IMAGE SERVICES

Activity Logging Manual

**IS 4.0 HP Integrity Edition
and IS 4.0 SP5**

9844110-003

November 2006

Contents

Category Specifics   22

Index   34

About This Manual

This manual contains information about the Activity Logging feature available with Image Services 4.0.

The contents include:

- Activity logging overview

- Activity logging configuration information

- Activity logging categories specifics

We assume you are familiar with FileNet system operations and terminology as described in the ***System Administrator's Handbook***.

New Features and Changes in This Release

This is the first release of this document.

FileNet Education

FileNet provides various forms of education. Please visit Global Learning Services on FileNet's Web site at (**www.filenet.com**).

Comments and Suggestions

FileNet invites all customers to communicate with the Documentation group on any question or comment related to FileNet manuals and on-line help. Send email to **docs@filenet.com**. We will make every effort to respond within one week. Your suggestions help us improve the products we deliver.

Overview

Image Service Activity Logging is a feature that allows for the auditing of all key system events within Image Services regardless of the database that is being updated. Activity is not just logged for documents but also for annotations, security, and other types of database updates.

Configuration Information

Description of IS Activity Logging Granularity

The IS Activity Logging has been implemented to log specific categories or events that occur on an IS Server. The System Administrator must activate this functionality or logging will not occur. The five abstracts or categories in which events can be logged are:

- Annotation Logging

- Document Table Logging - Deletion and Insertion

- Index Services Logging - Indexing and Foldering

- Security Logging - User and Group Maintenance

- WorkFlo Queue Services

The general functionality of IS Activity logging is not able to log events any more granular than these five categories.

A configuration file is read to determine what categories will be logged. As events occur, information is buffered in shared memory and eventually flushed to log files when necessary. Separate log files are created for each category. By default, new log files are created on an hourly basis or when the Image Service software is stopped and restarted.

Note	During a normal Image Services software shutdown (initfnsw, Xtask-man), event information still buffered in memory is automatically synchronized and flushed to the event files. If the software is shutdown abnormally, buffered data may be lost.

The log directory is the path where IS Activity Logging event files are located. The "Log Directory" path must be followed by a back-slash for Windows Server environments or a forward-slash for UNIX environments. By default, activity logging log files are in **/fnsw/local/logs/act/** (UNIX), or **<FNSW_drive>:\fnsw_loc\logs\act** (WIndows Server). However, this is configurable in the Activity Logging configuration file. A sample name of an activity log file might be:

Activity Logging Buffer Flushing

Two programs, **flush_act_log** and **flush_index_act_log**, are available to flush activity logging buffers.

Program	Event Types Flushed
flush_act_log	DOC, ANO, SEC
flush_index_act_log	INX, WQS

These two programs may be run any time the Image Services software is up to flush the shared memory buffers. When this is done, the current log files are renamed. This is done so no new data is appended to the current log file after it has been flushed. Once a log file has been flushed, it is renamed with the following naming convention:

<old filename>.NON_ACTIVE.<version number>

Examples of these filenames may look similar to the following:

2004021908.ANO.NON_ACTIVE.100
2004021908.DOC.NON_ACTIVE.075
2004021908.SEC.NON_ACTIVE.050

Note Only the current activity log for each category type is renamed. If the most current activity log does not match the current time, the renaming is not necessary. A version number is necessary in case multiple flushes occur within the same hour interval. The version numbers go from 000 to 999. The renaming of current log files is done when a user directly invokes either of the flush programs (flush_act_log, flush_index_act_log) or when the Image Services software is shutdown normally.

Activity Logging Configuration

To enable IS Activity Logging, you must create and configure the activity logging configuration file. This file must reside in the **/fnsw/local/sd** directory and be named **act_log_conf**.

The act_log_conf file is in ASCII text format. If the configuration file does not exist, no activity logging will occur. The system reads the configuration during system startup. The activity logging configuration information is then kept in shared memory.

For multiple server configurations, the act_log_conf file should be placed on every system where logging should occur. The best practice is to have the same copy of the act_log_conf file on every server. For example, on a dual server system consisting of a Root/Index server and a Storage Library server, the security and index services activity will only be logged on the Root/Index server, while the Docs table and annotation activity will only be logged on the Storage Library server.

