FIX Edge

Quick Start Guide

www.b2bits.com

B2BITS <epam>

Document History

Version	Date	Description of change
1.0	2007-06-19	Initial version
2.0	2007-08-13	Added SNMP section Added rotation for logging
3.0	2008-03-03	Added extra sections to “Configure FIX Layer” topic Added extra sections to “Configure transport adaptors” topic Added configuring routing layers Added SMTP section Added FE Admin Console guide section Added notes to configure FIX Edge for Linux Removed unnecessary details of “Check transport adaptors” topic Removed some examples from business scripting Merged the enumeration of transport adaptors from separated topics
4.0	2008-10-31	Added description for the new actions and conditions Changed description of the AddtionalField property Added section for the Histories and Automatic routing strategies
5.0	2008-11-12	Section “Sending rules” complemented with rules for sending into multiple destinations. Added section “Source identification”. Added section “FIX50/FIXT11 protocol configuration”. Updated section “Histories”.
6.0	2009-02-20	Configuring business rules are moved in Business Rules Guide.doc. Configuring SMTP are complemented with the new parameters.
7.0	2009-05-15	Added extra parameters to “Configure FIX Layer” topic Changed “Minimum Hardware” topic Changed “Configuring SMTP” section Removed other package versions of adaptors from section “Check Transport Adaptors”
8.0	2010-11-16	Removed FECC installation description. Added FIXICC description, setup and short information. Changed installation package structure. Added Socket Priority topic. Added Backup Connection topic. Complemented Customizing the FIX protocol topic. Changed Additional Parameters topic. Changed FIX Sessions topic. Complemented Tuning up the System for Maximum Performance topic. Changed Starting and Stopping FIX Edge sections. Changed Monitoring FIX Edge section. Changed FAQ and Glossary sections.
9.0	2012-05-17	Changed Contact Us section. Changed formatting.
10.0	2012-05-18	Changed FIX Edge Admin Console Utility section.
11.0	2012-05-21	Changed FIX Edge Admin Console Utility section: - parameters for – reload_bl command added.

Table of Contents

Introduction

The B2BITS (www.btobits.com) FIX Edge is a server application that provides FIX connectivity for standalone client applications. Client applications (hereinafter Clients) communicate with FIX Edge via one of the supported transport protocols (e.g. Simple Sockets, TIBCO, CORBA). FIX Edge can also be customized to use other connectivity bridges like CMS, ISE, INET etc.
Clients can use FIX protocol versions 4.0 - 4.4 inclusive and 5.0 over FIXT1,1.

Requirements

Hardware

- Pentium 233 MHz (Recommended: Pentium 500MHz or greater)
- 128 MB RAM (Recommended: 256 MB RAM or greater)
- 150 MB hard drive space (Recommended: 300 MB or greater)

Software

- Windows 2000 SP2 or higher
- Windows XP (Recommended for single mode)
- Windows Server 2003
- Windows Server 2003 Enterprise Edition (Recommended for highly available solution)
- Linux RedHat AS 4 or higher

Installation

Normally, FIX Edge is supplied as Windows installation package (*FixEdge-xxxx-setup.exe*, where xxxx is a FIX Edge version). Package consists from standard FIX Edge installation, FIXICC Agent wrapper and FIXICC Webstart Applciation. Use full installation or custom installation to choose a necessary configuration.

Note: advisable, the system user should have administrator's privileges to perform this installation. Otherwise, user can't install SNMP services if it will be needed for FIX Edge working.

Linux platforms: FIX Edge package is located in *FixEdge-xxxx.tar.gz* archive*.* Run the following command to extract it:

```
tar -zxvf FixEdge-xxxx.tar.gz
```

Multiple Instances of FIX Edge

It is possible to install more than one instance of FIX Edge on a single machine. All instances work independently (i.e. have separate list of sessions, independent configuration and log files), however they can be managed using common FIXICC control application.

Choosing Ports

It is important to make sure that ports are not used by other applications. In addition to that, the ports must be open on the firewall if the FIX connections must be made with clients outside the firewall. Port numbers are requested during the installation process. Contact your system administrator for proper port numbers.

Entering License

The license file is supplied separately from the installation package. The license information can be entered during the installation process when requested. If the step is skipped leaving the license field empty, it will be possible to enter the license key later either using control centre or by placing the license file to the directory where FIX Edge is installed. Same actions will can be applied to locate fixaj2-license.bin during an installation. This license file needed for working of the FIXICC control application

Check installation

Normally, this step can be skipped.
FIX Edge creates the following directory structure in the chosen system location:

```
FixEdge
|– bin
|   |– <FIX Edge binaries and batches>
|
|– doc
|    |– <FIX Edge documentation>
|
|– FixEdge1
|    |– conf
|    |     |–<FIX Edge 1st instance configuration files>
|    |
|    |– log
|         |–backup
|         |–<FIX Edge 1st instance log files>
|
|– …
|
|– FixEdgeN
|    |– conf
|    |     |–<FIX Edge Nth instance configuration files>
|    |
|    |– log
|         |–backup
|         |–<FIX Edge Nth instance log files>
|
|– engine.license
```

Check Transport Adaptors

The transport adaptor is usually a DLL stored in the FIX Edge binary directory (‘*FixEdge/bin*’). In some cases, however, additional files are required for the adaptor and/or the library can be placed in other directory. Transport adapter and its own configuration files can be extracted from the corresponding additional installation package in archive.

The following types of transport adaptors are available:

- Simple Mail Transport Protocol Adaptor
  SMTPAdaptorDll.dll library is placed in *…\FixEdge\bin* directory.

Linux platform: Unlike Windows versions Linux adaptor shared libraries have .so extensions.
Run the following command to extract packages from *.zip* archive:

```
unzip -j Adaptor_xx.zip
```

then copy the extracted .so adaptor libraries into *FixEdge/bin* directory.

Configuration

If no steps are skipped during the installation and no errors occur, FIX Edge will be fully installed and no additional configuration will be required.

Change license

Changing license is required, when a correspondent step is skipped during the installation process or when the license is expired. License information is stored in ‘*engine.license*’ file, which is placed in the FIX Edge directory. It is enough to replace the file with a newer version.
Note: FIX Edge will not start with an incorrect or expired license.

Configure service version

Normally, FIX Edge is installed as a service during installation. However this process can be blocked by the operation system or system security software in some cases. If FIX Edge does not appear among services (refer to ‘Control Panel/Administrative Tools/Services’), run ‘*FixEdge/bin/FixEdgeN.install.cmd’*, where N is the number of instance to be installed as a service (normally, it is 1 when only 1 instance is installed). If this installation goes without errors, FIX Edge service will appear in Services after refresh.
Optionally, all commands of services control are available from ‘Start Menu’.

Configure console version

No additional configuration is required for a console version.

