CentraSite

Logging

Version 9.5 SP1

November 2013

This document applies to CentraSite Version 9.5 SP1.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Document ID: IINM-AG-LOG-95SP1-20140410

Table of Contents

Logging

This document provides information about the logging functionality of CentraSite.

The content is organized under the following sections:

Introduction	Gives an introduction to the CentraSite's logging and purging functionality.
Overview of Logging	Gives an overview of the logging functionality of CentraSite.
Configuring Logs	Describes how to configure logging in CentraSite.
Monitoring Logs	Describes how to view the various logs in CentraSite.
Overview of Purging	Gives an overview of the purging functionality of CentraSite.
Purging Logs	Describes how to configure purging in CentraSite.
Exporting / Importing Purged Logs	Describes how to export and import the purged log records.
Configuring the Purger Properties for High Volume Data Handling	Describes how to configure purging for high volume data handling.

1 Introduction

To effectively manage the CentraSite registry/repository, it is necessary to get feedback about the activity and performance of the registry/repository. The CentraSite registry/repository provides comprehensive and flexible logging functionality for tracking design/change-time and run-time events in CentraSite Control. CentraSite stores the log and monitoring data of all registry objects (e.g., policies, assets, etc.) as log records in the CentraSite Log Database.

As your logs grow, you may want to purge log records to avoid performance degradation. CentraSite provides the ability to purge log records and back them up to another location. You can purge log records manually (on demand) or automatically on a scheduled basis.

2 Overview of Logging

Logging is the means by which CentraSite provides you with time-stamped and labeled information about the design/change-time and run-time events. Logging provides both a current snapshot of the event as well as a historical view.

By understanding and using Messaging Server log files, you can:

Gather and analyze

- Run-time performance data posted by a target (i.e., a policy enforcement point (PEP) or a run-time monitoring component such as Insight).
- Run-time event data (transaction events, policy violation events, etc.) posted by a target.
- Approval history data.
- Design/change-time policy data.
- Audit data.
- Federation job queue data.
- Consumer Registration data.

Troubleshoot problems

For example, if your CentraSite needs to add more disk storage due to an increase in the number of assets, you can use Audit log files to see what percentage the asset activities has increased by and plan for the amount of new disk storage you need.

System-defined CentraSite Logs

CentraSite can maintain five types of system logs:

- The Policy Log
- The Approval History Log
- Audit Logs
- Run-Time Event Logs

- Run-Time Performance Logs

The Policy Log

The Policy Log contains information about the design/change-time policies that CentraSite has executed. By default, CentraSite only logs information about policies that fail. However, you can optionally configure CentraSite to log information about policies that resulted in success, informational, warning, and failure alerts (see ***Configuring Logs***).

In addition, if you purge the policy log, CentraSite makes an entry in the policy log to indicate that entries have been purged.

To view the Policy Log, you must belong to a role that includes the "View Policy Log" permission. To see the list of predefined roles that include this permission, see the section *About Roles and Permissions* in the document *Users, Groups, Roles and Permissions*.

For the procedure to view the Policy Log, see the topic *Viewing the Policy Log* in the section *Functional Scope* in the document *Working with Design/Change-Time Policies*.

The Approval History Log

The Approval History Log contains a record of all approval requests that have been triggered by a policy with an approval workflow action. This log shows the status of each approval request that has been submitted to CentraSite.

To view the Approval History Log, you must belong to a role that includes the "View Approval History" permission. To see the list of predefined roles that include this permission, see the section *About Roles and Permissions* in the document *Users, Groups, Roles and Permissions*.

To view the Approval History Log, perform the following steps:

To view the Approval History Log

1 In CentraSite Control, go to **Administration > Logs > Approval History**.

2 Filter your search for approval log information by completing the following fields and clicking **Search**:

In this field...	Do the following...
Object Name	The object that was submitted for approval.
Approval Flow Name	The name assigned to the approval workflow.
Requestor	Click **Select User** and select the user who requested the approval.
Approver	Click **Select User(s)** and select the user(s) who approved the request.
Request Date From/To	Use the calendar to specify a date range for the requests.
Approval Date From/To	Use the calendar to specify a date range for the approvals.

In this field...	Do the following...
Status	Select a status of the approval request (e.g., In Progress, Approved, etc.).