The following table describes the tokens assigned to each category or abstract. These tokens will always appear in column one of the configuration file:

Token or Abstract	Description
ANO	For annotation logging
DOC	For Docs Table logging
INX	For Index Services logging
SEC	For security logging
WQS	For WorkFlo Queue Services logging

File Heading Information

Every configuration file has to have heading information. If column one has the characters **#**, **0**, or **\n** that line will be skipped. The file should begin with a Log Directory section in a comment line and then a line in the file which specifies the location of the log file:

```
cm-boulders(fnsw)/fnsw/local/sd> cat act_log_conf
#
#     Log Directory
#
directory=/fnsw/local/logs/act/

#
```

If the token **directory** (not case sensitive) starts in column one, it needs to be followed first by a blank (directory /fnsw) or an = sign (directory=/fnsw) and then by an absolute path. If any directories in the path do not exist, an attempt will be made to create them when the buffer is flushed to a physical log file. The directory token is required for

activity logging, and if it does not exist in the act_log_conf file, a warning message is logged.

Every section that follows for the various tokens you want to include in the file (ANO, SEC, etc.) should have a comment line describing the token setting followed by the actual configuration setting for that token.

Annotation Logging (ANO)

To manually configure the ANO abstract, you need to enter the following into the configuration file:

```
#
#      Annotion setting
#
ANO 1024
```

In the Annotation Setting area, you need to enter the ANO token and the size of the in memory buffer in kilobytes.

• If the buffer size is not specified, a default value of 4,500 bytes is used (average entry size is about 45 bytes).

• If the buffer size is set to 0 bytes, the writes to the activity log files is a synchronous flush to the log file.

• If the buffer exceeds 32 KB (32768 bytes), the buffer size is set to 32 KB. The maximum value for each shared memory buffer is 32 KB.

• The ANO abstract does not support token qualifiers.

Go to for more information.

Document Table Logging (DOC)

To manually configure the DOC abstract, you need to enter the following into the configuration file:

```
#
#     DOC create/delete setting
#
DOC 4000 create delete
```

In the DOC create or DOC delete setting area, you need to enter the DOC token and the size of the in memory buffer in kilobytes. You also need to specify DOC token qualifiers after the buffer size indicator. These token qualifiers are create or delete.

- If the buffer size is not specified, a default value of 4,500 bytes is used (average entry size is about 45 bytes).

- If the buffer size is set to 0 bytes (as in the example), the writes to the activity log files is a synchronous flush to the log file.

- If the buffer exceeds 32 KB (32768 bytes), the buffer size is set to 32 KB. The maximum value for each shared memory buffer is 32 KB.

Note IS Activity Logging allows you to separately configure document Creates and document Deletes.

Go to **"DOC" on page 24** for more information.

Index Logging (INX)

To manually configure the INX abstract, you need to enter the following into the configuration file:

```
#
#      INX folder and dir setting
#
INX 4000 dir folder
```

In the INX folder and dir setting area, you need to enter the INX token and the size of the in memory buffer in kilobytes. You also need to specify INX token qualifiers dir and or folder after the buffer size indicator.

In memory Buffer Size Settings

The buffer size particulars are the same as with the other tokens:

• If the buffer size is not specified, a default value of 4,500 bytes is used (average entry size is about 45 bytes).

• If the buffer size is set to 0 bytes (as in the example), the writes to the activity log files is a synchronous flush to the log file.

• If the buffer exceeds 32 KB (32768 bytes), the buffer size is set to 32 KB. The maximum value for each shared memory buffer is 32 KB.

Token Qualifier Settings for DIR logging

When specifying INX <memory_buffer_size> dir, all **dir** functionality is logged.

- This following table includes the token qualifiers and sub-token qualifiers for the **INX** token.

Token Qualifier	Sub-Token Qualifiers	
dir	dir_create	dir_close
	dir_update	dir_import
	dir_delete	dir_import_loc

- Much more selective and finer granularity options are offered with these sub-token possibilities because they allow for the individual **INX dir** functions to be logged separately. If you have more than one type of dir sub-token qualifier, you simply specify as in the following example:

```
#
#     INX folder and dir setting
#
INX 4000 dir_create dir_close
```

- The dir imports are also logged into two subtypes (dir_import and dir_import_loc) to indicate where the import originated from. dir_import_loc entries are different from dir_import entries because they use the F_DOC_LOCATION field. If the document index record being imported has an empty F_DOC_LOCATION field, it is referred to as a dir_import_loc entry instead of a dir_import entry. For example, Report Manager Cross Cataloging uses the F_DOC_LOCATION field. By logging dir_import_loc, you can distinguish between regular dir entries and Report Manager Cross Cataloging entries.

