

How to configure logging and logs rotation in FIXEdge



- [Overview](#)
- [Configuring engine.properties file](#)
- [Configuring FIXEdge.properties file](#)
- [Configuring logs rotation](#)
 - [Log.File.Rotate](#)
 - [Log.File.Rotate.SizeLimit](#)
 - [Log.File.Rotate.FileLimit](#)
 - [Log.File.Rotate.Dir](#)
- [Division by categories and severities](#)
- [Best practices and commonly used cases](#)
 - [engine.properties logging parameters](#)
 - [FIXEdge.properties logging parameters](#)




Overview

FIXEdge provides the possibility to set up logging and logs rotation by configuring **engine.properties** and **FIXEdge.properties** files.

Configuring *engine.properties* file

The following parameters related to logging can be set up in *engine.properties* file:

Parameter	Description	Example
Engine parameters		
LogFileName	Engine's log file name. If this parameter is commented or empty, the "engine.log" will be used.	LogFileName = FIXAntenna_C++.log
LogDirectory	This property is the path of the directory in which the logs for all incoming (if LogIncomingMessages is set to "true") and outgoing FIX messages are stored. It is possible to specify a path related to the EngineRoot directory. For example if LogDirectory is set to "logs\" then the real path is \$(EngineRoot)/logs. The specified directory must exist.	LogDirectory = logs
Session parameters		
Debug.LogSessionExtraParameters	<p>If this property value is true, general and extra session parameters of primary and backup connections will be printed to the log.</p> <p>If this property value is false, only general session parameters of primary and backup connections will be printed to the log.</p> <p>All FIXEdge specific session parameters will be printed to the log regardless of Debug.LogSessionExtraParameters property value.</p>	Debug.LogSessionExtraParameters = true
Session parameters: Storages		
LogIncomingMessages	<p>This property provides an option to log incoming FIX messages (those received) from a counterparty FIX Engine. They will be stored in the directory specified by the LogDirectory parameter in a file with extension "in".</p> <div style="border: 1px solid red; padding: 5px; margin-top: 10px;">  Changing this value will impact upon the performance of FIX Engine. In the development should be true. In production should be false. </div>	LogIncomingMessages = true
TimestampsInLogs	<p>An option to write timestamps in the log files. This parameter is optional, the default value is true.</p> <div style="border: 1px solid red; padding: 5px; margin-top: 10px;">  Changing this value will impact upon the performance of FIX Engine. In the development should be true. In production should be false. </div>	TimestampsInLogs = true
EnableIncrementalLogFileCreation	<p>Setting property "true" will reserve 10Mb of disc space for logging.</p> <p>If log reaches 10Mbs, another 10Mbs will be reserved and so on. When property is "true" FixAntenna performance is greatly increased.</p>	EnableIncrementalLogFileCreation = true

Persistents.LogIncrementSize	This property specifies amount of memory reserved for memory mapped file message storage for out/in file. <div style="border: 1px solid red; padding: 5px;"> Changing this value will impact upon the performance of FIX Engine. Large value is better.</div>	Persistents.LogIncrementSize = 1000000
Persistents.IdxIncrementSize	This property specifies amount of memory reserved for memory mapped file message storage for idx file. <div style="border: 1px solid red; padding: 5px;"> Changing this value will impact upon the performance of FIX Engine. Large value is better.</div>	Persistents.IdxIncrementSize = 10000
SplitPersistentMsgStorage.MaxSliceSize	This property specifies maximum size of one slice for sliced message storage. Note: Storage slices are created with step, specified with Persistents.LogIncrementSize parameter value. If MaxSliceSize value is no multiple of Persistents.LogIncrementSize, result file size will be less, than specified. If value is less than LogIncrementSize, LogIncrementSize value will be used. Recommended value: any multiple of Persistents.LogIncrementSize Default value is 500000000 - 500Mb.	SplitPersistentMsgStorage.MaxSliceSize = 500000000
SplitPersistentMsgStorage.BackupDir	This property is the path of the directory, where sliced message storage will search for log files in case they not found in primary location. It is user responsibility to move files from primary location to backup. Default value is empty - backup path disabled.	SplitPersistentMsgStorage.BackupDir = logs/BackupDir
MessageStorage.IndexRebuildEnabled	Optional. If this parameter is true, FIX Engine will try to restore the index file from the log file in case the index file is missed or corrupted. The default value is false. <div style="border: 1px solid red; padding: 5px;"> Please be careful when you enable this property: restoring process uses some heuristic algorithms, and some data (not messages) could be missed.</div>	MessageStorage.IndexRebuildEnabled = false
Session parameters: Unregistered Acceptors parameters		
UnregisteredAcceptor.SessionStorageType	Optional. Commented. Defines the storage type for unregistered sessions . <i>Valid values:</i> "persistent" "persistentMM" "splitPersistent" "transient" "null"	UnregisteredAcceptor.SessionStorageType = persistent
Backup parameters		
BackupDirectory	Relative path to the backup folder. This folder will be used for message storage files of the backup connections.	BackupDirectory = logs/backup
HiddenLogonCredentials	Optional. Provides an option to mask password in in/out log files. <i>Valid values:</i> "true" "false". <i>Default value:</i> HiddenLogonCredentials = false	HiddenLogonCredentials = true
Logging parameters: refer to Logging parameters page for details.		