3 To view details of a particular approval workflow, click the hyperlinked value in the **Approval Flow Name** column.

4 The **Approval Flow Information** panel provides the following information about the approval workflow.

Field	Description
Approval Flow Name	Name of the approval workflow.
Mode	Mode of the approval workflow (e.g., Anyone or Everyone).
Status	Status of the approval policy (e.g., In Progress, Approved, New, No Action, Pending or Rejected).
Creation Date	Date when the approval workflow was created.

5 The **Requestor Summary** panel provides the following information about the requestor of the approval workflow.

Field	Description
Requestor Name	User who triggered the approval workflow.
Approval Type	Name of the approval action (for example, Initiate Approval or Initiate Group-dependent Approval)
Entity	Name of the entity on which the approval was triggered.
Reason for Request	Additional comments or descriptive information stating the reason for the approval request.

6 The **Approver Summary** panel provides the following information about the approver(s) of the approval workflow.

Field	Description
Approver	User who approved or rejected the approval workflow.
Action	Action of the approver (e.g., Approved or Rejected).
Comments	Additional comments or descriptive information stating the reason for approval or rejection.

Audit Logs

An audit log reports on the creation/update activities of a particular asset (including changes in an asset's lifecycle state).

You can view the audit logs of any asset that you can view. For more information about asset permissions, see the section *Setting Permissions on an Asset* in the document *Using the Asset Catalog*.

To view an asset's audit log, perform the following steps:

▶ To view an asset's audit log

1. Display the asset's detail page and select the **Audit Log** profile as described in the section *Viewing Details for an Asset* in the document *Using the Asset Catalog*.
2. The **Audit Log** profile displays the following information:

Field	Description
Event Type	The type of event that was executed on the asset (e.g., created or updated).
Date/Time	The date and time that the event was executed.
User	The user who executed the event.
Comment	Descriptive information about the event executed on the asset.

Run-Time Event Logs

The run-time event log contains information about run-time events that have occurred in a target (i.e., a policy-enforcement point (PEP) or a run-time monitoring component).

The target publishes to CentraSite the run-time events that have occurred (assuming that the target type contains a MIB file in its target type definition file, as described in the section *Creating and Managing Target Types* in the document *Managing Targets and Run-Time Events*). CentraSite provides predefined event types for use with webMethods Mediator or any third-party PEP that is integrated with CentraSite (or more information, see the section *Run-Time Events* in the document *Managing Targets and Run-Time Events*).

Each asset has its own run-time event log, which is located on the Events profile on its detail page. (To view an asset's detail page, see the section *Viewing Details for an Asset* in the document *Using the Asset Catalog*.) For a description of the Events profile of a virtual service, see the topic *The Events Profile* in the section *Viewing or Editing the Profiles of Virtualized Services* in the document *Working with Virtualized Services*. The Events profile is similar for all assets.

By default, all the predefined event types are logged, but you may disable any type (see ***Configuring Logs***).

Users with the proper permissions can perform these additional tasks:

- View a log of all run-time events that have occurred in a particular target or in all targets system-wide.
- Create and manage custom run-time event types for use with webMethods Mediator or any third-party PEP that is integrated with CentraSite.

For more information about these additional tasks, see the section *Run-Time Events* in the document *Managing Targets and Run-Time Events*.

Run-Time Performance Logs

Targets capture run-time metrics for assets. If you are using the Mediator target, Mediator's data collector captures Key Performance Indicator (KPI) metrics for each virtual service and publishes them to CentraSite at regular intervals. If you are using a run-time monitoring component such as Insight, the monitoring component captures the KPI metrics of all rogue assets and publishes them to CentraSite at regular intervals.

Each asset has its own performance log, which is located on the Performance profile on its detail page. (To view an asset's detail page, see the section *Viewing Details for an Asset* in the document *Using the Asset Catalog*.) For a description of the performance metrics of a virtual service, see the section *Run-Time Events* in the document *Managing Targets and Run-Time Events*. The Performance profile is similar for all assets.

By default this log is enabled, but you may disable it as described in ***Configuring Logs***.

3 Configuring Logs

CentraSite uses the system-defined log settings to store activities and performances of some of the registry objects, such as policies and assets, as log records. However, in some cases, you might want to modify the log settings configuration, in order to determine the optimal performance for CentraSite. This section describes some of the methods that you can use for such purposes.

These methods allow you to display and modify the log settings configuration. The log settings configuration applies to each log record.

The command line tool is CentraSiteCommand.cmd (on Windows) or CentraSiteCommand.sh (on UNIX/Linux), and is located in `<CentraSiteInstallDir>`*/utilities*.