- The activity of unspecified tokens is not logged.

• If the dir token qualifier is used in conjunction with other dir sub-to-ken qualifiers, all dir functions are logged. A warning is generated during the parsing of the configuration file to let the user know their meaning may have been overwritten by the dir token

Go to **"INX" on page 26** for more information.

Security Logging (SEC)

To manually configure the SEC abstract, you need to enter the following into the configuration file:

```
#
#      Security setting
#
SEC 0
```

In the Security Setting area, you need to enter the SEC token and the size of the in memory buffer in kilobytes.

• If the buffer size is not specified, a default value of 4,500 bytes is used (average entry size is about 45 bytes).

• If the buffer size is set to 0 bytes (as in the example), the writes to the activity log files is a synchronous flush to the log file.

• If the buffer exceeds 32 KB (32768 bytes), the buffer size is set to 32 KB. The maximum value for each shared memory buffer is 32 KB.

• The SEC abstract does not support token qualifiers.

IS Activity logging logs Security creates, updates and deletes for all User and Group changes. It is also suggested that the buffering be set to zero (0), which ensures all Security events are included at every interval.

Note   Any changes through IS Security Administration to Security "Devices", "Functions" or User passwords are NOT logged by the IS Activity logging.

Go to **"SEC" on page 29** for more information.

WorkFlo Queue (WQS)

To manually configure the WQS abstract, you need to enter the following into the configuration file:

```
#
#      WorkFlo Queue setting
#
WQS 1024
```

In the WorkFlo Queue setting area, you need to enter the WQS token and the size of the in memory buffer in kilobytes.

• If the buffer size is not specified, a default value of 4,500 bytes is used (average entry size is about 45 bytes).

• If the buffer size is set to 0 bytes (as in the example), the writes to the activity log files is a synchronous flush to the log file.

- If the buffer exceeds 32 KB (32768 bytes), the buffer size is set to 32 KB. The maximum value for each shared memory buffer is 32 KB.

- The WQS abstract does not support token qualifiers.

Note   Currently only WorkFlo Queues are supported queue structures for IS Activity Logging. These queues should not be confused with the Workflow Queues found in the Visual Workflow or the eProcess Products. Those structures are not supported at this time.

Go to **"WQS" on page 31** for more information.

File Ending Information

Every configuration file can end with a comment line noting the end of the file. This is not required.

```
# End of activity log configuration file
```

IS Activity Logging Events

The following is the input required for all of the possible events to be logged through IS Activity Logging as shown in the activity log configuration file **act_log_conf**:

```
cm-boulders(fnsw)/fnsw/local/sd> cat act_log_conf
#
#     Log Directory
#
directory=/fnsw/local/logs/act/

#
#     Annotion setting
#
ANO 1024

#
#     WorkFlo Queue setting
#
WQS 1024

#
#     INX dir/folder setting
#
INX 4000 dir folder

#
#     DOC create/delete setting
#
DOC 4000 create delete

#
#     Security setting
#
SEC 0

#  End of activity log configuration file
```

Basic Log Content

All of the logs created, regardless of category, have basic similarities in file naming convention, and date and time stamp information.

Log File Naming Convention

Each category or abstract has a separate set of log files that is distinguished by the suffix. The suffixes for the categories are the same as shown in the table on **page 11**. The naming convention is described below:

```
<year><month><day><hour>.[ANO|SEC|INX|WQS|DOC]
year = 4 digit representation
month = 2 digit representation (e.g. Jan = 01)
day = 2 digit representation
```

The following display shows possible examples with the date and hour stamp in bold:

```
2004021908.ANO
2004021910.SEC
2004021914.WQS
2004021920.DOC
2004021923.INX
```

For instance, in the first example above, the ANO file was created on February 19, 2004 at 8:00 A.M.

By default, a new file is created hourly if activity has occurred and flushing is needed.

Content of date and time stamp

Each activity entry within a log file has a date and time stamp with the following format:

```
<year>/<month>/<day> <hour>:<min>:<sec>.<milsec>

year = 4 digit representation
month = 2 digit representation (e.g. Jan = 01)
day = 2 digit representation
hour = 2 digit representation (military time)
min = 2 digit representation
sec = 2 digit representation
milsec = 3 digit representation
```

The following display shows possible examples with the date and time stamp in bold:

```
2004/02/19 08:55:22.521 create user newuser:milo:FileNet
2004/02/19 08:55:23.307 update user newuser:milo:FileNet
2004/02/19 10:44:35.845 rename user newuser:milo:FileNet newuser1:milo:FileNet
2004/02/19 21:57:45.838 create group newgroup:milo:FileNet
```

For instance, in the first example above, the create user occurred on February 19, 2004 at 8:55 and 22.521 seconds A.M.