Logging parameters of the backup sessions


If the backup parameters of the session are the same as the primary parameters of the session then Host, Port, HeartbeatInterval, and INFO message "The backup session has the same parameters as the primary session" will be printed in the log.

If at least one backup parameter of the session differs from the primary one, then Host, Port, HeartbeatInterval, and **all the backup parameters** of the session will be printed in the log.

Configuring *FIXEdge.properties* file

The following parameters related to logging can be set up in *FIXEdge.properties* file:

Parameter	Description	Example
-----------	-------------	---------

FixLayer.FixEngine.Sessions.ArchivePath	<p>Defines the path to move log files after the session is closed.</p> <p>Note: directory must exist otherwise the backup procedure will fail.</p> <div style="border: 1px solid red; padding: 5px; margin-top: 10px;">  It shouldn't point to path from BackupDirectory (defined in engine. properties), otherwise error message will occur on session start, and session won't start </div>	FixLayer.FixEngine.Sessions.ArchivePath = FixEdge1/log/archive
FixLayer.FixEngine.Session.Session_Name.StorageType	<p>Mandatory. Defines the session storage type.</p> <p><i>Valid values:</i> "persistent" "persistentMM" "splitPersistent" "transient" "null"</p>	FixLayer.FixEngine.Session.TestFIXAcceptor.StorageType = persistentMM
Logging parameters: refer to Logging parameters page for details.		

If you want FIXEdge to store logs in system logs or print it to the console you should change [Log.Device](#) property:

Accepted values:

- File - messages will be written to the file.
- Console - messages will be written to the console.
- EventLog - messages will be written to the Windows Event Log (Windows specific).
- WinDebug - messages will be written to the OutputDebugString (Windows specific)
- Syslog - messages will be written to the system log (Linux/Unix specific).

It also could be configured per category.



Please see [How to divide different categories and severities of log files into different files in the Logging section](#) for details.

Configuring logs rotation

In order to set up logs rotation use the following parameters in the FIXEdge.properties file:

Log.File.Rotate

Enables/disables log file rotation.

Valid values:

- true - to enable rotation;
- false - to disable rotation.

Log.File.Rotate.SizeLimit

Size of file in bytes. When log file reaches this size it is renamed (follow standard rotation approach) and the new file is created.

Example: Log.File.Rotate.SizeLimit = 104857600

Log.File.Rotate.FileLimit

A number of files to be involved in the rotation. When the number is exceeded the older file is removed.

Example: Log.File.Rotate.FileLimit = 10

Log.File.Rotate.Dir

Rotation directory. Directory to store rotated files.

Example: Log.File.Rotate.Dir = <%SERVICE_NAME%>/log

Division by categories and severities

Please see [How to divide different categories and severities of log files into different files in the Logging section](#) for details.

Best practices and commonly used cases

Below are examples of client's logging configuration which are set up for production.