Prerequisites

To be able to configure the log settings, please note the following points:

- To configure from a command line, the CentraSite Registry Repository must be online, and the tool requires a Java 6 runtime.
- Changes to the log configuration do not affect the currently running tasks.

Configuring Log Settings from the Command Line

You can configure log settings by executing the following command in the command line interface *CentraSiteCommand.cmd* (Windows) or *CentraSiteCommand.sh* (UNIX) of Command Central. The tool is located in `<CentraSiteInstallDir>`/utilities.

If you start this command line tool with no parameters, you receive a help text summarizing the required input parameters.

The parameters of the command are case-sensitive, so for example the parameter "-url" must be specified as shown and not as "-URL".

To invoke log setting from the command line, you must perform the following high-level steps:

1. Create a configuration (*config.xml*) file as described in **Creating a Configuration File to Define Log Setting**.
2. Execute the script file with appropriate input parameters as described in **Executing the Script File that Invokes Log Setting**.

⚠ **Important:** You must be a user with CentraSite Administrator role to execute the script file.

Creating a Configuration File to Define Log Setting

This section will describe how to create a config.xml file that defines a log setting. Examine the config.xml file. It contains at least the XML namespace used for providing uniquely named elements and attributes.

- Define a Log Unit
- Define Log Settings

Define a Log Unit

The unique element `LogUnit` represents the `com.centrasite.config.log.unit` property in config,xml. The `LogUnit` element specifies the log record to be stored in the CentraSite Log Database.

In the `LogUnit` element, add an attribute called `name` and assign it a value equal to either of the following:

Log Unit	Description
`Policy_Log`	Contains information about the design/change-time policies that CentraSite has executed.
`Approval_Log`	Contains information about all approval requests that has been triggered by a policy with an approval workflow action.
`Audit_Log`	Contains information on the creation/update activities of a particular asset (including changes in an asset's lifecycle state).
`Runtime_Event_Log`	Contains information about run-time events that have occurred in a target (i.e., a policy-enforcement point (PEP) or a run-time monitoring component).
`Runtime_Performance_Log`	Contains information about the KPI metrics of all rogue assets.

This `LogUnit` element should look similar to this:

```
<LogUnit name="Policy_Log">
    </LogUnit>
```

Define Log Settings

The `com.centrasite.config.log.setting` property represents the `LogSetting` attribute. This attribute defines specific information about the performance of the registry object.

In this section, you create log settings that will be used within the log unit definition.

Within the `LogUnit` element, create `LogSetting` attributes say, `Success`, `Failure` etc.

You can configure the `LogUnit` element with one of the appropriate `LogSetting` attributes.

Log Unit	Description	
Policy Log	**Log Setting**	**Logs**
	Success	Policies that have resulted in success alert.
	Info	Policies that have resulted in informational alert.
	Warning	Policies that have resulted in warning alert
	Failure	Policies that have resulted in failure alert.
	For more information, see *The Policy Log*.	
Approval Log	**Log Setting**	**Logs**
	Approval and Rejection	Approval policies that have resulted both in Approved and Rejected state.
	Rejection only	Approval policies that have resulted in Rejected state.
	For more information, see *The Approval History Log*.	
Audit Log	**Log Setting**	**Logs**
	Enable Audit	All activities (such as creation, modified etc) performed on a particular asset.
	For more information, see *Audit Logs*.	
Run-Time Event Log	**Log Setting**	**Logs**
	Policy Violation Events	Monitors and tracks all the policy violation events.
	Transaction Events	Monitors and tracks all the transaction events.
	Monitoring Events	Monitors and tracks all the monitoring events.
	Lifecycle Events	Monitors and tracks all the lifecycle events.
	For more information, see *Run-Time Event Logs*.	
Run-Time Performance Log	**Log Setting**	**Logs**
	Log Performance Events	Monitors and tracks all the performance events
	For more information, see *Run-Time Performance Logs*.	