Category Specifics

ANO

Description

Annotation logging - Logs the creation, update and deletion of all annotations on documents.

ANO Log File Entry Syntax

<date_time stamp> {create | update | delete} <doc_id> <page number> <annot_id> <process_id>

or

<date time_stamp> delete_all <doc_id> who

<date_time stamp>	Logs date and time of the creation of the log entry.
create | update | delete	Logs the creation, deletion or update of an annotation.
<doc_id>	The document ID of the document with the annotation.
<page number>	The page number in the document that has the annotation.
<annot_id>	The ID number assigned to the annotation. They are assigned sequentially starting with one.
delete_all	Logs the deletion of all of the annotations in the document.

<process_id>   Logs who made the annotation to the document. This is appended to all ANO log entries at the end of the line in the form of user_name:domain:organization.

Note   The **<process_id>** parameter is currently only available for the Annotation abstract with IS 4.0 SP1, but will be added to all of the logging categories in a later release of Image Services.

Note   In some cases, the user name is **ServiceProcess:System:System**. When using deldocs to delete a document with an annotations or importing a document with annotations, the user name in the annotation activity log file is ServiceProcess:System:System.

"Viewing an annotation" is NOT a logged event.

Sample Output

The following sample output is an example of an Annotation log file:

```
2000/09/12 15:36:00.423 delete_all 2000004767 SysAdmin:happy:FileNet
2000/09/12 15:36:00.424 delete_all 2000004765 SysAdmin:happy:FileNet
2000/09/12 15:36:00.452 delete_all 2000004763 SysAdmin:happy:FileNet
2000/09/12 15:36:00.453 delete_all 2000004761 SysAdmin:happy:FileNet
2000/09/12 15:39:01.037 create 2000004769 1 1 SysAdmin:happy:FileNet
2000/09/12 15:39:01.063 create 2000004769 1 2 SysAdmin:happy:FileNet
2000/09/12 15:39:01.086 create 2000004769 1 3 SysAdmin:happy:FileNet
2000/09/12 15:39:27.536 delete_all 2000004769 SysAdmin:happy:FileNet
2000/09/12 15:49:49.389 create 2000004770 1 1 SysAdmin:happy:FileNet
2000/09/12 15:49:49.417 create 2000004770 1 2 SysAdmin:happy:FileNet
2000/09/12 15:50:11.337 delete_all 2000004770 SysAdmin:happy:FileNet
```

DOC

Description

Document Services Logging - Logs the creation and deletion of documents in the system.

DOC Log File Entry Syntax

<date_time stamp> {delete | create} <doc_id> <orig_ssn> <orig_doc_id>

<date_time stamp>   Logs the Date and time of the creation of the log event.

delete | create   Document create or delete actions are logged by any document added to or removed from an IS. These can be initiated via IDM Desktop, IDM Web, any WAL application or Fast Batch process, Import, etc

<doc_id>   The Doc ID of the document being created or deleted.

<orig_ssn>   The system serial number of the system the Doc ID is being either imported from during a create or deleted from during a delete.

<orig_doc_id>   The original Doc ID of the document that is being either imported during a create or deleted during a delete.

Sample Output

The following sample output is an example of a Document Services log file:

```
2000/09/12 15:21:42.036 create 2000004761 0 0
2000/09/12 15:21:42.093 create 2000004763 0 0
2000/09/12 15:27:27.022 create 2000004765 0 0
2000/09/12 15:27:27.086 create 2000004767 0 0
2000/09/12 15:36:00.422 delete 2000004767 2002 2000004767
2000/09/12 15:36:00.423 delete 2000004765 2002 2000004765
2000/09/12 15:36:00.424 delete 2000004763 2002 2000004763
2000/09/12 15:36:00.453 delete 2000004761 2002 2000004761
2000/09/12 15:37:32.965 create 2000004769 0 0
2000/09/12 15:39:27.534 delete 2000004769 2002 2000004769
2000/09/12 15:49:03.913 create 2000004770 0 0
2000/09/12 15:50:11.336 delete 2000004770 2002 2000004770
```

INX

Description

Index Service Logging - Logs any manipulation of the document or folder properties.