engine.properties logging parameters

```
# The FIX Engine configuration file.
#
# $Revision: FA 2.8 $
# For UNIX platforms, the filename-separator character is "/".
# For Win32 platforms, the filename-separator character is "\\\" or "/".
#####
# Engine settings
#####
# The top of the directory tree under which the engine's configuration,
# and log files are kept.
...
# Engine's log file name.
# If this parameter is commented or empty, the 'engine.log' will be used.
# LogFileName = FIXAntenna_C++.log
...
# Default storage type of the created unregistered sessions. By default persistent storage
# type used. Use "transient" value to use transient storage for the sessions.
# UnregisteredAcceptor.SessionStorageType =
...
# This property is the path of the directory in which the logs for all incoming
# (if LogIncomingMessages is set to "true") and outgoing FIX messages are stored.
# It is possible to specify a path related to the EngineRoot directory.
# For example if LogDirectory is set to "logs\" then the real path is $(EngineRoot)/logs.
# The specified directory must exist.
LogDirectory = FIXOEGW1/log
# Relative path to the backup folder. This folder will be used for message storage files
# of the backup connections.
BackupDirectory = FIXOEGW1/log/backup
#####
# Session settings
#####
# This property provides an option to log incoming FIX messages (those received) from
# a counterparty FIX Engine. They will be stored in the directory specified by
# the LogDirectory parameter in a file with extension "in".
LogIncomingMessages = true
...
#####
# Validation
#####
...
#####
...
# An option to write timestamps in the log files.
# This parameter is optional, the default value is true.
TimestampsInLogs = true
...
# Setting property "true" will reserve 10Mb of disc space for logging.
# If log reaches 10Mbs, another 10Mbs will be reserved and so on.
# When property is "true" FixAntenna performance is greatly increased.
EnableIncrementalLogFileCreation = false
# Setting property "true" will enable logging session parameters on creation.
Debug.LogSessionExtraParameters = true
#####
# Engine monitoring
#####
...
#####
# Engine measuring
#####
...
#####
# Logger settings
#####
# Default target devices
#
# Valid values:
# * File - messages will be written to the file.
# * Console - messages will be written to the console.
```

```

# * EventLog (Windows specific) - messages will be written to the Windows Event Log.
# * WinDebug (Windows specific) - messages will be written to the OutputDebugString.
# * Syslog (Linux/Unix specific) - messages will be written to the system log.
#
# The properties of the device are in the "Log.<name>" section. For instance Log.File contains
# all properties of the File device.
Log.Device = File
# Enable Debug messages.
# Valid values:
# * true - to enable Debug messages
# * false - to disable Debug messages
Log.DebugIsOn = true
Log.TraceIsOn = true
# Enable Note messages.
# Valid values:
# * true - to enable Note messages
# * false - to disable Note messages
Log.NoteIsOn = true
# Enable Warning messages.
# Valid values:
# * true - to enable Warning messages
# * false - to disable Warning messages
Log.WarnIsOn = true
# Enable Error messages.
# Valid values:
# * true - to enable Error messages
# * false - to disable Error messages
Log.ErrorIsOn = true
# Enable Fatal Error messages.
# Valid values:
# * true - to enable Fatal Error messages
# * false - to disable Fatal Error messages
Log.FatalIsOn = true
# Enables/disables log file cycling
# Valid values:
# * true - to enable cycling
# * false - to disable cycling
Log.Cycling = false
#####
# Logger settings: File device
#####
# Timezone of the log message timestamp
# Valid values:
# * Local - display time as local
# * UTC - display time as UTC
Log.File.TimeZone = UTC
# Root directory for log files. It will be concatenated with
# each file name specified by 'Log.File.Name'
Log.File.RootDir = ./logs
# File name. If more than one category uses files with the same name
# the same file will be used simultaneously.
Log.File.Name = _log
# Separated log for 'Version' category
Log.Version.File.Name = version.log
# If set to true then lock file <Log.File.RootDir/Log.File.Name>.lock will be created. This
# will prevent other instance of the application use the same file.
#
# Valid values:
# * true - lock log file
# * false - do not lock log file
Log.File.Locked = false
# Enables/disables log file rotation
# Valid values:
# * true - to enable rotation
# * false - to disable rotation
Log.File.Rotate = false
# Local time when logs should be backed up. Should be in format 'hh:mm'
# or 'hh:mm:ss'.
# Optional.
Log.File.Backup.Time =
# If true then file will be recreated on each start.

```

```

# If false then new records will be appended to the existent file.
Log.File.Recreate = true
# If set to true then buffer will be flushed after each logging call.
# If set to false then flush is not called.
#
# Setting to true decreases program performance; setting to false increases
# a risk of records loss in case of program failure.
Log.File.AutoFlush = true
# String that specifies the name of the source. The source name must be a
# subkey of a logfile entry under the EventLog key in the registry. For
# example, "WinApp" is a valid source name if the registry has the following
# key: HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Event Log\Application\WinApp
Log.EventLog.EventSource = TestService
# Specifies amount of memory reserved for memory mapped file message storage for out/in file
# WARNING: Changing this value will impact upon the performance of FIX Engine.
# Large value is better.
Persistents.LogIncrementSize=1000000
# Specifies amount of memory reserved for memory mapped file message storage for idx file
# WARNING: Changing this value will impact upon the performance of FIX Engine.
# Large value is better.
Persistents.IdxIncrementSize=10000