This `LogSetting` attribute should look similar to this:

```
LogUnit name="Policy_Log">
        <LogSetting>Success</LogSetting>
        <LogSetting>Failure</LogSetting>
    </LogUnit>
```

The final config.xml should look similar to this:

```
<LogSettings xmlns="http://namespaces.centrasite.com/configurations/logs">
    <LogUnit name="Policy_Log">
        <LogSetting>Success</LogSetting>
    </LogUnit>
    <LogUnit name="Approval_Log">
        <LogSetting>Approval_and_Rejection</LogSetting>
    </LogUnit>
    <LogUnit name="Audit_Log">
        <LogSetting>Enable_Audit</LogSetting>
    </LogUnit>
   <LogUnit name="Runtime_Event_Log">
        <LogSetting>Transaction_Events</LogSetting>
        <LogSetting>Policy_Violation_Events</LogSetting>
        <LogSetting>Monitoring_Events</LogSetting>
        <LogSetting>Lifecycle_Events</LogSetting>
   </LogUnit>
   <LogUnit name="Runtime_Performance_Log">
        <LogSetting>Log_Performance_Events</LogSetting>
   </LogUnit>
</LogSettings>
```

Note: Make sure you copy this file somewhere within the file system of the machine where CentraSite is installed.

Executing the Script File that Invokes Log Setting

To invoke log setting, you must use the command line utility CentraSiteCommand with the config.xml input parameter, as follows:.

```
CentraSiteCommand set Log [-url <CENTRASITE-URL>] -user <USER-ID> -password
<PASSWORD> -file <CONFIG-FILE>
```

Example

```
CentraSiteCommand set Log -url "http://localhost:53307/CentraSite/CentraSite" -user ↩
"Administrator" -password
"manage" -file "config.xml"
```

Input Parameters

The following table describes the complete set of input parameters that you can use with the `set Log` utility:

Parameter	Description
`-url`	The fully qualified URL (http://localhost:53307/CentraSite/CentraSite) for the CentraSite registry/repository.
`-user`	The user ID of a user who has the "CentraSite Administrator" role.
`-password`	The password of the user identified by the parameter "-user".
`-file`	The URI (file: or http:) of the configuration file.

4

Monitoring Logs

When you've configured CentraSite to log messages you can monitor its activities by viewing the log messages. By examining the log files you can monitor many aspects of CentraSite's events.

Policy Log

To view the policy log

1 In CentraSite Control, go to **Administration > Logs > Policy Log**.

2 Complete the following fields to specify which type of log entries you want to view:

<table>
<tr><th>In this field...</th><th>Specify...</th></tr>
<tr><td><b>Object Name</b></td><td><i>Optional</i>. A pattern string that describes the names of the objects (of <b>Object Type</b>) whose log entries you want to view.
You can provide the exact name or use a pattern string consisting of a character sequence and/or the % wildcard character (which represents any string of characters). For example, if you specify the pattern string ‘A%’, CentraSite displays entities whose names start with ‘A’.
Leave <b>Entity Name</b> empty to view all names.</td></tr>
<tr><td><b>Policy Type</b></td><td>The type of policy whose log entries you want to view. To view the log entries for design/change-time policies, select <b>Design/Change Time</b> from the drop-down list (if it is not already selected).</td></tr>
<tr><td><b>Object Type</b></td><td>The object type whose log entries you want to view.</td></tr>
<tr><td><b>Event Type</b></td><td>The event type whose log entries you want to view.</td></tr>
<tr><td><b>Policy Status</b></td><td>The policy execution status that you want to view.
A policy's execution status is the result set of each of its action's execution result. CentraSite writes the following policy execution status to the policy log depending on the log configuration:
<table>
<tr><th>Icon</th><th>Description</th></tr>
<tr><td>🟩</td><td><i>Success</i>. Displays policies that have resulted in success alert.</td></tr>
<tr><td>ℹ</td><td><i>Info</i>. Displays policies that have resulted in informational alert.</td></tr>
<tr><td>△</td><td><i>Inprogress</i>. Displays policies that have resulted in inprogress alert.</td></tr>
<tr><td>⚠</td><td><i>Warning</i>. Displays policies that have resulted in warning alert.</td></tr>
<tr><td>●</td><td><i>Failure</i>. Displays policies that have resulted in failure alert.</td></tr>
</table></td></tr>
</table>

In this field...	Specify...
Execution Date	*Optional*. The time period that you want to examine. Leave the **From** and **To** fields empty to view log entries for all dates.

3 Click **Search** to retrieve the specified log entries.

4 To view details for a particular entry in the returned list, click the name of the policy.

Note: If a policy included a WS-I action, the log entry for the policy will include a link to the results of the WS-I action.

The Approval History Log

The Approval History Log contains a record of all approval requests that have been triggered by a policy with an approval workflow action. This log shows the status of each approval request that has been submitted to CentraSite.