INX Log File Entry Syntax

<date_time stamp> dir {create | import | update | delete | close | import_loc} <doc_id>

or

<date time_stamp> folder {create | update | close} <name>

or

<date time_stamp> folder delete <name> <unfile documents TRUE=1/FALSE=2>

or

<date time_stamp> folder {move | copy} <source name> <target name>

or

<date time_stamp> folder rename <old name> <new name>

or

<date time_stamp> folder {file | unfile} <name> <doc_id>

or

<date time_stamp> folder {file_after | reorder} <name> <first docid> <num docids> <after docid>

<date_time stamp>   Logs date and time of the creation of the log event.

dir {create | import | update | delete | close | import_loc}   Document Index Record create events are logged when a Document create is logged. DIR update events are logged when Index values are changed as well as document attributes. DIR import events are logged when documents are imported into an IS via COLD and HPII. DIR

import_loc events are logged when Report Manager Cross Cataloging takes place. DIR delete events are logged when a document is deleted from the FILENET system.

folder {create | update | close}
Any folders that are created or have their attributes changed cause a folder create or folder update action to be logged.

folder {move | copy}
Any folders that are moved or copied cause a folder move or folder copy to be logged.

folder {file | unfile}
This log entry appears when a document (with a specific doc_id) is removed from or placed in a folder.

folder {file_after | reorder}
The file after and reorder logging is complicated. The entry point INX_file_doc_after allows for an array of doc_ids to be filed after a specified doc_id. The specified doc_id is assumed to already be in the folder. The entry point INX_reorder_folder allows for an array of doc_ids, which have already been filed, to be reordered in a folder after a specified doc_id. These two entry points are not used in native mode (IDM Desktop). These entry points are supported IS Toolkit (WAL) entry points but are probably not widely used.

folder delete
A deleted folder causes a folder delete action to be logged.

folder rename
A folder has been renamed.

<doc_id>
The document ID of the document being inserted.

<name>
The name of the folder whose attributes are changing.

<source name>
The name of the folder before it is moved or copied.

<target name>
The name of the folder after it is moved or copied.

<old name>
The name of the folder before it was changed.

<new name>	The name of the folder after the change.
<unfile documents TRUE=1/ FALSE=2>	This logs whether or not you are going to unfile documents during a folder delete.
<first docid>	This logs the first document in a specific folder when reorganizing files in a folder
<num docids>	This logs intermediate document(s) in a specific folder when reorganizing files in a folder.
<after docid>	This logs the last document in a specific folder when reorganizing files in a folder.

Sample Output

The following sample output is an example of an Index Services log file:

```
2000/09/12 15:27:27.293 dir create 2000004765
2000/09/12 15:27:27.326 dir create 2000004767
2000/09/12 15:28:22.932 folder create /folder10
2000/09/12 15:30:44.930 folder file /folder10 2000004767
2000/09/12 15:32:08.024 folder rename /folder10 /folder11
2000/09/12 15:33:01.722 folder delete /folder11 1
2000/09/12 15:33:48.645 folder create /folderold
2000/09/12 15:34:07.421 folder copy /folderold /foldernew
2000/09/12 15:36:00.266 dir delete 2000004767
2000/09/12 15:36:00.266 dir delete 2000004765
2000/09/12 15:36:00.266 dir delete 2000004763
2000/09/12 15:36:00.266 dir delete 2000004761
2000/09/12 15:37:33.033 dir create 2000004769
2000/09/12 15:39:27.420 dir delete 2000004769
2000/09/12 15:49:04.202 dir create 2000004770
2000/09/12 15:50:11.229 dir delete 2000004770
```

SEC

Description

Security Logging - Logs the creation, update, and deletion of security objects such as users and groups.

SEC Log File Entry Syntax

<date_time stamp> {create | import | update | delete} {user | group} <object>

<center>or</center>

<date time_stamp> rename {user | group} <old object> <new object>

<date_time stamp> Logs date and time of the creation of the log event.

create | import | update | delete | re-name Logs the action performed on the user of group to follow.

user | group The creation or deletion of a "user" causes an event to be logged. Changing the attributes of an existing user causes a "user update" to be logged. When the "add user" or "edit user" functionality is used to add a user to a group, the IS Activity Logging logs a "user create" or "user update" as well as a "group update".