Configure FIX Integrated Control Centre

Normally, no additional work is required to configure the FIX Integrated Control Centre application. It is a Java stand-alone application with provides monitoring and administration capabilities out-of-the-box for FIX Edge and any application embedding FIX Antenna C++, FIX Antenna Java, FIX Antenna .Net.
FIXICC can manage multiple FIX Edges on a single machine. It is automatically configured to manage all installed instances during the installation. However, it is possible to add a new FIX Edge instance to the FIXICC manually. To do this, make some changes to the FIXICC configuration file *…/FIXEdge/FIXICC-Webstart/webapps/webstart-2.2.6/fixicc-2.2.6/servers.xml*. Below you can find an example of FIXICC configuration file for two FIX Edge instances:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<connectionParamsList version="3">
        <connectionParams>
                <name>FixEdge1</name>
                <serverType>FIXEdge</serverType>
                <host>127.0.0.1</host>
                <port>8903</port>
                <senderCompID>AdminClient</senderCompID>
                <senderSubID></senderSubID>
                <senderLocationID></senderLocationID>
                <targetCompID>admin</targetCompID>
                <targetSubID></targetSubID>
                <targetLocationID></targetLocationID>
                <FIXVersion>FIX 4.4</FIXVersion>
                <HBI>120</HBI>
                <username>user</username>
                <password>123</password>
                <rememberPassword>true</rememberPassword>
                <EncryptMethod>NONE</EncryptMethod>
        </connectionParams>
        <connectionParams>
                <name>FixEdge2</name>
                <serverType>FIXEdge</serverType>
                <host>127.0.0.1</host>
                <port>8906</port>
                <senderCompID>AdminClient</senderCompID>
                <senderSubID></senderSubID>
                <senderLocationID></senderLocationID>
                <targetCompID>admin</targetCompID>
                <targetSubID></targetSubID>
                <targetLocationID></targetLocationID>
                <FIXVersion>FIX 4.4</FIXVersion>
                <HBI>120</HBI>
                <username>user</username>
                <password>123</password>
                <rememberPassword>true</rememberPassword>
                <EncryptMethod>NONE</EncryptMethod>
        </connectionParams>
  </connectionParamsList>
```

"FIXICC User Guide.pdf" is a document about the more detailed information for setup and working FIXICC

Configure FIX Layer

Normally, no additional effort is required to configure FIX Layer. FIX Edge is ready to create and/or maintain FIX sessions/connections right after being installed. However, some additional customization can be made to make FIX Edge friendlier.

FIX sessions

FIX session can play the role of an initiator or an acceptor. The only difference in these roles is that the session initiator sends the first logon and it is responsible for reconnection process whereas the session acceptor waits for an incoming logon message and replies with a confirming logon message. Also, in case of connection failure the acceptor session waits for the counterparty to initiate reconnection. The rest of the behaviour is identical for both initiator and acceptor.

FIX Edge accepts all incoming sessions by default. This means that there is no need to setup any specific pair of SenderCompID and TargetCompID additionally. Although this is a very convenient way, which does not require any manual intrusion, it is quite insecure. The configuration property in the engine.properties – UnregisteredAcceptor.CreateSession – can be set to 'true' or 'false' to turn this mode on or off. If this property is set to 'false', incoming logon messages from unknown sessions are ignored.

You can force FIX Edge to create FIX session on start using the following properties:
Default session settings to be configured:

- FixLayer.FixEngine.Sessions.BackupPath
- FixLayer.FixEngine.Sessions.DefaultStartTime
- FixLayer.FixEngine.Sessions.DefaultTerminateTime

Setting the enumeration of session to be configured:

- FixLayer.FixEngine.Sessions = XXX, YYY, ZZZ, …

For both session initiator and session acceptor:

- FixLayer.FixEngine.Session.XXX.SenderCompID
- FixLayer.FixEngine.Session.XXX.TargetCompID
- FixLayer.FixEngine.Session.XXX.SenderSubID
- FixLayer.FixEngine.Session.XXX.TargetSubID
- FixLayer.FixEngine.Session.XXX.SenderLocationID
- FixLayer.FixEngine.Session.XXX.TargetLocationID
- FixLayer.FixEngine.Session.XXX.Description
- FixLayer.FixEngine.Session.XXX.Role
- FixLayer.FixEngine.Session.XXX.StorageType
- FixLayer.FixEngine.Session.XXX.Version
- FixLayer.FixEngine.Session.XXX.Username
- FixLayer.FixEngine.Session.XXX.Password
- FixLayer.FixEngine.Session.XXX.InSeqNum
- FixLayer.FixEngine.Session.XXX.OutSeqNum
- FixLayer.FixEngine.Session.XXX.IntradayLogoutTolerance
- FixLayer.FixEngine.Session.XXX.ForceSeqNumReset
- FixLayer.FixEngine.Session.XXX.StartTime
- FixLayer.FixEngine.Session.XXX.TerminateTime
- FixLayer.FixEngine.Session.XXX.EncryptMethod
- FixLayer.FixEngine.Session.XXX.SourceIPAddress
- FixLayer.FixEngine.Session.XXX.RejectMessageWhileNoConnection
- FixLayer.FixEngine.Session.XXX.TcpBufferDisabled
- FixLayer.FixEngine.Session.XXX.MaxMessagesAmountInBunch
- FixLayer.FixEngine.Session.XXX.ActiveConnection
- FixLayer.FixEngine.Session.XXX.KeepState
- FixLayer.FixEngine.Session.XXX.EnableAutoSwitchToBackupConnection
- FixLayer.FixEngine.Session.XXX.EnableCyclicSwitchBackupConnection
- FixLayer.FixEngine.Session.XXX.ResetSeqNumAtScheduledStartTime
- FixLayer.FixEngine.Session.XXX.SocketPriority

For session initiator only:

- FixLayer.FixEngine.Session.XXX.Host
- FixLayer.FixEngine.Session.XXX.Port
- FixLayer.FixEngine.Session.XXX.HBI
- FixLayer.FixEngine.Session.XXX.RecreateOnLogout
- FixLayer.FixEngine.Session.XXX.ForcedReconnect
- FixLayer.FixEngine.Session.XXX.IgnoreSeqnumTooLowAtLogon
- FixLayer.FixEngine.Session.XXX.CustomLogonFileName

To register custom or FIXT11/FIX50 sessions:

- FixLayer.FixEngine.Session.XXX.Protocol
- FixLayer.FixEngine.Session.XXX.DefaultApplicationProtocol
- FixLayer.FixEngine.Session.XXX.PredefinedMessages = A, B

- FixLayer.FixEngine.Session.XXX.PredefinedMessage.A.Type
- FixLayer.FixEngine.Session.XXX.PredefinedMessage.A.Direction
- FixLayer.FixEngine.Session.XXX.PredefinedMessage.A.AppProtocol

Please refer to FIX50/FIXT11 protocol configuration for detailed information.

To initialize the session backup connection use the prefix Session.XXX.Backup.<Session Parameter>.
Sample:

```
FixLayer.FixEngine.Session.XXX.Backup.Port = 9107
FixLayer.FixEngine.Session.XXX.Backup.Host = BackHost
FixLayer.FixEngine.Session.XXX.Backup.HBI = 60
```

Note: the specified session name will be associated with source identifier of session (see Source identification).

Session time schedule

FIX Edge supports the time schedule for both acceptor and initiator sessions. StartTime and TerminateTime session properties can be used to define the local time to start and terminate the specified FIX session (HH:MM[:SS]).
DefaultStartTime and DefaultTerminateTime properties are optional. They specify the default session working time. If the start-up time is greater than the specified value, the session will be started immediately and the value will take effect only starting from midnight.

Intraday Logout Tolerance Mode

FIX Edge supports the mode that violates the FIX protocol, de-facto. However, it has become standard in practical use. If this mode is turned on (i.e. IntradayLogoutTolerance is set to ‘true’), the session will continue sequence numbers even in case it is re-established after correct termination. The default value of this parameter is ‘false’. This means that the session does not exist after correct termination and hence a newly created session with the same SenderCompID and TargetCompID starts with 1 as it is specified by the FIX protocol. It is crucial that both sides – the initiator and the acceptor – work in the same mode. Otherwise, the “sequence number is too low” fatal error appears when the session is re-established. This results in immediate session termination.
In the Intraday Logout Tolerance mode, the sequence number is never reset during the day. During the end-of-day procedure, sessions’ logs are archived so the next day session is created from scratch and hence sequence number starts from 1.

Transient and Persistent Sessions

FIX Edge supports two types of sessions: transient and persistent. Transient session stores all related information in memory. This means that this type of session has greater throughput and lower latency. However, such session cannot be restored after failure, and hence message loss is possible. Persistent session stores all related information on disk. This decreases performance but allows full restoration of session state after failure. It is possible to set session type on creation. The default type is persistent. ‘StorageType’ property specifies the mode to be used for automatically created unregistered acceptor-sessions.

Force Sequence Number Reset Mode

The “sequence number too low” error is specified by the FIX protocol as fatal situation that may occur as the result of the Intraday Logout Tolerance mode, when the sequence number is never reset during the day. The recommended behavior is immediate session termination. In real life, this situation can come as a result of sequence number de-synchronization (e.g. after non-graceful termination log files are cleared on one side and kept on the other side). When such situation is possible and intrusion is impossible, the “Force Sequence Number Reset Mode” can be used. In this mode, a reset sequence number logon is always sent when a session is created. This forces the remote side to reset sequence number if a higher value is expected. Use ‘ForceSeqNumReset’ property to turn this mode.

The following values can be applied for parameter:

- value "0" or "false" - Mode off
- value "1" or "true" – Mode on
- value "2" - Reset sequence number at every Logon
- another value - use default value from engine.properties file

Additionally, the property ‘ResetSeqNumAtScheduletStartTime’ can be used for sequence number resetting when session starts by scheduler. It is useful on case when session starts every day by some timetable and its sequence numbers restores from the session log.

RecreateOnLogout

This is an optional parameter for the session initiator. If it is set to “false” (by default) the session is removed from the list of sessions after successful disconnection. If it is set to “true” the session will be recreated after disconnection. Recreation will take place only if disconnection is initialized by the counterparty.

Encryption

FIX Edge can operate message encryption in the session by means of several cryptographic technologies: PGP, PEM, PKCS, DES or DES-MD5. Problems in use of encryption may occur in case of separated configuration of sessions that will be connected subsequently. In this case a session will be started, but errors will be logged during the action of message sending.

Both session participants should use the same encryption method. The server configuration does not support session participants’ work with different encryption technologies. The encryption methods should not differ for both sessions. The server configuration doesn’t provide yet the incompatible configuration of session participants.

Socket priority

This parameter can be applied for several modes of socket read and write operations.

Valid values:

- EVEN (default) - share worker thread among all session in the Engine
- AGGRESSIVE_SEND - use dedicated thread to send outgoing messages
- AGGRESSIVE_RECEIVE - use dedicated thread to receive incoming messages
- AGGRESSIVE_SEND_AND_RECEIVE - enables the both aggressive sending and aggressive receiving options

If parameter not specified for session then ‘SocketOpPriority’ engine parameters will be used instead.

Message Reject

When session connection is lost, all outgoing messages are automatically removed from the session's outgoing queue and sent to the target during SequenceGap resolving (when target sends ResendRequest message). Since session reconnection may take a lot of time, there is no sense to send messages when session is restored - they have to be removed as if they were not sent at all. Message rejecting is the mechanism of removing messages from the session's outgoing queue.
RejectMessageWhileNoConnection session property enables rejection of application messages for initiator and acceptor, when the session is unable to send them. By default message rejecting is disabled.
You can find more information about message rejecting configuration in *MessageRejecting_FunctionSpecification.html*.

Backup connection

FIX Edge provides an automatic switching between the primary and backup connections when one of these connections has lost or can’t be established. Following properties are uses for controlling of backup auto switching:
Session property ‘ActiveConnection’ specifies which connection is used, Valid values 'primary', 'backup', 'restore'. The 'restore' means the previous connection (primary or backup) is used to establish connection.
Session property ‘EnableAutoSwitchToBackupConnection’ enables auto switching between connections. Valid values ‘true’ or ‘false’.
Session property ‘EnableCyclicSwitchBackupConnection’ specifies auto switching from primary to backup and back while connection will be established. Valid values ‘true’ or ‘false’.
Session property ‘KeepState’ notifies what the backup connection executes without clearing sequence numbers and with old message files. Valid values ‘true’ or ‘false’.
Below is the example with valid parameters that can be accepted for backup connection setting:

```
FixLayer.FixEngine.Session.xxx.Backup.Host = localhost
FixLayer.FixEngine.Session.xxx.Backup.Port = 4434
FixLayer.FixEngine.Session.xxx.Backup.HBI = 5
FixLayer.FixEngine.Session.xxx.Backup.SenderSubID = SenderSubID
FixLayer.FixEngine.Session.xxx.Backup.TargetSubID = TargetSubID
FixLayer.FixEngine.Session.xxx.Backup.SenderLocationID = SenderLocationID
FixLayer.FixEngine.Session.xxx.Backup.TargetLocationID = TargetLocationID
FixLayer.FixEngine.Session.xxx.Backup.IntradayLogoutTolerance = true
FixLayer.FixEngine.Session.xxx.Backup.ForceSeqNumReset = true
FixLayer.FixEngine.Session.xxx.Backup.ForceReconnect = true
FixLayer.FixEngine.Session.xxx.Backup.RejectMessageWhileNoConnection = true
FixLayer.FixEngine.Session.xxx.Backup.IgnoreSeqNumTooLowAtLogon = true
FixLayer.FixEngine.Session.xxx.Backup.EnableAutoSwitchToBackupConnection = primary
FixLayer.FixEngine.Session.xxx.Backup.EnableCyclicSwitchBackupConnection = true
```

Customizing the FIX protocol

It is possible to extend the standard FIX protocol in FIX Edge. In particular, you can add a custom additional required field to message, create a privately defined message type etc. ‘Validation.AdditionalFieldsFileName’ property in the engine.properties is used to specify a file with the FIX protocol extension definition. Example of the FIX protocol extension:

```
<?xml version="1.0" encoding="UTF-8"?>
<fixdics>
<update>
<fixdic fixversion="4.4" title="FIX 4.4" date="2007/03/22">
  <fielddic>
      <fielddef tag="4999" name="4999" type="int">
      </fielddef>
      <fielddef tag="4998" name="4998" type="String">
      </fielddef>
      <fielddef tag="4997" name="4997" type="String">
      </fielddef>
      <fielddef tag="4996" name="4996" type="int">
      </fielddef>
      <fielddef tag="4995" name="4995" type="String">
      </fielddef>
      <fielddef tag="4994" name="4994" type="String">
      </fielddef>
  </fielddic>

  <msgdic>
    <msgdef msgtype="6" >
      <field tag="4999" />
      <group nofield="4999" startfield="4997">
        <field tag="4997" req="Y"/>
        <field tag="4998" req="N"/>
        <field tag="4996" req="N"/>
        <field tag="4995" req="N"/>
        <field tag="4994" req="N"/>
      </group>
    </msgdef>
  </msgdic>
</fixdic>

</update>
</fixdics>
```

Such abilities of protocol customization can be specified especially qualified for one or several sessions by following way:

- Describe in the FIXEdge.properties enumeration of protocol names:

```
        FIXEdge.CustomVersions = FIX41Custom, FIX42Custom,…
```

,there FIX41Custom, FIX42Custom is the user defined names of custom protocols.

- Specify base version of custom protocol and the path to additional fields XML file, what provides protocol customization:

```
        FIXEdge.CustomVersion.FIX41Custom.BaseProtocol = FIX41
        FIXEdge.CustomVersion.FIX41Custom.AdditionalFieldsFileName =
FixEdge1/conf/additionalFieldsFIX41.xml

        FIXEdge.CustomVersion.FIX42Custom.BaseProtocol = FIX42
        FIXEdge.CustomVersion.FIX42Custom.AdditionalFieldsFileName =
FixEdge1/conf/additionalFieldsFIX42.xml
```

- Specify FIXVersion of sessions where protocol customization will be applied:

```
        FixLayer.FixEngine.Session.Custom1.Version = FIX41Custom
        FixLayer.FixEngine.Session.Custom2.Version = FIX42Custom
```

Additional parameters

Below you can see a set of other FIX Layer properties with default values. All these parameters are stored in ‘*engine.properties*’ file and accessible from the FIXICC for each FIX Edge instance.

Parameter Name	Default Value	Description
EngineRoot	.	Do NOT add a slash at the end of the directory path
ListenPort	9106	Engine listen port. Must be > 0
ListenAddress	#commented	Engine local IP address to bind to. It can be used on a multi-homed host for a FIX Engine that will only accept or connect requests to one of its addresses. If this parameter is commented or empty, the engine will accept connections to any/all local addresses
LogFileName	engine.log	Engine log file name If this parameter is commented or empty, the *'engine.log'* will be used.
ConnectAddress		Engine local IP address to send from. It can be used on a multi-homed host for a FIX Engine that will only send IP datagrams from one of its addresses. If this parameter is commented or empty, the engine will send IP datagrams from any/all local addresses.
NumberOfWorkers	10	Number of threads that FIX sessions serve. This is independent of the number of sessions. Changing this value affects the performance of FIX Engine. The recommended value is 10. The value must be a positive integer.
LogDirectory	logs	This property is the path to the directory where the logs for all incoming (if LogIncomingMessages is set to "true") and outgoing FIX messages are stored. It is possible to specify a relative path from the EngineRoot directory. For example, if LogDirectory is set to “logs", the real path will be $(EngineRoot)/logs.

Parameter Name	Default Value	Description
		The specified directory must exist.
BackupDirectory	logs/backup	Relative path to the backup folder. This folder will be used for message storage files of the backup connections.
LogIncomingMessages	true	This property provides an option to log incoming FIX messages from some counterparty FIX Engine. The messages are stored in the directory specified by the LogDirectory parameter in a file with "in" extension.
LogonTimeFrame	3	This parameter sets the time period after which a session is terminated with “non-gracefully” flag (if no response is received to the first "Logon" message type A). The corresponding Logout message is sent to the counterparty. This value is in seconds. The recommended value is 30 seconds for broadband connections or private networks. Trading connections via the Internet will require calibration. If it is set to "0", the time period is unlimited. The value must be a non-negative integer.
LogoutTimeFrame	3	This parameter sets the time period after which the session is automatically terminated (if no response is received to a "Logout message" type 5). This value is in seconds. The recommended value is 10 seconds for broadband connections or private networks. Trading connections via the Internet require calibration. The value must be non-negative integer.
MessageMustBeValidated	true	This parameter controls the validation of application level messages. The possible values are "true" and "false". If it is set to "true", all application level messages are validated. If it is set to "false", the responsibility for message validity rests upon the counterparty. Please note that session level messages are validated in all cases. The Recommended setting is "true". Note: parameter is valid for FIX Edge assembled with old dynamic FIX parser.
ReasonableTransmissionTime	20	This parameter specifies the delta (increment) of

Parameter Name	Default Value	Description
		the Heartbeat interval between a TestRequest message sent by FIX Engine and received Response Heartbeat. The session attains a "telecommunication failed state" if no Response Heartbeat message is received after the normal Heartbeat interval plus delta. For example if no message (application or session level) is received during the Heartbeat interval, the Engine sends a TestRequest message. If the required Response Heartbeat message is not received during Heartbeat interval plus Delta, the session moves to "Telecommunication link failed" state. This parameter is specified in (Heartbeat Interval/100). The recommended value is twenty percent.
ThirdPartyRoutingIsEnabled	true	FIX Engine has inbuilt FIX message routing capability and fully supports the "Deliver To On Behalf Of" mechanism as specified by the FIX protocol. If this parameter is set to "True", the Engine redirects FIX messages automatically to other FIX sessions it maintains. If this parameter is set to "False", the Engine sends all received messages just to the client application.
UnregisteredAcceptor.CreateSession	true	This parameter provides an option whereby FIX Engine will accept a FIX session for which it has no registered application (an acceptor). When an application is registered behaviour is as standard. If set to false then Logon messages are ignored.
DelayedProcessing.DeliveryTriesInterval	500	This parameter specifies the time interval between attempts to deliver an application level message to a registered client application in case the application does not send a confirm receipt in response to the message. The value is specified in milliseconds. The value must be an integer greater than 0. This parameter is required only if the DelayedProcessing.MaxDeliveryTries parameter is specified.
DelayedProcessing.MaxDeliveryTries	2	This parameter specifies the number of attempts

Parameter Name	Default Value	Description
		that will be made to deliver an application level message to the registered client application. If this value is exceeded the session will be closed with the logout reason "Application not available". The recommended value is 10. The value must be integer and not negative. This parameter is optional.
Reconnect.MaxTries	3	This parameter specifies the number of attempts to restore the session. The session is considered as restored if the telecommunication link is restored and the exchange of Logon messages is successful. If it is set to "-1", the number of attempts is unlimited. This value must be an integer.
Reconnect.Interval	300	This parameter specifies the time interval between reconnection attempts in order to restore a communications link. This value is specified in milliseconds (seconds*10-3). The recommended value is 1000 for dedicated connections and private networks. Internet connections require calibration. The value must be an integer greater than 0.
LicenseFile	engine.license	The license file path.
OutgoingMessagesStorageSize	1000	This parameter defines the upper limit for the number of outgoing messages that are resent in case of a Resend Request. If the parameter is set to "-1", the number of messages is unlimited. The recommended value is 1000 if no information on mean activity is available. The value must be an integer not less than -1.
CheckVersionOfOutgoingMessages	true	This parameter is an option whereby the FIX protocol version used for the outgoing message is validated against that of the established session. If it is set to "true", the application must use the same version of the protocol as the established session, otherwise an error occurs. If it is set to “false”, the application level message will be sent to the counterparty. The recommended value is "true".
ExtraSafeMode	true	If this parameter is set to “true”, file streams are

Parameter Name	Default Value	Description
		flushed after each I/O operation.
ResetSeqNumAfter24hours	true	An option to send a Logon message with the ResetSeqNumFlag set to “true”. This means that after each 24 hour period of session activity a new set of sequence numbers is established (starting with 1).This parameter is optional, the default value is “false”. NOTE: This option does not affect sessions which use version 4.0 of the FIX protocol.
TimestampsInLogs	true	An option to write timestamps in the log files. This parameter is optional, but the default value is “true”.
Validation. AdditionalFieldsFileName	#commented	This parameter contains the name of an XML file with extensions of the FIX protocols.
Validation.CheckRequiredGroupFields	true	This parameter controls the validation of required fields in a repeating group. The possible values are "true" and "false". If it is set to "true", repeating groups will be checked for presence of required fields. If it is set to "false", the responsibility for repeating group validity rests with the counterparty. The recommended setting is "true". This parameter is optional. Note: parameter is valid for FIX Edge assembled with old dynamic FIX parser.
EncryptionConfigFile	encryption.properties	The path to an encryption configuration file.
ForceSeqNumReset	false	This parameter allow to automatically resolve sequence gap problem. When mode is ON session uses 141(ResetSeqNumFlag) tag in sending/confirming Logon message to reset SeqNum at the initiator or acceptor. Valid values: "0" or "false" - Disable ForceSeqNumReset mode "1" or "true" - Enable SeqNum reset at first time of session initiation "2" - Enable SeqNum reset for every session initiation
ForceSeqNumResetOnLogon	false	This parameter allow to automatically reset SeqNum after every logon.
UnregisteredSessionStorageType	#commented	Default storage type of the unregistered

Parameter Name	Default Value	Description
		sessions. By default a persistent storage type is used. Use "transient" value to use transient storage for the sessions.
MessageTimeToLive	500	This parameter sets the time period after which message rejecting is started while a session does not exist. The parameter isn't required. The value is specified in milliseconds (seconds*10-3), must be an integer and > 0.
IntradayLogoutTolerance	false	An option not to reset sequence numbers after Logout. Logout sender should initiate session recovery by sending Logon message with SeqNum = <last outgoing SeqNum> + 1; expecting reply Logon with SeqNum = <last incoming SeqNum> + 1. If a gap is detected, standard message recovery or gap filling process takes place.
ResendMessagesBlockSize	1000	Resend Request resends messages by blocks. This parameter defines how many messages proceed in a block. By default the parameter is set to 0 - all messages will be resent in one block. The value must be an integer and not less than 0.
UnregisteredAcceptor. IgnoreSeqNumTooLowAtLogon	#commented	This parameter allows resolving seqNum too low problem at logon for unregistered acceptors. If it is set to “true”, the session will continue with received seqNum.
UnregisteredAcceptor. RejectMessageWhileNoConnection	#commented	If it is set to “true”, unregistered acceptors will reject messages in case they cannot be sent during interval.
AllowZeroNumInGroup	false	If it is set to “true”, a leading group tag with 0 value will be ignored in a raw message. Otherwise exception will be fired.
UnregisteredAcceptor.tcpBufferDisabled	#commented	If it is set to “true”, the TCP buffer (Nagle algorithm) will be disabled for the unregistered acceptors. Otherwise, TCP may join and enqueue small packages until timeout ends.
UnregisteredAcceptor. maxMessagesAmountInBunch	#commented	FA is able to join packages that wait for sending into the socket, this parameter controls the number of messages that can be joined. 0 means infinite. Value should be less than

Parameter Name	Default Value	Description
		1000000.
AllowEmptyFieldValue	false	When it is set to “true”, the raw message may contain tags without values - they will be ignored. Otherwise exception will be fired.
TotalOutgoingStorageMemoryLimit	0	Specifies total amount of the memory (in MB) that active session may use. 0 - means infinite. When limit is overbound - the "hardest" sessions will be closed nongracefully.
DuplicateResendRequestLimit	0	Specified how much same ResendRequests received before Application::onResendRequestLoop is called. This option is disabled if value is less than 2.
EnableIncrementalLogFileCreation	false	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" FIX Antenna performance is greatly increased.
Debug.LogSessionExtraParameters	true	Setting property "true" will enable logging session parameters on creation.
Monitoring.Enable	false	If it is set to “true”, FIX Antenna monitoring will be enabled.

Tuning up the System for Maximum Performance

The following properties can be changed to increase the FIX Edge performance.

‘LogIncomingMessages’ can be set to “false” since incoming messages are not required for normal session work. FIX session should care only about outgoing messages, which can be requested for resending.

‘NumberOfWorkers’ is directly linked to the number of threads used inside the FIX Edge. Increasing this number on multi CPU machine may increase performance. However setting too big value may give the opposite effect since more time will be spent on switching between threads.

‘ExtraSafeMode’ can be turned off. This will lead to more effective HDD buffer use. However, it increases the probability of loosing messages in case of failure since message can be partially stored in HDD buffer and will never be stored on disk.

The session property ‘SocketPriority’ or ‘SocketOpPriority’ in the engine.properties can be turned on ‘AGGRESSIVE_SEND_AND_RECEIVE’ value to reduce latency by means of using the separate workers for read and write socket operations.

By default, all these properties are set to the recommended values.

FIX50/FIXT11 protocol configuration

FIXEdge can optionally be configured for sessions with support of the FIX50/FIXT11 protocols. Parameters that are enumerated below are not required for sessions with protocol versions earlier than FIX50, only for FIX50/FIXT11 explicitly.

Session.XXX.Protocol
Description: Specifies the underlying protocol of session XXX.
Values: “FIX_TCP”, “FIXT11_TCP”, “FIXT11_FAST_TCP”.
Not required for sessions based on FIX_TCP.

Session.XXX.DefaultApplicationProtocol
Description: Specifies the default application protocol of custom sessions.
Values: *“FIX40”, “FIX41“,“FIX42“,“FIX43“,“FIX44“,“FIX50“,“FIXT11”*.
Required only for sessions based on FIXT11_TCP.

Session.XXX.PredefinedMessages
Description: Enumerates the names of messages that will be used in *PredefinedMessage* group parameter.
Applicable only for sessions based on FIXT11_TCP.

Session.XXX.PredefinedMessage
Description: This is a group parameter that describes parameters of some FIX messages.
PredefinedMessage supports the following parameters:

- *Type* –FIX message type;
- *Direction* – describes the direction of message routing. Values – *“receive”, “send”*;
- *AppProtocol* – contains the FIX protocol version. Values are similar to

DefaultApplicationProtocol values.

Example of “CustomSession” configuration that supports the FIX50 protocol:

```
FixLayer.FixEngine.Session.CustomSession.Version = FIX50
FixLayer.FixEngine.Session.CustomSession.Protocol = FIXT11_TCP
FixLayer.FixEngine.Session.CustomSession.DefaultApplicationProtocol= FIX50

FixLayer.FixEngine.Session.CustomSession.PredefinedMessages= Din, Dout

FixLayer.FixEngine.Session.CustomSession.PredefinedMessage.Din.Type= D
FixLayer.FixEngine.Session.CustomSession.PredefinedMessage.Din.Direction = receive
FixLayer.FixEngine.Session.CustomSession.PredefinedMessage.Din.AppProtocol = FIX44

FixLayer.FixEngine.Session.CustomSession.PredefinedMessage.Dout.Type= D
FixLayer.FixEngine.Session.CustomSession.PredefinedMessage.Dout.Direction = send

FixLayer.FixEngine.Session.CustomSession.PredefinedMessage.Dout.AppProtocol = FIX43
```

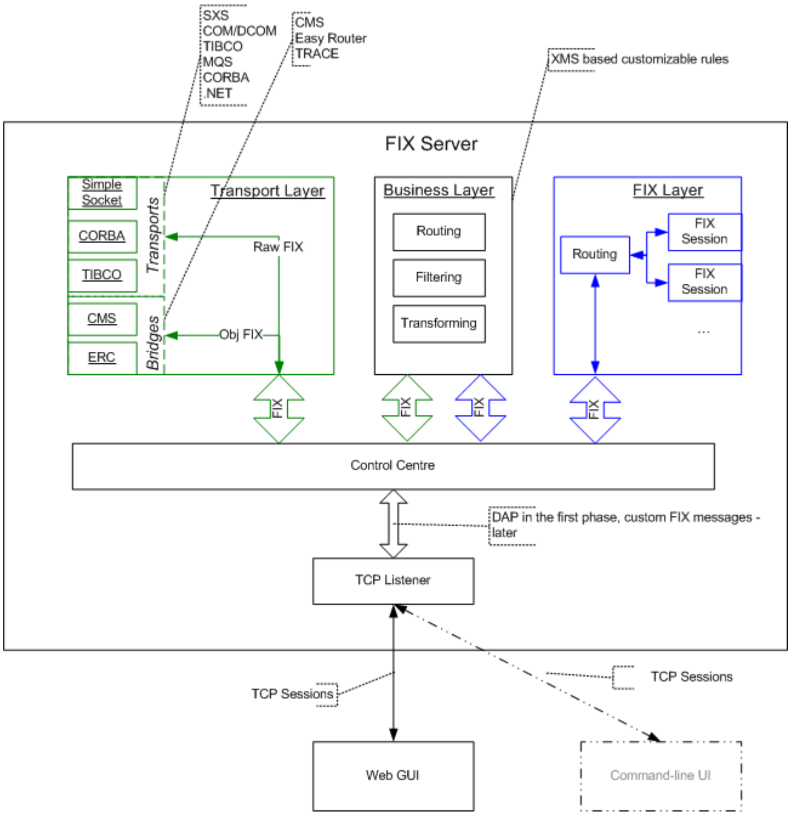
Configuring business rules

Business rules are stored in '*FixEdge/conf/BL_Config.xml'* file in XML format. The name of a configuration file can be changed in '*FixEdge.properties*' file using '*BusinessLayer.RoutingRules'* property. The XML formatting is described in '*FixEdge/conf/BusinessLayer.dtd'*. Rules can be changed from FIXICC or by editing the file directly. They can also be reloaded immediately from FIXICC.
You can find the detailed information about the syntax and configuration of business rules in FIX Edge - Business Rules Guide.

Configure Transport Adaptors

Overview

Transport Adaptors (TA) is a library that can be loaded by FIX Edge under Transport Layer. Technically it is a DLL that is loaded by server in the run time. TA handles sessions, which are identified by the ClientID in contrast to FIX Sessions, which are identified by the pair of SenderCompID and TargetCompID.



There are Transport Adaptors (TAs) and Bridge Adaptors (BAs). The difference between them is that TAs use raw interface i.e. FIX message is serialized to byte array to pass through the adaptors' border, whereas bridges have object interface. This also means that bridges require knowledge of FIX Message object, whereas transports adaptors can overlook the type and handle message as a sequence of bytes.

Integration

Libraries are placed in FIX Edge ‘*bin*’ directory and added to the properties file. To make the Transport Layer load TA the following lines are added:

```
# Comma separated list of identifiers of Transport Adapters should be loaded.
TransportLayer.TransportAdapters = TransportLayer.Adapor1_Name TransportLayer.Adaptor2_NAME

# Description of Transport Adaptor
TransportLayer. Adapor1_Name.Description = Test Adaptor

# Relative path and file name of DLL (use .so extension of adaptor libraries for Linux)
TransportLayer. Adapor1_Name.DllName = bin/TestAdaptorAddin_xx.dll

# Abbreviation specifying the type of communication with Transport Adapter
TransportLayer. Adapor1_Name.Type = DLL
```

All properties related to this adaptor should be started with ‘*TransportLayer. Adapor1_Name*’ prefix. All adaptor properties are passed to the adaptor as during initialization.

Configuration

Adaptors manage sessions and inform the Transport Layer about handled sessions sending their ClientIDs. The Transport Layer in its turn sends this information to the Business Layer, which makes routing between sessions possible. ClientIDs can be used in routing rules in the same way as FIX Session IDs. For example:

```
<Rule>
        <Source>
                <FixSession SenderCompID=".*" TargetCompID=".*"/>
        </Source>
        <Condition>
                <MatchField Field="35" Value="(D|F|G)"/>
        </Condition>
        <Action>
                <Send>
                        <Client Name="XASE"/>
                </Send>
        </Action>
</Rule>
```

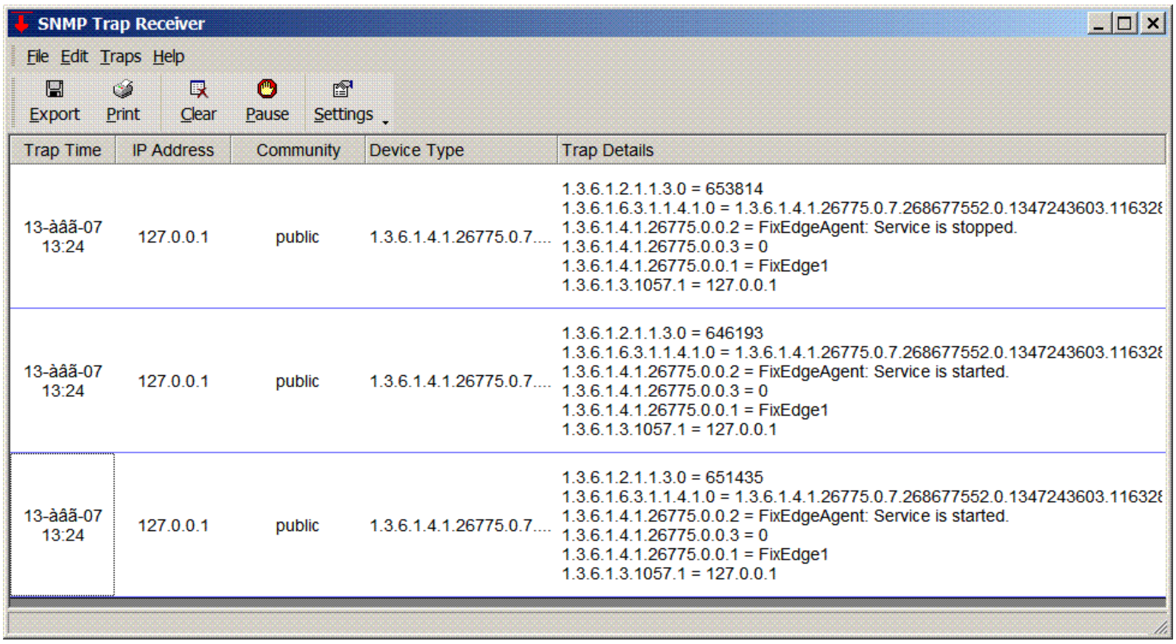
Every transport adaptor can be configured by using FECC or by editing the ‘*FixEdge.properties*’ file of the correspondent FIX Edge instance. Integration Guide with more detailed information about settings and configurations is included in the installation package of the corresponding transport adaptor.

- *…\package\doc\SXSAdaptor_IntegrationGuide.html*
- *…\ package\doc\SFSAdaptor_IntegrationGuide.html*
- *…\ package\doc\COMAdaptor_IntegrationGuide.html*

- *…\ package\doc\MQAdaptor_IntegrationGuide.html*
- *…\ package\doc\NPipeAdaptor_IntegrationGuide.html*
- *…\ package\doc\CMSAdaptor_IntegrationGuide.html*

Configuring SNMP

FIX Edge installation package is supplied with SNMP module. It is required to have SNMP service installed (standard Windows component) prior to installing FIX Edge, otherwise SNMP module will not be installed.
After the installation use SNMP monitor to control the FIX Edge status. The Private Enterprise Number assigned by IANA to B2BITS is 26775 so the B2BITS FIX Edge branch can be viewed at 1.3.6.1.4.1.26775. There is also an MIB that can be loaded to SNMP monitoring software. Most vendors integrate MIBs into their MIB bases automatically.
Below you can see an example of how FIX Edge looks like in Solar Winds toolset:



Trap Time	IP Address	Community	Device Type	Trap Details
13-àâã-07 13:24	127.0.0.1	public	1.3.6.1.4.1.26775.0.7....	1.3.6.1.2.1.1.3.0 = 653814 1.3.6.1.6.3.1.1.4.1.0 = 1.3.6.1.4.1.26775.0.7.268677552.0.1347243603.116328 1.3.6.1.4.1.26775.0.0.2 = FixEdgeAgent: Service is stopped. 1.3.6.1.4.1.26775.0.0.3 = 0 1.3.6.1.4.1.26775.0.0.1 = FixEdge1 1.3.6.1.3.1057.1 = 127.0.0.1
13-àâã-07 13:24	127.0.0.1	public	1.3.6.1.4.1.26775.0.7....	1.3.6.1.2.1.1.3.0 = 646193 1.3.6.1.6.3.1.1.4.1.0 = 1.3.6.1.4.1.26775.0.7.268677552.0.1347243603.116328 1.3.6.1.4.1.26775.0.0.2 = FixEdgeAgent: Service is started. 1.3.6.1.4.1.26775.0.0.3 = 0 1.3.6.1.4.1.26775.0.0.1 = FixEdge1 1.3.6.1.3.1057.1 = 127.0.0.1
13-àâã-07 13:24	127.0.0.1	public	1.3.6.1.4.1.26775.0.7....	1.3.6.1.2.1.1.3.0 = 651435 1.3.6.1.6.3.1.1.4.1.0 = 1.3.6.1.4.1.26775.0.7.268677552.0.1347243603.116328 1.3.6.1.4.1.26775.0.0.2 = FixEdgeAgent: Service is started. 1.3.6.1.4.1.26775.0.0.3 = 0 1.3.6.1.4.1.26775.0.0.1 = FixEdge1 1.3.6.1.3.1057.1 = 127.0.0.1

Note: Solar Winds toolset is available on the official site of SolarWinds.

If events are not caught by trap, check the following:

- "SNMP Service" and "SNMP Trap Service" are installed and started
- The "SNMP Service" is configured
  - Community name "public" is added to the list of traps
  - Localhost (or 127.0.0.1) IP address is added to traps
  - "Profile 1" is enabled
  - "Local system account" is enabled
- The "SNMP Trap Service" is configured
  - "Profile 1" is enabled
  - "Local system account" is enabled

Please note that the agent writes information about SNMP to *Systems32* directory. It is recommended to delete "*snmpFixEdgeAgent.log*" periodically. The "SNMP Service" must be stopped to delete file.

Note: FIX Edge for Linux does not support SNMP traps currently. The update in order was inserted into the scope of FIX Edge milestones where it should be solved for perspective.

Configuring SMTP Transport Adaptor

The SMTP service support is provided by the SMTP Transport Adaptor which has to meet the following requirements:

- A FIX message has to be converted into an e-mail message
- An e-mail message has to be sent to the client by the SMTP protocol
- The adaptor establishes a connection when it has a message to send
- The adaptor has no message queue, all messages are sent synchronously.

The SMTP Transport Adaptor can be configured by using FECC or by editing the '*FixEdge.properties'* file of the correspondent FIX Edge instance.

Use "TransportLayer.SmtpTA" prefix in the SMTPAdaptor properties file. The following properties are used to configure the SMTPAdaptor:

Description - description of the SMTPAdaptor
DllName - path and filename of the SMTPAdaptor dll. The property is required.
Type - type of the SMTPAdaptor, "DLL" must be included. The property is required.
SMTPSessions – enumeration of the registered SMTP servers. The property is required for working sessions set. Session names must be separated by comma.
SMTPSessions.DefaultServerName - default SMTP server name or IP address. The property is not required.
SMTPSessions.DefaultServerPort - default SMTP server port. The property is not required.
SMTPSessions.DefaultFrom - default value of the 'From' field. The property is not required.
SMTPSessions.DefaultCC - default value of the 'CC' field. The property is not required.
SMTPSessions.DefaultBCC - default value of the 'BCC' field. The property is not required.
SMTPSessions.DefaultSmartEmailProcessing - when it is set to "true", the corresponding tags will be used for e-mail Subject and Body at FIX EMail message conversation. The property is not required.
SMTPSessions.DefaultSendingTimeout – message sending timeout, common for all SMTP sessions. The property is not required.
SMTPSessions.DefaultConnectionTimeout – SMTP server connection timeout, common for all SMTP sessions. The property is not required.

The following properties are defined for the SMTPServer:
SMTPSession.XXX.Name - unique name of the connection. The property is required.
SMTPSession.XXX.ServerName - SMTP server name or IP address. The property is required when DefaultServerName is not defined.
SMTPSession.XXX.ServerPort - SMTP server port. The property is required when DefaultServerPort is not defined.
SMTPSession.XXX.SecureConnection - type of secure connection: TLS, SSL. The property is required if secure connection is needed, otherwise – not required.
SMTPSession.XXX.Login - user login. The property is required if authorization is needed, otherwise – not required.
SMTPSession.XXX.Password - user password. The property is required if authorization is needed, otherwise – not required.

SMTPSession.XXX.From - value of the 'From' field. The property is not required if DefaultFrom is defined, otherwise – required.
SMTPSession.XXX.To - value of the 'To' field. The property is required.
SMTPSession.XXX.CC - value of the 'CC' field. The property is not required.
SMTPSession.XXX.BCC - value of the 'BCC' field. The property is not required.
SMTPSessions.XXX.SmartEmailProcessing - when it is set to "true", the corresponding tags will be used for e-mail Subject and Body at FIX EMail message conversation. The property is not required.
SMTPSessions.XXX.SendingTimeout – message sending timeout, defined for a specified session. The property is not required.
SMTPSessions.XXX.ConnectionTimeout - SMTP server connection timeout, defined for a specified session. The property is not required.

Property file example:

```
TransportLayer.SmtpTA.Description = Test SMTP adaptor
TransportLayer.SmtpTA.DllName = ./SMTPAdaptorDll.dll
TransportLayer.SmtpTA.Type = DLL
TransportLayer.SmtpTA.SMTPSessions = My
TransportLayer.SmtpTA.SMTPSessions.DefaultServerName = wildfly
TransportLayer.SmtpTA.SMTPSessions.DefaultServerPort = 25
TransportLayer.SmtpTA.SMTPSessions.DefaultFrom = dymmy@foo.ua
TransportLayer.SmtpTA.SMTPSession.My.Name = SmtpClient
TransportLayer.SmtpTA.SMTPSession.My.To = dymmy@foo2.ua
```

E-mail message configuration:
SMTPAdaptor creates an e-mail message and fills it in the following way:
Field 'From' - using the value of SMTPServer.XXX.From property
Field 'To' - using the value of SMTPServer.XXX.To property
Field 'CC' - using the value of SMTPServer.XXX.CC property
Field 'BCC' - using the value of SMTPServer.XXX.BCC property
Field 'Subject' - contains the text "Message 'X' from 'Y' to 'Z'.", where X - message type, Y - value of the SenderCompId tag, Z - value of the TargetCompId tag.
Message body contains a text representation of FIX message.

Configuring Logging in FIX Edge

FIX Edge provides multiple configuration parameters for flexibility in logging. Each FIX Edge instance stores log files separately in "*FixEdge/FixEdge1/log*" directory. Logging parameters can be configured either from FECC or directly in "*FixEdge.properties*" file. By default, no additional configuration is required for logging. However, to make log analyzing more convenient the administrator can turn on/off details, direct logging in different files, etc.

Log.File.RootDir	C:/Program Files/B2Bits/FixEdge	Target directory. Usually filled automatically during installation. Please specify absolute directory path for NT Service application.
Log.Device	File Console	Target devices. Supported; File, Console, EventLog (only for Windows), Syslog (only for Unix)

Log.DebugIsOn	true	Turns on/off logging on the debug level
Log.NoteIsOn	true	Turns on/off logging on the notice level
Log.WarnIsOn	true	Turns on/off logging on the warning level
Log.ErrorIsOn	true	Turns on/off logging on the error level
Log.FatalIsOn	true	Turns on/off logging on the fatal level
Log.Cycling	false	Turns cycling on/off
Log.Cycling.Ignore	3	Number of repeating records to be placed to log before cycling is started
Log.Cycling.BlockSize	10	Number of repeating records to hold hidden before writing the "cycle record" to the log.
Log.Cycling.Multiplier	10	Multiplier for the Block Size. If the number of messages reaches BlockSize and the same message still appears, the next BlockSize is calculated as the previous one multiplied by Multiplier.
Log.File.Name	FixEdge1/log/FixEdge.log	Log file name. If more than one category uses files with the same name, the same file will be used simultaneously.
Log.File.Recreate	false	If it is set to "true", the file will be recreated on each start. Otherwise old records will be appended with new ones in the existing file.
Log.File.AutoFlush	true	If it is set to "true" the buffer will be flushed after each logging call. # If it is set to "false" the 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.Rotate	false	Enables rotation in logs when size limit is reached.
Log.File.Rotate.SizeLimit	100000000	Size of file in bytes. When log file reaches this size it is renamed (following the standard rotation approach) and a new file is created.
Log.File.Rotate.FileLimit	5	The number of files to be involved in rotation. When the number is exceeded the oldest file is removed.
Log.File.Rotate.Dir	logs/rotate	Rotation directory. Directory to store rotated files.
Log.EventLog.EventSource	FixEdge_FixEdge1	It specifies the event source when FIX Edge starts as Windows Service.

Starting FIX Edge

Before starting FIX Edge make sure that it is not running. Otherwise the start procedure will fail.

Starting Service version

To start FIX Edge as a service, you can use FIXICC or system services. You can also start FIX Edge service from the command line using: "*net start FixEdge.FixEdge1*" instruction, where 1 is an instance number. Service initialization problems are logged to the "*FixEdgeInit.log*" file which is located in *%TEMP%* directory of system account (usually *%SYSTEMROOT%\Temp*).
Note: only the console version of FIX Edge can be used for Linux so far.

Starting Console version

To start the console version of FIX Edge, write the following instruction in the command line: "*FixServer.exe PATH_TO_FixEdge_PROPERTIES_FILE*". You can also use *"FixEdge1.run.cmd*" in the bin directory.
Notes: FIX Edge blocks the console after the start so you can use the 'start' keyword when starting FIX Edge from other console application.
Launching the version for Linux you should start from command line: *./FixEdge PATH_TO_FixEdge_PROPERTIES_FILE*

Checking if FIX Edge Is Started and Running

If FIX Edge is up and running the following must be true:

- If FIX Edge is started as a service it must be indicated in Services administration tool.
- FIXICC shows that a FIX Edge instance is started
- The following records appear in "*engine.log*":

```
   NOTE: 20060522-13:45:26.656 : The FIX Engine was started successfully.
   WARNING: 20060522-13:45:26.703 : The license expires in XXX days. Please contact us at fix@btobits.com for further
assistance.
```

- The following records appear in "*FixEdge.log*":

```
   [NOTE]  20060518-13:21:59.531 - FixEdge Version XXX started.
   [NOTE]  20060518-13:21:59.546 - Module 'BusinessLayer' version XXX was loaded.
   [NOTE]  20060518-13:21:59.875 - Module 'FixEngine' version XXX was loaded.
   [NOTE]  20060518-13:21:59.875 - Module 'FixLayer' version XXX was loaded.
```

Stopping FIX Edge

Before stopping FIX Edge make sure that it is running. Otherwise the stop procedure will fail.

Stopping Service Version

FIX Edge service can be stopped either using FIXICC or from Services. It can also be stopped from the command line using: “*net stop FixEdge.FixEdge1*” instruction, where 1 is an instance number.
Note: only the console version of FIX Edge can be used for Linux so far.

Stopping Console Version

The console version of FIX Edge can be stopped by pressing “Ctrl+C” keys combination.

Checking If FIX Edge Is Stopped

When FIX Edge is stopped (not running) the following must be true:

- If the service version is used, it must be shown as stopped in Services
- FIXICC shows the corresponding instance as stopped
- The following record appears in “*engine.log*”:

```
NOTE: 20060519-15:20:15.062 : The FIX Engine is closing.
```

Monitoring FIX Edge

FIX Edge can be monitored using log files or from FIXICC application. Log files for each instance are stored separately in “*FixEdge/FixEdge1/log*” directory.

Monitoring sessions

Each FIX session has its own logs, which consist of: incoming messages log file (.in), outgoing messages log file (.out), configuration file (.conf) and other auxiliary files (e.g. index file). Session state is shown in the configuration file. File name is constructed from session SenderCompID, TargetCompID and a unique number based on the current time. It is possible that more than one set of files exists for a certain SenderCompID and TargetCompID. In this case files with greater number in suffix are used.

Monitoring setup

By default FIX engine has one administration session to keep connection with FIXICC control application. There is a set of monitoring parameters that is accessible from engine.properties:

Parameter Name	Default Value	Description
Monitoring.Enable	true	If it is set to “true”, FIX Edge monitoring will be enabled.
Monitoring.AdminSessionDef.TargetSubID		TargetSubID (tag 57) - assigned value used to identify specific individual or unit intended to receive message.
Monitoring.AdminSessionDef.SenderLocationID		SenderLocationID (tag 142) - assigned value used to identify specific message originator's location (i.e. geographic location and/or desk, trader).
Monitoring.AdminSessionDef.TargetLocationID		TargetLocationID_ (tag 143) - assigned value used to identify specific message destination's location (i.e. geographic location and/or desk, trader).
Monitoring.AdminSessionDef.Username		The expected value of the Username (Tag 553) field in the incoming Logon message. If the real value is not equal to the expected one then the session is disconnected without sending a message and the error condition is generated in the log output.
Monitoring.AdminSessionDef.Password		The expected value of the Password (Tag 554) field in the incoming Logon message. If the real value is not equal to the

Parameter Name	Default Value	Description
		expected one then the session is disconnected without sending a message and the error condition is generated in the log output.
Monitoring.AdminSessionDef.SourceIPaddress		The expected value of the source IP address. If the real value is not equal to the expected one then the session is disconnected without sending a message and the error condition is generated in the log output.
Monitoring.AdminSessionDef.EncryptMethod	NONE	The expected value of the encryption method.
Monitoring.AdminSessionDef. IntradayLogoutToleranceMode	false	Intraday logout tolerance mode. An option to reset or not to reset sequence numbers after Logout.
Monitoring.AdminSessionDef. ForceSeqNumResetMode	false	Force SeqNum reset mode. An option to use 141 tag in Logon message to reset sequence number.
Monitoring.AdminSessionDef. IgnoreSeqNumTooLowAtLogon	false	When true, session ignore 'SeqNum too low' at incoming Logon message and continue with received SeqNum.
Monitoring.AdminSessionDef.DisableTCPBuffer	false	When true TCP buffer (Nagle algorithm) will be disabled for session.
Monitoring.AdminSessionDef. MaxMessagesAmountInBunch	0	Enqueued outgoing messages could merged and sent as a single buffer. This parameter controls how many messages could be merged into the bunch. The 0 means infinite amount.
Monitoring.AdminSessionDef.SocketOpPriority	EVEN	Priority of the socket SendReceive operations. Valid values: EVEN (default) - share worker thread among all session in the Engine AGGRESSIVE_SEND - use dedicated thread to send outgoing messages
Monitoring.AdminSessionNames	AdminClient	Enumeration of the administration sessions names. Actually, each session can use default value are described before and such parameters can be specified for session

Parameter Name	Default Value	Description
		obviously.

Some example with parameters to describe two administrative sessions:

```
Monitoring.AdminSessionNames = AdminClient, AdminClient2

# User monitoring tool (TargetCompId = AdminClient)
Monitoring.AdminSession.AdminClient.TargetCompId = AdminClient
Monitoring.AdminSession.AdminClient.Version = FIX42
Monitoring.AdminSession.AdminClient.TargetSubID =
Monitoring.AdminSession.AdminClient.SenderLocationID =
Monitoring.AdminSession.AdminClient.TargetLocationID =
Monitoring.AdminSession.AdminClient.Username = sds_fix_os
Monitoring.AdminSession.AdminClient.Password = Starts123
Monitoring.AdminSession.AdminClient.SourceIPaddress = 194.10.0.54
Monitoring.AdminSession.AdminClient.EncryptMethod = DES
Monitoring.AdminSession.AdminClient.IntradayLogoutToleranceMode = true
Monitoring.AdminSession.AdminClient.ForceSeqNumResetMode = true
Monitoring.AdminSession.AdminClient.IgnoreSeqNumTooLowAtLogon = false
Monitoring.AdminSession.AdminClient.DisableTCPBuffer = false
Monitoring.AdminSession.AdminClient.MaxMessagesAmountInBunch = 10
Monitoring.AdminSession.AdminClient.SocketOpPriority = AGGRESIVE_SEND

# User monitoring tool (TargetCompId = AdminClient2)
Monitoring.AdminSession.AdminClient2.TargetCompId = AdminClient2
Monitoring.AdminSession.AdminClient2.Version = FIX44
```

View logs

Logs can be viewed using ordinary text editor/viewer (e.g. notepad). There are several types of log files: general logs (contain records that reflect events; can be painlessly cleared); FIX session logs (contain FIX session configuration and messages; should not be cleared since they are required for session restoration, however can be removed to make session to be created from scratch i.e. prevent session restoration) and technical logs (e.g. indexes, persistence queues files etc; should never be removed manually). Ordinary logs contain records, which consist of: log level (debug, note, warn, error, fatal), timestamp, thread ID, category name (only for debug log level) and description. Each log category is written to the log file when registered. For example:

```
[NOTE]  20060607-13:47:31.328 - Log Category is registered to FileLogger
 File name: 'C:/Blue/FixEdge/ARCAGW/log/FixEdge.log'
 Recreate file on restart: off
 Flush on each record: on
 Category contains default settings
```

The separate log file is ‘*engine.log*’. This specific file is used exclusively for FIX engine information i.e. session creation, destruction, reconnection. It also contains information about license expiration date. If FIX Edge cannot be started correctly because of the license, it will contain the error record.

FIX Edge Admin Console Utility

The main goal of the Admin Console Utility (ACU) is providing the way to control FIX Edge using command line interface. The utility exists for both Windows and Linux platforms. The utility file is located in the binary directory of FIX Edge.

Main features

The ACU calls FIX Edge commands using XmlSocket Monitor.
The ACU provides the following functionality:

- *-create_acceptor* - creates FIX session as acceptor
  Parameters:
  - -host *Host_Name* - the FIX Edge host name. Default: localhost.
  - -port *Port_Number* - the FIX Edge XML Socket monitor port number. Default: 9211.
  - -sender *Sender_Name* - the sender ID. REQUIRED.
  - -target *Target_Name* - the target ID. REQUIRED.
  - -sender_location_id *Sender_ID* - the sender location ID. Empty by default.
  - -target_location_id *Target_ID* - the target location ID. Empty by default.
  - -sender_sub_id *Sender_Sub_ID* - the sender sub ID. Empty by default.
  - -target_sub_id *Target_Sub_ID* - the target sub ID. Empty by default.
  - -version *Version_ID* - version of the FIX protocol. REQUIRED.
  - -inseqnum *Seq_Number* - a new input sequence number. Default: 0.
  - -outseqnum *Seq_Number* - a new output sequence number. Default: 0.
  - -username *Name* - username for FIX Session authentication. Empty by default.
  - -password *Password* - password for FIX Session authentication. Empty by default.
  - -encrypt_method *Flag* - encrypt method. Default: false.
  - -start_time *Time* - local time to start the session (HH:MM). Empty by default.
  - -terminate_time *Time* - local time to terminate the session (HH:MM). Empty by default.
  - -source_ip_address *Address* - the expected value of the source IP address. Empty by default.
  - -intraday_logout_tolerance *Flag* - an option not to reset sequence numbers after logout. Default: false.
  - -force_seqnum_reset *Flag* - use ResetSeqNumFlag to reset seqNum. Default: false
  - -recreate_on_logout *Flag* - defines whether the session must be recreated on logout. Default: false.
  - -force_reconnect *Flag* - defines whether the session will reconnect. Default: false.
  - -ignore_seqnum_too_low_at_logon *Flag* - When it is set to “true”, the session continues with received seqNum. Default: false.
  - -reject_message_while_no_connection *Flag* - When it is set to “true”, the message that wasn't sent since the time interval will be rejected. Default: false.

- *-create_initiator* - creates FIX session as initiator
  Parameters:
  - -host *Host_Name* - the FIX Edge host name. Default: localhost.
  - -port *Port_Number* - the FIX Edge XML Socket monitor port number. Default: 9211.
  - -sender *Sender_Name* - the sender ID. REQUIRED.
  - -target *Target_Name* - the target ID. REQUIRED.
  - -sender_location_id *Sender_ID* - the sender location ID. Empty by default.
  - -target_location_id *Target_ID* - the target location ID. Empty by default.
  - -sender_sub_id *Sender_Sub_ID* - the sender sub ID. Empty by default.
  - -target_sub_id *Target_Sub_ID* - the target sub ID. Empty by default.
  - -version *Version_ID* - version of the FIX protocol. REQUIRED.

- -inseqnum *Seq_Number* - a new input sequence number. Default: 0.
- -outseqnum *Seq_Number* - a new output sequence number. Default: 0.
- -username *Name* - username for FIX Session authentication. Empty by default.
- -password *Password* - password for FIX Session authentication. Empty by default.
- -encrypt_method *Flag* - encrypt method. Default: false.
- -start_time *Time* - local time to start the session (HH:MM). Empty by default.
- -terminate_time *Time* - local time to terminate the session (HH:MM). Empty by default.
- -source_ip_address *Address* - the expected value of the source IP address. Empty by default.
- -intraday_logout_tolerance *Flag* - an option not to reset sequence numbers after logout. Default: false.
- -force_seqnum_reset *Flag* - use ResetSeqNumFlag to reset seqNum. Default: false.
- -recreate_on_logout *Flag* - defines whether the session must be recreated on logout. Default: false.
- -force_reconnect *Flag* - defines whether the session will reconnect. Default: false.
- -ignore_seqnum_too_low_at_logon *Flag* - When it is set to “true”, the session continues with received seqNum. Default: false.
- -reject_message_while_no_connection *Flag* - When it is set to “true”, the message that wasn't sent since the time interval will be rejected. Default: false.
- -remote_host *Host_Name* - network address of the computer, to which connection is established. REQUIRED.
- -remote_port *Port_Number* - port's network number on the computer, to which connection is established. REQUIRED.
- -hbi *HBI_Value* - the HeartBeat interval. REQUIRED.

- *-start_session* - starts FIX session using the properties file
  Parameters:
  - -host *Host_Name* - the FIX Edge host name. Default: localhost.
  - -port *Port_Number* - the FIX Edge XML Socket monitor port number. Default: 9211.
  - -sender *Sender_Name* - the sender ID. REQUIRED.
  - -target *Target_Name* - the target ID. REQUIRED.

- *-change_session* - changes FIX session parameter
  Parameters:
  - -host *Host_Name* - the FIX Edge host name. Default: localhost.
  - -port *Port_Number* - the FIX Edge XML Socket monitor port number. Default: 9211.
  - -sender *Sender_Name* - the sender ID. REQUIRED.
  - -target *Target_Name* - the target ID. REQUIRED.
  - -sender_location_id *Sender_ID* - the sender location ID. Empty by default.
  - -target_location_id *Target_ID* - the target location ID. Empty by default.
  - -sender_sub_id *Sender_Sub_ID* - the sender sub ID. Empty by default.
  - -target_sub_id *Target_Sub_ID* - the target sub ID. Empty by default.
  - -inseqnum *Seq_Number* - a new input sequence number. Default: 0.
  - -outseqnum *Seq_Number* - a new output sequence number. Default: 0.
  - -terminate_time *Time* - local time to terminate the session (HH:MM). Empty by default.
  - -intraday_logout_tolerance *Flag* - an option not to reset sequence numbers after logout. Default: false.
  - -recreate_on_logout *Flag* - defines whether the session must be recreated on logout. Default: false.
  - -force_reconnect *Flag* - defines whether the session will reconnect. Default: false.
  - -reject_message_while_no_connection *Flag* - When it is set to “true”, the message that wasn't sent since the time interval will be rejected. Default: false.

- *-restart or -r* - restarts the registered FIX session
  Parameters:
  - -host *Host_Name* - the FIX Edge host name. Default: localhost.

  - -port *Port_Number* - the FIX Edge XML Socket monitor port number. Default: 9211.
  - -sender *Sender_Name* - the sender ID. REQUIRED.
  - -target *Target_Name* - the target ID. REQUIRED.
  - -inseqnum *Seq_Number* - a new input sequence number. Default: 0.
  - -outseqnum *Seq_Number* - a new output sequence number. Default: 0.

- *-delete or -d* - deletes the registered FIX session
  Parameters:
  - -host *Host_Name* - the FIX Edge host name. Default: localhost.
  - -port *Port_Number* - the FIX Edge XML Socket monitor port number. Default: 9211.
  - -sender *Sender_Name* - the sender ID. REQUIRED.
  - -target *Target_Name* - the target ID. REQUIRED.
  - -sendlogout *Flag* - defines whether a logout must be sent. Default: true.

- *-reload_bl* - reloads Business Layer
  Parameters:
  - -host *Host_Name* - the FIX Edge host name. Default: localhost.
  - -port *Port_Number* - the FIX Edge XML Socket monitor port number. Default: 9211.
  - -reload_history *Flag* - reload histories during BL reload. Default: true

- *-to_backup* - switches FIX session to backup connection
  Parameters:
  - -host *Host_Name* - the FIX Edge host name. Default: localhost.
  - -port *Port_Number* - the FIX Edge XML Socket monitor port number. Default: 9211.
  - -sender *Sender_Name* - the sender ID. REQUIRED.
  - -target *Target_Name* - the target ID. REQUIRED.

- *-shutdown or -s* - stops the instance of FIX Edge
  Parameters:
  - -host *Host_Name* - the FIX Edge host name. Default: localhost.
  - -port *Port_Number* - the FIX Edge XML Socket monitor port number. Default: 9211.

- *-stat* - shows FIX Edge statistics
  Parameters:
  - -host *Host_Name* - the FIX Edge host name. Default: localhost.
  - -port *Port_Number* - the FIX Edge XML Socket monitor port number. Default: 9211.

- *-online_sessions_status* - shows FIX Edge online sessions status
  Parameters:
  - -host *Host_Name* - the FIX Edge host name. Default: localhost.
  - -port *Port_Number* - the FIX Edge XML Socket monitor port number. Default: 9211.

- *-online_sessions_list* - shows FIX Edge online sessions list
  Parameters:
  - -host *Host_Name* - the FIX Edge host name. Default: localhost.
  - -port *Port_Number* - the FIX Edge XML Socket monitor port number. Default: 9211.

- *-session_stat* - shows FIX Edge session statistics
  Parameters:
  - -host *Host_Name* - the FIX Edge host name. Default: localhost.
  - -port *Port_Number* - the FIX Edge XML Socket monitor port number. Default: 9211.
  - -sender *Sender_Name* - the sender ID. REQUIRED.
  - -target *Target_Name* - the target ID. REQUIRED.

- *-offline_sessions* - shows offline sessions
  Parameters:
  - -host *Host_Name* - the FIX Edge host name. Default: localhost.
  - -port *Port_Number* - the FIX Edge XML Socket monitor port number. Default: 9211.

- *-exec_xml* - sends XML text to adaptor for executing
  Parameters:
  - -host *Host_Name* - the FIX Edge host name. Default: localhost.
  - -port *Port_Number* - the FIX Edge XML Socket monitor port number. Default: 9211.
  - -xml *Text* - defines instructions in XML format. REQUIRED.
  - -adaptor *Adaptor_Name* - defines the Adaptor that must execute instructions. REQUIRED.

Uninstallation

To uninstall FIX Edge, go to the Control Centre, select “Add or remove programs” and remove B2B FIX Edge.
To remove FIX Edge manually, you need to stop all FIX Edge instances, uninstall FIX Edge services and FECC service and remove FIX Edge directory.
Also, uninstallation link is accessible from ‘Start Menu’.

FAQ

1. How to configure FIX Edge to create a session automatically on application start?
2. How to prevent FIX Edge from automatic accepting of incoming session initiators?
3. How to setup FIX Edge to accept only incoming sessions, which contain a correct user name and password.
4. Is it possible to set IntradayLogoutTolerance for the particular session?
5. How can I make sure that a port is not currently in use by other application on my machine?

Q1. How to configure FIX Edge to create a session automatically on application start?
It is possible to create FIX sessions (initiators and acceptors) on start. Set the following parameters in the properties file (using a text editor or the FIXICC control application):

- FixLayer.FixEngine.Sessions = TestAcceptor
- FixLayer.FixEngine.Session.TestAcceptor.SenderCompID
- FixLayer.FixEngine.Session.TestAcceptor.TargetCompID
- FixLayer.FixEngine.Session.TestAcceptor.Role
- FixLayer.FixEngine.Session.TestAcceptor.Version

Note that all changes in the properties file are applied only after restart.

Q2. How to prevent FIX Edge from automatic accepting of incoming session initiators?
Refer to *UnregisteredAcceptor.CreateSession* property. If it is turned off, ony pre-created session initiators will have the corresponding session-acceptors.

Q3. How to setup FIX Edge to accept only incoming sessions, which contain a correct user name and password.
First of all, it is necessary to turn off automatic creation of sessions-acceptors (refer to UnregisteredAcceptor.CreateSession property in the engine.properties). Then corresponding session-acceptors must be setup for the on-start creation with a correct username and password. See the following properties:

- FixLayer.FixEngine.Sessions = TestAcceptor
- FixLayer.FixEngine.Session.TestAcceptor.SenderCompID
- FixLayer.FixEngine.Session.TestAcceptor.TargetCompID
- FixLayer.FixEngine.Session.TestAcceptor.Role
- FixLayer.FixEngine.Session.TestAcceptor.Version
- FixLayer.FixEngine.Session.TestAcceptor.Username
- FixLayer.FixEngine.Session.TestAcceptor.Password

By default, when a session is correctly terminated, it is removed. Setting

FixLayer.FixEngine.Session.TestAcceptor.RecreateOnLogout to “true” will allow FIX Edge to accept the correspondent session-initiator more than once.

Q4. Is it possible to set IntradayLogoutTolerance for the particular session?
Yes. Refer to *FixLayer.FixEngine.Session.*TestAcceptor*.IntradayLogoutTolerance* property.

Q5. How can I make sure that a port is not currently in use by other application on my machine?
You can get the list of all busy ports using the instruction ‘netstat -na’.

Troubleshooting

	Problem	Resolution
1.	FIX Edge service does not start. The following message appears in FixEdgeInit.log file: "Cannot open logging file './log/FixEdge.log' for File Device."	Paths are not configured properly. Please make sure that *log.file.RootDir*, *FixEdge.RootDir*, *EngineRoot* properties are set correctly and contain a full path to the FIX Edge root directory.
2.	FIX Edge service does not start. The following message appears in Windows Event Log as well as in FIX Edge log file: "Unable to initialize EngineAdaptor. Reason: Please check the FIX Engine's settings: Cannot open the file "engine.license" (the "LicenseFile" property) : The system cannot find the file specified. (2)" .	Most probably, the engine.license file is not installed in the correct location. The Engine license file is searched by default in the FIX Edge root directory.

Glossary

FIX Protocol—The Financial Information eXchange (FIX) Protocol is a messaging standard developed specifically for the real-time electronic exchange of securities transactions. FIX is a public-domain specification owned and maintained by FIX Protocol, Ltd.

FIX Edge—Server application that provides FIX connectivity for standalone client applications.

FIXICC – FIX Integrated Control Centre is a Java stand-alone application with provides monitoring and administration capabilities out-of-the-box for FIX Edge and any application embedding FIX Antenna C++, FIX Antenna Java, FIX Antenna .Net.

FIX Session—A FIX Session is comprised of one or more FIX Connections over a period of time, usually intraday, meaning that the FIX Session may span multiple logins.

FIX Session-acceptor—Session that waits for incoming logon right after creation and then responds with confirming logon. When connection is lost, session-acceptor switches to the state, in which it waits for reconnect logon message.

FIX Session-initiator—Session that initiates connection by sending logon message and waits for the confirming logon. It is also responsible for session reconnection i.e. sends logon to restore the session.

FIX Connection—A FIX Connection is comprised of three parts: logon, message exchange, and logout.

SOD—Start Of Day. It is a standard practice across the industry that FIX sessions with different remote sites (aka counterparties) are scheduled to be initiated at a specified time of day.

EOD—End Of Day. It is a standard practice across the industry that FIX sessions with different remote sites (aka counterparties) are scheduled to be terminated at a specified time of day.

ClientID—A string that uniquely identifies the client application within the Server. ClientID is permanent and assigned by the System Administrator. It is used on Transport Level for client identification as well as on Business Level for message routing.

Counterparty—The corresponding side of the FIX Session, e.g. an exchange, ECN, investment bank with which the Server corresponds via FIX messages.

Sequences Number—the ordered message identifier of message controlled by FIX layer. The SN provides both-side synchronization of the session to establish the successful message transmission. The special session modes resolve problems with sequence gaps, transmission losses using SN of messages.

Incoming message—A message from counterparty to Customer.

Outgoing message—A message from Customer to counterparty.

Transport adaptor (TA) —Transport Adaptor is a plug-in (library) which provides support for a particular transport protocol (e.g. Simple Socket, CORBA)

Contact Us

sales@btobits.com
Phone: +1-888-378-0666

Global Headquarters
US Client Support and Delivery Center

EPAM Systems, Inc
41 University Drive
Suite 202
Newtown, PA 18940
Phone: +1-267-759-9000
Fax: +1-267-759-8989