```

FIXEdge.properties logging parameters

```

# The FIX Server configuration file.
#
# $Revision: 1.5.2.3 $
# The top of the directory tree under which the FIXEdge's configuration,
# and log files are kept.
#-----
# Control Centre Section
#-----
...
#-----
# Business Layer Section
#-----
...
#-----
# FIX Layer Section
#-----
# This parameter determines the names of FIX session-acceptors. For each
# such a session it is necessary to define the parameters described below.
# The format is 'session.X.ParameterName' where 'X' is the name of session.
FixLayer.FixEngine.Sessions = FIXEDGE-CONTROL, TEST001, ...
# Path to move log files after the appointed session was closed.
# Warning: directory must exist otherwise the backup procedure will fail.
# Note: Optional.
FixLayer.FixEngine.Sessions.ArchivePath = /log/FIXEdge1/logs/archive
...
#-----
# Test FIX 4.4 sessions
#-----
FixLayer.FixEngine.Session.TEST001.Version = FIX44
FixLayer.FixEngine.Session.TEST001.StorageType = persistentMM
...
#-----
# Transport Layer Section
#-----
...
#-----
# The SMTP Adaptor configuration file.
#-----
...
#-----
# Default log settings section
#-----
# Target directory.
# Please specify absolute directory path for NT Service application.

```

```

Log.File.RootDir = /log/FIXEdge1/logs
# Target devices
Log.Device = File
# Turns on/off logging on the debug level
Log.DebugIsOn = false
# Turns on/off logging on the trace level
Log.TraceIsOn = false
# Turns on/off logging on the notice level
Log.NoteIsOn = true
# Turns on/off logging on the warning level
Log.WarnIsOn = true
# Turns on/off logging on the error level
Log.ErrorIsOn = true
# Turns on/off logging on the fatal level
Log.FatalIsOn = true
# Turns cycling on/off.
Log.Cycling = false
# Number of repeating records to be placed to log before cycling is started.
Log.Cycling.Ignore = 3
# Number of repeating records to be accumulated (hidden) before write the
# "cycle record" to the log.
Log.Cycling.BlockSize = 10
# Multiplier for the Block Size. If BlockSize number of messages is
# accumulated and the same message still appears then next BlockSize
# is calculated as the previous one multiplied by Multiplier.
Log.Cycling.Multiplier = 10
# File name. If more than one category uses files with the same name
# the same file will be used simultaneously.
Log.File.Name = FixEdge.log
# Separated log for 'Version' category. Not required
Log.Version.File.Name = version.log
# Separated log for level 'Trace' in 'CCAdminApplication' category. Not required.
Log.CCAdminApplication.Trace.File.Name = FixEdge_CCAdminApplication_Trace.log
Log.CCAdminApplication.TraceIsOn = false

# Custom format of log file record.
# Can be defined separately for each log file e.g. Log.Version.File.Format = ...
# Format string can contain any symbols and keywords:
# %%          - put '%' symbol
# %level     - put severity level name: 'ERROR', 'WARN' etc.
# %tablevel  - put extra spaces for vertical alignment of records after %level
# %date{FIX} - date in 'YYYYMMDD-HH:MM:SS.sss' format
# %date{ISO8601} - date in 'YYYY-MM-DD HH:MM:SS,sss' format
# %date      - date in 'DD MMM YYYY HH:MM:SS,sss' format
# %timezone  - time zone abbreviation or name (setup TZ env variable to take abbreviation in windows) or
'UTC'
# %logger    - log category, component
# %thread    - thread id
# %message   - logged message
Log.File.Format = %date{ISO8601} %timezone %level%tablevel [%logger] %thread %message
#Log.File.Format = [%level] %tablevel%date{FIX} [%thread] [%logger] - %message
# Timezone of the log message timestamp
# Valid values:
# * Local - display time as local
# * UTC - display time as UTC
Log.File.TimeZone = UTC
# If true then file will be recreated on each start.
# If false then new records will be appended to the existent file.
Log.File.Recreate = false
# If set to true then buffer will be flushed after each logging call.
# If set to false then flush is not called.
# Setting to true decreases program performance; setting to false increases
# a risk of records loss in case of program failure.
Log.File.AutoFlush = true
# Enables rotation in the logs when size limit is reached
Log.File.Rotate = true
# Size of file in bytes. When log file reaches this size it is renamed
# (follow standard rotation approach) and the new file is created
Log.File.Rotate.SizeLimit = 104857600
# Number of files to be involved in rotation. When the number is exceeded
# the older file is removed

```

```
Log.File.Rotate.FileLimit = 0
# Rotation directory. Directory to store rotated files.
Log.File.Rotate.Dir = /log/FIXEdge1/logs
```