To view the Approval History Log, you must belong to a role that includes the "View Approval History" permission. To see the list of predefined roles that include this permission, see the section *About Roles and Permissions* in the document *Users, Groups, Roles and Permissions*.

To view the Approval History Log, perform the following steps:

To view the Approval History Log

1 In CentraSite Control, go to **Administration > Logs > Approval History**.

2 Filter your search for approval log information by completing the following fields and clicking **Search**:

In this field...	Do the following...
Object Name	The object that was submitted for approval.
Approval Flow Name	The name assigned to the approval workflow.
Requestor	Click **Select User** and select the user who requested the approval.
Approver	Click **Select User(s)** and select the user(s) who approved the request.
Request Date From/To	Use the calendar to specify a date range for the requests.
Approval Date From/To	Use the calendar to specify a date range for the approvals.
Status	Select a status of the approval request (e.g., In Progress, Approved, etc.).

3 To view details of a particular approval workflow, click the hyperlinked value in the **Approval Flow Name** column.

4 The **Approval Flow Information** panel provides the following information about the approval workflow.

Field	Description
Approval Flow Name	Name of the approval workflow.
Mode	Mode of the approval workflow (e.g., Anyone or Everyone).
Status	Status of the approval policy (e.g., In Progress, Approved, New, No Action, Pending or Rejected).
Creation Date	Date when the approval workflow was created.

5 The **Requestor Summary** panel provides the following information about the requestor of the approval workflow.

Field	Description
Requestor Name	User who triggered the approval workflow.
Approval Type	Name of the approval action (for example, Initiate Approval or Initiate Group-dependent Approval)
Entity	Name of the entity on which the approval was triggered.
Reason for Request	Additional comments or descriptive information stating the reason for the approval request.

6 The **Approver Summary** panel provides the following information about the approver(s) of the approval workflow.

Field	Description
Approver	User who approved or rejected the approval workflow.
Action	Action of the approver (e.g., Approved or Rejected).
Comments	Additional comments or descriptive information stating the reason for approval or rejection.

Audit Logs

An audit log reports on the creation/update activities of a particular asset (including changes in an asset's lifecycle state).

You can view the audit logs of any asset that you can view. For more information about asset permissions, see the section *Setting Permissions on an Asset* in the document *Using the Asset Catalog*.

To view an asset's audit log, perform the following steps:

▶ To view an asset's audit log

1. Display the asset's detail page and select the **Audit Log** profile as described in the section *Viewing Details for an Asset* in the document *Using the Asset Catalog*.
2. The **Audit Log** profile displays the following information:

Field	Description
Event Type	The type of event that was executed on the asset (e.g., created or updated).
Date/Time	The date and time that the event was executed.
User	The user who executed the event.
Comment	Descriptive information about the event executed on the asset.

Run-Time Event Logs

The run-time event log contains information about run-time events that have occurred in a target (i.e., a policy-enforcement point (PEP) or a run-time monitoring component).

The target publishes to CentraSite the run-time events that have occurred (assuming that the target type contains a MIB file in its target type definition file, as described in the section *Creating and Managing Target Types* in the document *Managing Targets and Run-Time Events*). CentraSite provides predefined event types for use with webMethods Mediator or any third-party PEP that is integrated with CentraSite (for more information, see the section *Run-Time Events* in the document *Managing Targets and Run-Time Events*).

Each asset has its own run-time event log, which is located on the Events profile on its detail page. (To view an asset's detail page, see the section *Viewing Details for an Asset* in the document *Using the Asset Catalog*.) For a description of the Events profile of a virtual service, see the topic *The Events Profile* in the section *Viewing or Editing the Profiles of Virtualized Services* in the document *Working with Virtualized Services*.

By default, all the predefined event types are logged, but you may disable any type (see ***Configuring Logs***).

Users with the proper permissions can perform these additional tasks:

- View a log of all run-time events that have occurred in a particular target or in all targets system-wide.
- Create and manage custom run-time event types for use with webMethods Mediator or any third-party PEP that is integrated with CentraSite.

For more information about these additional tasks, see the section *Run-Time Events* in the document *Managing Targets and Run-Time Events*.

Run-Time Performance Logs

Targets capture run-time metrics for assets. If you are using the Mediator target, Mediator's data collector captures Key Performance Indicator (KPI) metrics for each virtual service and publishes them to CentraSite at regular intervals. If you are using a run-time monitoring component such as Insight, the monitoring component captures the KPI metrics of all rogue assets and publishes them to CentraSite at regular intervals.

