per above scenario, there is logging.properties file per config. https://glassfish.dev.java.net/issues/show_bug.cgi?id=12572

    GUI also provides a way to update logger settings. https://glassfish.dev.java.net/issues/show_bug.cgi?id=12510

    Copy-config command should create a logging.properties file under config specific directory (ex.glassfish/domains/domain1/config/cluster1-co... new config element is created in domain.xml. https://glassfish.dev.java.net/issues/show_bug.cgi?id=12570 & https://glassfish.dev.java.net/issue... /show_bug.cgi?id=12571

    Replication framework should handle the replication of file to appropriate target. Replication framework remove synchronization of config/log... file to cluster. User is not allowed to set logging properties for the instance which is part of the cluster. Start-up code is going to take care for ... logging.properties file to be monitor and used.

How the anySearch would work in Log Viewer?

- Planning to add new option in "Log Viewer" screen in "Advance Search" to allow user to search data as part of the Message. User should pro... characters to search data(e.g. Thread=1, NullPointer, ...). It uses '**messageDataCheck'** function from LogFilterForInstance class as des... interface section to filter out logging data.

How the collect-log-file command would work?

- Planning to create new class called CollectLogFiles. CollectLogFiles file use SSH and SCP functionality to download files from remote ma...

target. The target value should be DAS/standalone instance name/cluster name. It gets all log files for given target in case of instance is down

- **CLI Command detail:**
  ./asadmin collect-log-files --target server/cluster's-config-name/instance's-config-name --outputFilePath=<directory specified by user to downl files>

  - **Code Snapshot for CollectLogFiles.java**

    - Uses SSH and SCP utility to get the log file for given instance. Going to use '**downloadAllInstanceLogFiles**' and '**getInstanceLog** functions from LogFilterForInstance class as mentioned in Interface section.

    - Payload.Outbound outboundPayload = context.getOutboundPayload();

    - Uses this outbound payload to attach the file.
      outboundPayload.attachFile(
      "application/octet-stream",
      tempFile.toURI().relativize(fileFromInstance.toURI()),
      "files",
      props,
      fileFromInstance);

How the Log Viewer integrated for cluster?

- Log Viewer screen will provide a new drop down list for all available log files for given instance. To generate the drop do '**getInstanceLogFileNames**' functions from LogFilterForInstance class as mentioned in Interface section. Based on user selection it passes the back end code.

- For the first request in log viewer, it will get the log files from remote instance. Since we are not doing incremental updates, we will need to e that says something like "Refresh". When user clicks on this button, REST API will pass this boolean to logging back end to re-fetch the log fil instance. Otherwise, we will operate on the local log file. It uses SSH functionality and FTP to get log file for given instance. It gets log file for in case of instance is down.

  The log files will be stored under "logs" directory at DAS. We will create instance specific directory as described below.

  For example,
  glassfish/domains/domain1/logs
  |- server.log
  |- jvm.log
  |- instance1/
  |-- server.log
  |- instance2
  |-- server.log

4.5.2 Imported interfaces

N/A

4.5.3 Other interfaces (Optional)

Interface	Stability	Former Stability	Exporting Project: Name, Specification or other Link.	Comments
logging.properties file		Committed	external	Used internally to maintain current log configuration. This logging.properties file available for all instances and properties value may change using CLI commands. This file contains all the logger names and levels, as well as the properties to add custom handlers and properties to specify log file rotation. This logging.properties file is available under all instances and domain. User can allow to configure them using cli command by specifying –target option.
build.xml			external	This script will check log messages in server.log file for common issues (no id, no local string found, no bundle for this logger, unsubstituted parameters, etc.). If anything is missing it would break the quick look at that point. Planning to integrate the script with quicklook.

The properties that is listed under logging.properties. The following properties are available in the v3 release. No plan to add any new properties i

file	com.sun.enterprise.server.logging.GFFileHandler.file	logs/server.log

use-system-logging	com.sun.enterprise.server.logging.SyslogHandler.useSystemLogging	false
log-handler	handlers	java.util.logging.ConsoleHandler
log-filter	com.sun.enterprise.server.logging.GFFileHandler.logFilter	N/A
log-to-console	com.sun.enterprise.server.logging.GFFileHandler.logtoConsole	false
log-rotation-limit-in-bytes	com.sun.enterprise.server.logging.GFFileHandler.rotationLimitInBytes	2000000
log-rotation-timelimit-in-minutes	com.sun.enterprise.server.logging.GFFileHandler.rotationTimelimitInMinutes	Refer to item # 4, 5, 6 on http://wiki.glassfish.java.net/Wiki.jsp?page=GuiRestRequirements 0
Retain-error-statistics-for-hours	com.sun.enterprise.server.logging.GFFileHandler.retainErrorsStasticsForHours	0
Flush-frequency	com.sun.enterprise.server.logging.GFFileHandler.flushFrequency	1

4.6. Doc Impact

All diagnostic info should be part of this page (http://docs.sun.com/app/docs/doc/820-7700/gjkmp?l=en&a=browse).

Logging CLI commands (set-log-level, list-logger-levels) need to be updated for –target option.

Need to add some new commands which are imported from GlassFish v2.1 for backward compatibility.

4.7. Admin/Config Impact

set-log-level, list-logger-levels commands need to be upgraded to support –target options.

New search options need to be added as part of Log Viewer. Search option yet to finalized.

New CLI commands need to be added from previous release of GlassFish 2.1.1 to support backward compatibility. List of the commands are (display display-error-distributions, display-error-statistics).

4.8. HA Impact

N/A

4.9. I18N/L10N Impact

N/A

4.10. Packaging & Delivery

N/A

4.11. Security Impact

N/A

4.12. Compatibility Impact

The new version of the logging code uses the logging.properties file as described in sections 4.1 and 4.5.3.

4.13. Dependencies

- **http://wiki.glassfish.java.net/Wiki.jsp?page=3.1BasicClustering**

     Availability of create-instance command. Also, update create-instance command to create default logging.properties file during creation.

- **http://wiki.glassfish.java.net/Wiki.jsp?page=ClusterDynamicReconfig**

     Availability of new infrastructure to support –target option

- **Need help from Admin GUI team to upgrade LogViewer screen to support new search options.**

- **SFTPClient from SSH team to get log files from remote machine.**

- **REST API related work from Ludo and Jasson.**

5. Reference Documents

// List of related documents, if any (BugID's, RFP's, papers, Blogs).
// Explain how/where to obtain the documents, and what each

// contains, not just their titles.

- **http://wikis.sun.com/display/glassfish/3.1Logging**

    Project link for 3.1 Logging.

6. Schedule

Per GlassFish 3.1 schedule.