Configuring Project Components for Oracle® Java CAPS Communication Adapters

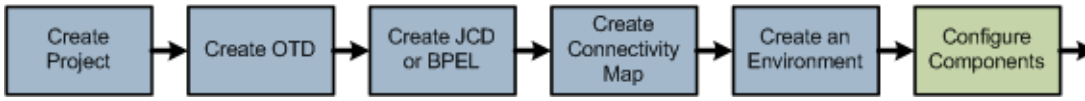
Part No: 821–2589
December 2011

ORACLE®

120126@25097

Contents

Configuring Java CAPS Project Components for Communication Adapters



What You Need to Do

All adapters contain a unique set of default configuration parameters. After the adapters are established and an External System is created in the Project's Environment, the adapter parameters are modified for your specific system. These topics describes how to set the connectivity map properties for Java CAPS Communication Adapters.

- "Configuring Java CAPS Adapter Connectivity Map Properties" on page 7

What You Need to Know

These topics, listed by type, describe the Connectivity Map component properties and Connectivity Map property values for Java CAPS Communication Adapters.

SNA Inbound Adapters

- "SNA Inbound Adapter Connectivity Map Properties" on page 8

Batch Adapters

The Batch Adapter is described in a separate book, *Oracle Java CAPS Adapter for Batch User's Guide*.

CICS Adapters

- "Configuring CICS Adapter Connectivity Map Properties" on page 17

e-Mail Adapters

- "e-Mail Inbound Adapter Connectivity Map Properties" on page 21

File Adapters

- "File Adapter Inbound Connectivity Map Properties" on page 22
- "File Adapter Outbound Connectivity Map Properties" on page 23

HTTPS Adapters

- "HTTPS Adapter Connectivity Map Properties" on page 25

IMS Adapters

- "IMS Adapter Connectivity Map Properties" on page 26

LDAP Adapters

- "LDAP Adapter Connectivity Map Properties" on page 27

MSMQ Adapters

- "MSMQ Adapter Inbound Connectivity Map Properties" on page 39
- "MSMQ Adapter Outbound Connectivity Map Properties" on page 43

TCP/IP Adapters

- "TCP/IP Adapter Inbound Connectivity Map Properties" on page 47
- "TCP/IP Adapter Outbound Connectivity Map Properties" on page 60

TCP/IP HL7V2 and HL7V3

The TCP/IP HL7 Adapter is described in a separate book, *Oracle Java CAPS Adapter for TCP/IP HL7 User's Guide*.

Related Topics

- "Configuring Java CAPS Adapter Connectivity Map Properties" on page 7
- *About Oracle Java CAPS Communication Adapters*
- *Designing with Oracle Java CAPS Communication Adapters*
- *Developing OTDs for Oracle Java CAPS Communication Adapters*
- *Configuring Environment Components for Oracle Java CAPS Communications Adapters*

Configuring Java CAPS Adapter Connectivity Map Properties

This topic describes how to set the connectivity map properties for Java CAPS Adapters.

Adapter configuration parameters are modified from the following locations:

- **Connectivity Map**: These parameters most commonly apply to a specific component adapter, and may vary from other adapters (of the same type) in the Project.
- **CAPS Environment**: These parameters are commonly global, applying to all adapters (of the same type) in the Project. The saved properties are shared by all adapters in the External System window.
- **Collaboration or Business Process**: Adapter properties may also be set from your Collaboration or Business Process, in which case the settings will override the corresponding properties in the adapter's Connectivity Map configuration. Any properties that are not overridden retain their configured default settings.

Configuring the Adapter Connectivity Map Properties

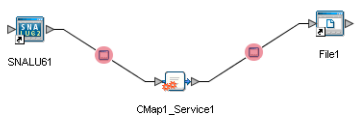
When you connect an External Application to a Collaboration, Netbeans IDE automatically assigns the appropriate adapter to the link. Each adapter is supplied with a template containing default configuration properties that are accessible on the Connectivity Map.

Adapters can be configured for inbound and/or outbound modes in a Connectivity Map.