Each asset has its own performance log, which is located on the Performance profile on its detail page. (To view an asset's detail page, see the section *Viewing Details for an Asset* in the document *Using the Asset Catalog*.) For a description of the performance metrics of a virtual service, see the section *Run-Time Events* in the document *Managing Targets and Run-Time Events*. The Performance profile is similar for all assets.

By default this log is enabled, but you may disable it as described in ***Configuring Logs***.

5 Overview of Purging

The term "obsolete" refers to any data not needed by the database but still occupying space on it. This unwanted or obsolete data can eventually fill up the disk and decrease the database performance, causing time-consuming database lookups, contention issues and so on. The process of systematically removing this unwanted data from the database is called "purging". This process basically involves running the Purger scripts. They are available for all log unitCentraSites stored in the CentraSite Log Database.

Before purging log records, you must adjust the automatic or manual purging configuration parameters. Typically, you need the CentraSite Log Database to include only the recent log records. For instance, each CentraSite object will have a number of actions performed on it, and for each action a separate audit log is generated and stored in the CentraSite Log Database. This will increase the size of the CentraSite Log Database dramatically. Failure to purge the excess log records in the database could cause problems. Therefore, the purge interval should be adjusted as required.

Each time you purge a log unit, the corresponding purging log entry is created and can be viewed in CentraSite Control. For example, when you purge a certain set of policy logs, the corresponding purging log entry is automatically created and will be visible in the Policy Log page.

6 Purging Logs

For performance reasons, CentraSite uses system log records to store activities and performances of some of the registry objects (say, policies, assets etc.) that are defined via the log settings. However, in some cases, you might want to modify the log settings configuration, in order to avoid performance degradation of CentraSite. This section describes some of the methods that you can use for such purposes.

These methods allow you to display and modify the log purging configuration. The log purging configuration applies to each log record.

Prerequisites

To be able to configure the log purging, please note the following points:

- To configure from a command line, the CentraSite Registry Repository must be online, and the tool requires a Java 6 runtime.
- Changes to the log purging configuration do not affect the currently running tasks.

Note that CentraSite does not purge:

- Policy logs in "InProgress" state.
- Auditable events that log on object creation.
- Approval logs in "Pending" state.
- Federation job queue data.
- Consumer registration data.
- Purged audit log data (that is, records of the audit logs that were purged earlier).

Note: When you purge policy log entries, CentraSite creates a new informational policy log entry indicating that policy log entries have been purged. If any such informational policy log entries exist from previous invocations of the policy log purger, they are also purged, so that the only such policy log entry remaining is the one for the current invocation.

Configuring Log Purging from the Command Line

You can configure log purging by executing the following command in the command line interface *CentraSiteCommand.cmd* (Windows) or *CentraSiteCommand.sh* (UNIX) of Command Central. The tool is located in `<CentraSiteInstallDir>`/utilities.

If you start this command line tool with no parameters, you receive a help text summarizing the required input parameters.

The parameters of the command are case-sensitive, so for example the parameter "-url" must be specified as shown and not as "-URL".

To invoke log purging from the command line, you must perform the following high-level steps:

1. Create a configuration (*config.xml*) file as described in **Creating a Configuration File to Define Log Purging**.
2. Execute the script file CentraSiteCommand with appropriate input parameters as described in **Executing the Script File that Invokes Log Purging**.

⚠ **Important:** You must be a user with CentraSite Administrator role to execute the script file.

Creating a Configuration File to Define Log Purging

This section describes how to create a config.xml file that defines a log purging. Examine the config,xml file. It contains at least the XML namespace used for providing uniquely named elements and attributes.

- Define a Log Purge Setting
- Define Log Purge Type

Define a Log Purge Setting

The `LogPurgeSetting` element specifies the type of log record to be purged from the CentraSite Log Database.

In the `LogPurgeSetting` element, add an attribute called `log` and assign it a value equal to either of the following:

Log Unit	Description
`Policy_Log`	Purge log record that contains information about the design/change-time policies that CentraSite has executed.
`Approval_Log`	Purge log record that contains information about all approval requests that has been triggered by a policy with an approval workflow action.
`Audit_Log`	Purge log record that contains information on the creation/update activities of a particular asset (including changes in an asset's lifecycle state).
`Runtime_Event_Log`	Purge log record that contains information about run-time events that have occurred in a target (i.e., a policy-enforcement point (PEP) or a run-time monitoring component).
`Runtime_Performance_Log`	Purge log record that contains information about the KPI metrics of all rogue assets.