The creation or deletion of a "group" causes an event to be logged. Changing the attributes of an existing group causes a "group update" to be logged. When the "add group" or "edit group" functionality is used to add a user to a group, the IS Activity Logging logs a "group create" as well as a "group update".

<object>   This is the <name>:<domain>:<organization> of the system. The <name> field on NT/Win2000 may include DOMAIN/USERNAME information.

<old object>   Old name of the user or group that changed.

<new object>   New name that the user or group changed to.

Sample Output

The following sample output is an example of an Security log file:

```
2000/09/12 15:57:22.521 create user newuser:milo:FileNet
2000/09/12 15:57:44.307 update user newuser:milo:FileNet
2000/09/12 15:58:36.845 rename user newuser:milo:FileNet newuser1:milo:FileNet
2000/09/12 15:59:00.838 create group newgroup:milo:FileNet
2000/09/12 15:59:31.609 update group newgroup:milo:FileNet
2000/09/12 15:59:47.823 rename group newgroup:milo:FileNet newgroup1:milo:FileNet
2000/09/12 16:00:00.901 delete group newgroup1:milo:FileNet
2000/09/12 16:00:09.509 delete user newuser1:milo:FileNet
```

Note   The local date/time stamp do not always match the date and hour stamp in the activity log. In the example above, the first 6 entries don't match. This happens because of buffering. The date/time stamp is generated shortly after the event. When the shared memory buffer is flushed, it is written to the current activity log.

WQS

Description

WorkFlo Queue Logging - Logs the creation, update, and deletion of WorkFlo Queues and Workspaces. Also logs queue activity such as **move**, **rename**, and **empty**.

WQS Log File Entry Syntax

<date_time stamp> queue {create | delete | update} <queue_name> <workspace> <service>

or

<date_time stamp> queue empty <queue_name> <workspace>

or

<date_time stamp> queue rename <old_queue_name> <new_queue_name> <workspace> <service>

or

<date_time stamp> queue move <old_queue_name> <old_workspace> <new_queue_name> <new_workspace> <service>

or

<date_time stamp> workspace {create | delete | update} <workspace> <service>

or

<date_time stamp> workspace rename <old_workspace> <new_workspace> <service>

<date_time stamp>	logs the date and time of the creation of the log event.
queue	Logs a "queue" create, delete or update.
queue empty	Logs the emptying of a specific queue within a specific workspace.

queue rename	Logs the renaming of a queue.
queue move	Logs the moving of a queue to from an old workspace to a new one.
workspace	Logs a specific workspace create, delete or update.
workspace rename	Logs the renaming of a workspace.
create | delete | up-date	The creation of a queue causes a "create queue" to be logged. Any updates on an existing queue causes a "update queue" to be logged. When a queue is deleted, the IS Activity Logging logs a "delete queue". It is important to remember that queue changes are bound by the same IS functionality restrictions as workspace changes. Please refer to the Image Services documentation for more information on these restrictions. Should errors result in the attempted operations the action will not be logged. The creation of a workspace causes a "create workspace" to be logged. Any other updates on an existing workspace or deletes create "update workspace" or "delete workspace". However, any changes attempted on an existing workspace with queues are bound by IS functionality restrictions and will not be logged should errors result in the attempted operation.
<queue_name>	Name of queue (for example: move, rename, empty)
<old_queue_name>	Name of old queue during the moving or renaming of a queue.
<new_queue_name>	Name of new queue during the moving or renaming of a queue.
<workspace>	Name of the workspace during a queue update, delete, rename, etc.
<old_workspace>	Name of the old workspace during the renaming of a workspace.

<new_workspace>   Name of the new workspace during the renaming of a workspace.

<service>   Equals <name>:<domain>:<organization>

Sample Output

The following sample output is an example of an WorkFlo Queue Services log file

```
2001/03/01 11:17:32.379 workspace create act_log WflServer:moosehead:FileNet
2001/03/01 11:18:08.059 workspace update act_log WflServer:moosehead:FileNet
2001/03/01 11:18:28.905 queue create Queue1 act_log WflServer:moosehead:FileNet
2001/03/01 11:23:28.532 queue update act_log act_log WflServer:moosehead:FileNet
2001/03/01 11:24:04.734 queue delete act_log act_log WflServer:moosehead:FileNet
2001/03/01 11:24:08.250 workspace delete act_log WflServer:moosehead:FileNet
```

Index