▼ To Configure the Inbound Adapter Properties

- **On the Connectivity Map, double-click the Adapter icon.**

FIGURE 1 Connectivity Map with Components - Inbound

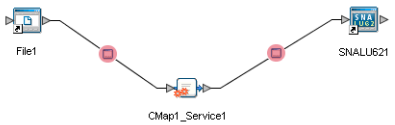


The adapter Properties window appears, displaying the default properties for the Inbound adapter.

▼ To Configure the Outbound Adapter Properties

- **On the Connectivity Map, double-click the Adapter icon.**

FIGURE 2 Connectivity Map with Components - Outbound



The adapter Properties window appears, displaying the default properties for the Outbound adapter.

SNA Inbound Adapter Connectivity Map Properties

The SNA Inbound Adapter Connectivity Map properties are grouped into the following categories.

Connectivity Map Inbound Adapter General Settings

The following table lists and describes the Inbound Adapter General Settings.

TABLE 1 Inbound Adapter—General Settings

Name	Description	Required Value
Scope of State	Defines the scope of the State object, which is an OTD sub-node. The valid options for this parameter are: ■ **Connection Level**: The State has the same life cycle as the connection. ■ **Resource Adapter Level**: The State has the same life cycle as the resource adapter. The life terminates when the resource adapter is recycled. ■ **OTD Level**: The State has the same life cycle as the OTD object. This scope represents the life cycle of the State. The life terminates when the collaboration finishes.	Connection Level, Resource Adapter Level, or OTD Level. The default is Connection Level.

Connectivity Map Inbound Adapter SNA Settings

The following table lists and describes the Inbound Adapter SNA Settings.

TABLE 2 Inbound Adapter—SNA Settings

Name	Description	Required Value
Packet Size	The number of bytes per packet of data. This number also determines the size of the buffers.	A valid numeric value. The default is 1024.
Timeout	Specifies the milliseconds of pause before receiving a response from a server following a sent request.	A valid numeric value. The default is 1000.
Initialize Conversation	Specifies how the adapter will establish a SNA conversation. Options are: ■ **True**: The adapter will initialize SNA conversations as an invoking TP. ■ **False**: The adapter will accept SNA conversations as an invoking TP.	Select True or False. The default is False.
Deallocation Type	Specifies the type of deallocation required at the end of a conversation when a shutdown is issued. Please refer to your SNA documentation for more information.	Select one of the following four options: ■ 0 - SYNC_LEVEL. ■ 1 - FLUSH ■ 2 - CONFIRM ■ 3 - ABEND The default is 0 - SYNC_LEVEL.

TABLE 2 Inbound Adapter—SNA Settings *(Continued)*

Name	Description	Required Value
Synchronization Level	Specifies the synchronization level parameter (CM_SYNC_LEVEL). Please refer to your SNA manual for more information.	Select one of the following: ■ 0 - None ■ 1 - Confirm The default is 0 - None.
Custom Handshake Class Name	Defines your SNA handshake logic (see Appendix B to deploy a custom handshake class).	A fully qualified class name such as `com.abc.MyClass`. The class must implement the interface `com.stc.connector.snalu62.api.snaCustomerHandshake`. No value (leaving the property blank) indicates that no SNA conversation handshake logic is defined. Instead, a built-in standard handshake logic is used.

Connectivity Map Inbound Adapter Connection Establishment

The Inbound Adapter Connection Establishment properties are included in the table.

TABLE 3 Inbound Adapter—Connection Establishment

Name	Description	Required Value
Max Connection Retry	Specifies the maximum number of retries to establish a connection upon failure to acquire one.	A valid numeric value. The default is 3.
Retry Connection Interval	Specifies the milliseconds of pause before each attempt to reaccess the SNA LU62 destination. This setting is used in conjunction with the Max Connection Retry setting. For example, In the event that the adapter cannot connect to the SNA destination, the adapter will try to reconnect three times in 30 second intervals when the Connection Retries value is 3 and the Connection Retry Interval is 30000.	A valid numeric value. The default is 30000.