This LogPurgeSetting element should look similar to this:

```
<LogPurgeSetting log="Runtime_Event_Log">
        </LogPurgeSetting>
   <LogPurgeSetting log="Runtime_Performance_Log">
        </LogPurgeSetting>
```

Define Log Purge Type

The log purge type attributes provide information about the different type of logs configured to be purged on different days, times and even how many days of logs to keep.

In this section, you create the purge type attributes that will be used by the log purge setting definition.

Within the `LogPurgeSetting` element, create the following attributes (as required) and assign values.

Log Purge Setting	Description
`ExportLocation`	Specifies a location for the exported files on the database.
`Until`	Deletes log records and exports them as an archive to a configurable directory that can be imported later. However, if the export location is not specified, then CentraSite simply deletes the log records. **Note:** Specify the xs:dateTime values as required. The default time zone is UTC/GMT.
`OlderThan`	Deletes log records that are older than the specified date and exports them as an archive to a configurable directory that can be imported later. However, if the export location is not specified, then CentraSite simply deletes the log records. **Note:** Specify the xs:dateTime values as required. The default time zone is UTC/GMT.
`CommitThreshold`	Defines the threshold value for batch purging each of the log units. The default threshold value is "10000".

The LogPurgeSetting element should now look similar to this:

```
<LogPurgeSetting log="Runtime_Event_Log">
        <OlderThan>-P5D</OlderThan>
    </LogPurgeSetting>
   <LogPurgeSetting log="Runtime_Performance_Log">
             <Until>2002-05-30</Until>
            <CommitThreshold>1000</CommitThreshold>
    </LogPurgeSetting>
```

The final config.xml should look similar to this:

```
<LogPurgeSettings xmlns="http://namespaces.centrasite.com/configurations/logs">
   <LogPurgeSetting log="Approval_Log">
        ↩
<ExportLocation>/home/usr/admin/centrasite/log/backups/approval</ExportLocation>
    </LogPurgeSetting>
   <LogPurgeSetting log="Runtime_Event_Log">
        <OlderThan>-P5D</OlderThan>
    </LogPurgeSetting>
   <LogPurgeSetting log="Runtime_Performance_Log">
             <Until>2002-05-30</Until>
            <CommitThreshold>1000</CommitThreshold>
    </LogPurgeSetting>
</LogPurgeSettings>
```

Note: Make sure you copy this file somewhere within the file system of the machine where CentraSite is installed.

Executing the Script File that Invokes Log Purging

To invoke log purging function, you must use the script file CentraSiteCommand with the config.xml input parameter, as follows:

```
CentraSiteCommand purge Logs [-url <CENTRASITE-URL>] -user <USER-ID> -password
<PASSWORD> [-assets <Asset Name/Key (s)>] -file <CONFIG-FILE>
```

Example

```
CentraSiteCommand purge Logs -url "http://localhost:53307/CentraSite/CentraSite" ↩
-user "Administrator" -password
"manage" -assets "MyService" -file "config.xml"
```

Input Parameters

The following table describes the complete set of input parameters that you can use with the `purge Logs` utility:

Parameter	Description
`-url`	The fully qualified URL (http://localhost:53307/CentraSite/CentraSite) for the CentraSite registry/repository.
`-user`	The user ID of a user who has the "CentraSite Administrator" role.

Parameter	Description
`-password`	The password of the user identified by the parameter "-user".
`-assets`	The name or UUID assigned to the asset and which uniquely identifies it within the registry.
`-file`	The URI (file: or http:) of the configuration file.

Note: If the specified key or name of the asset to purge does not match any existing asset in the registry, then no purging is done.

7

Exporting / Importing the Purged Log Records

You can selectively export log records to a location on disk, and import the log records back to the same CentraSite installation.

Exporting the Purged Log Records from the Command Line

Additionally, you can modify the config.xml file to specify the export location of purged log records and the date range to purge log records, and execute the command line utility to export the defined log records. For example, if you specify a directory location and the date until which the log records need to be purged, then executing the command line, CentraSite deletes the log records that were generated until the specified date and exports them as archive to the specified directory.

The following table identifies the log purging behavior of CentraSite for each purge setting configuration:

Log Purge Settings	Until		Older Than	
	Export Location Specified	Export Location Not Specified	Export Location Specified	Export Location Not Specified
Date Specified	Deletes log records that are generated until the specified date and exports them as an archive to the specified directory that can be imported later.	Permanently deletes log records that were generated until the specified date from the CentraSite Log Database.	Deletes log records that are older than the specified date and exports them as an archive to the specified directory that can be imported later.	Permanently deletes log records that are older than the specified date from the CentraSite Log Database.
Date Not Specified	Deletes log records that are generated until the current date and exports them as an archive to the specified directory that can be imported later.	Permanently deletes log records that are generated until the current date from the CentraSite Log Database.	Deletes log records until the current date and exports them as an archive to the specified directory that can be imported later.	Permanently deletes log records until the current date from the CentraSite Log Database.

Viewing the Exported Purged Log Records

Perform the following steps to view the exported log records.

To view the exported log records

1. Display the Log Configuration page as described in *Configuring Logs*.
2. Go to the **CentraSite** > **Location Management** > **Initial Location** node to view all the exported log records.

Importing the Purged Log Records

Perform the following steps to import the purged log records back into the CentraSite installation.

▶ To import the purged log records back into CentraSite

1. Display the Log Configuration page as described in *Configuring Logs*.
2. Click **Browse** next to the **Import File** field, select the archive file that you want to import and click **OK**
3. In the Log Configuration page, click **Import** to start the importer.

   The imported log records are restored in the CentraSite Log Database.

8 Configuring the Purger Properties for High Volume Data Handling

Over time, a very large number of log records will accumulate in the log database, which makes the handling of log records difficult, and directly affects the performance of a log database. Therefore, it is important to purge unwanted log records from the log database occasionally.

However, purging large number of log records at one time would definitely result in system failures. In order to avoid such failures, CentraSite supports purging of the log records in a batch mode. Purging of log records in a batch mode can be initiated by enabling the `enable.partial.commit` property to `true` in the *purger.properties* file. The *purger.properties* file is located in the *logpurging/resources* folder under the CentraSite installation directory.

Note: Disabling the `enable.partial.commit=true` property would lead to failures if the system is not able to process the huge amount of log records.

Basically, the log records accumulate at a different rate for each log unit. CentraSite defines a default threshold value for batch purging each of the log units as follows:

```
##partial commit enabled/disabled
enable.partial.commit=true
##policy log threshold
policy.log.partial.commit.threshold=1000
##approval log threshold
approval.log.partial.commit.threshold=500
##events threshold
runtime.events.partial.commit.threshold=1000
##metrics log threshold
runtime.performance.log.partial.commit.threshold=1000
## No of days before the current day for which the data will not be included in the ↩
purging,
## in case of Export Log Entries & Delete Log Entries (i.e. No date criteria ↩
specified).
## purge.older.than=1
```

You can modify the threshold values for batch purging the log units as required.

Note: If the purging type in the Log Configuration page is set to **Export** or **Delete** (No date criteria), then purging would exclude the log records by the number of days defined in the `purge.older.than` property. For example, if the purging is scheduled on a **weekly** on Saturday with `purge.older.than=7`, then log records that are available a ahead of the week or 7 days would be purged from the log database.

To change the batch purging properties

1. Edit the *purger.properties* that is located in the CentraSite installation directory.
2. Enable/disable batch purging setting the `enable.partial.commit` property to `true` or `false`, as required. The default value is “true”.

3 Specify the threshold values in the `partial.commit.threshold` property of each log unit, as required.

4 Save and close the file.

The changes will take effect immediately in the next purging.

Removing Leftover Auditable Events

In previous releases of CentraSite, some auditable events remained in the log after purging, even though the events referred to CentraSite objects that no longer existed.

There is one known situation where many of these events were created: the Integration Server can be configured to write metrics information for virtual services into CentraSite at regular intervals. Each metrics update resulted in some leftover events.

If you have upgraded your CentraSite Registry Repository from a previous release, such leftover auditable events might still exist in the log. To remove these auditable events, you can use a command line tool. The tool consists of an executable jar file in the bin folder of the CentraSite installation. It requires a Java 6 runtime and needs to be called in the following way:

```
java -jar CentraSiteOptimizeAuditableEvents.jar <CentraSite DB URL> <administrator ↩
user id> <password>
```

For example:

```
java -jar CentraSiteOptimizeAuditableEvents.jar ↩
"http://localhost:53307/CentraSite/CentraSite" DOMAIN\admin pAsSwOrD
```

Note that you need the "CentraSite Administrator" role to run this tool.