Connectivity Map Inbound Adapter Inbound Connection Management

The following table lists and describes the Inbound Adapter Inbound Connection Management properties.

TABLE 4 Inbound Adapter—Inbound Connection Management

Name	Description	Required Value
Max Connection Pool Size	Defines the maximum number of concurrent connections for the particular listener/monitor over the specified SNALU62 destination. 0 (zero) indicates that there is no maximum.	A valid numeric value. The default is 50.
Scope of Connection	Defines the scope of the accepted connection used by the adapter. Options are: ■ **Collaboration Level**: The connection will be closed once the execution of the Collaboration is completed. The connection has the same life cycle as the Collaboration. ■ **Resource Adaptor Level**: The resource adapter will close the connection upon closure request. The connection may remain live across multiple executions of the Collaboration.	Select Collaboration Level or Resource Adaptor Level. The default is Resource Adaptor Level.

Connectivity Map Inbound Adapter Inbound Schedules

This section describes the following,

- "Listener Schedule" on page 11
- "Service Schedule" on page 13

Listener Schedule

Listener Schedule properties specify the schedule that the server must wait for the new client connection establishment request. This schedule is for the listener/monitor. The following table lists and describes the Listener Schedule properties.

TABLE 5 Inbound Schedules—Listener Schedule

Name	Description	Required Value
Scheduler	Specifies the scheduler type for this inbound communication. Options are: ■ **Timer Service**: The task is scheduled according to the Schedule Type, Delay, Period, and At Fixed Rate values. ■ **Work Manager**: The work is scheduled according to the Schedule Type, Delay, and Period values. If your container does not support JCA Work Management (prior to JCA1.5), select Timer Service.	Select Timer Service or Work Manager. The default is Work Manager.
Schedule Type	Defines the type of schedule for inbound communication. Repeated indicates a task is scheduled for repeated execution at regular intervals defined by the parameter Period.	The configured default is Repeated. **Note –** This value cannot be changed.
Delay	Specifies the delay in milliseconds before a task is executed. For further details, refer to the SNA Adapter Javadoc.	A valid numeric value. The default is **0**.
Period	Specifies the regular interval in milliseconds between successive task executions. This parameter is used in conjunction with the Schedule Type parameter when set to Repeated.	A valid numeric value. The default is 100.
At Fixed Rate	Used in conjunction with the Repeated setting for the Schedule Type parameter and the Timer Service type of Scheduler. Options are: ■ **True**: Denotes a fixed rate. Each execution is scheduled relative to the scheduled time of the initial execution. If an execution is delayed for any reason, two or more executions will occur in rapid succession to return to the preset execution schedule. Overall, the frequency of executions will be exactly the reciprocal of the specified period. ■ **False**: Denotes a fixed delay. Each execution is scheduled relative to the actual execution time of the previous execution. If an execution is delayed for any reason, subsequent executions are delayed as well. Overall, the frequency of executions will generally be lower than the reciprocal of the specified period.	Select True or False. The default is False.

Service Schedule

The following table lists and describes the Inbound Adapter Service Schedule properties.

TABLE 6 Inbound Schedules—Service Schedule

Name	Description	Required Value
Scheduler	Specifies the scheduler type for this inbound communication. Options are: ■ **Timer Service**: The task is scheduled according to the Schedule Type, Delay, Period, and At Fixed Rate values. ■ **Work Manager**: The work is scheduled according to the Schedule Type, **Delay**, and Period values. If your container does not support JCA Work Management (prior to JCA1.5), select Timer Service.	Select Timer Service or Work Manager. The default is Work Manager.
Schedule Type	Defines the type of schedule for inbound communication. Options are: ■ **One Time**: A task is scheduled for a one-time execution. ■ **Repeated**: A task is scheduled for repeated execution at regular intervals defined by the parameter Period.	Select One Time or Repeated. The default is Repeated.
Delay	Specifies the delay in milliseconds before a task is executed. For further details, refer to the SNA Adapter Javadoc.	A valid numeric value. The default is 0.
Period	Specifies the regular interval in milliseconds between successive task executions. This parameter is used in conjunction with the Schedule Type parameter when set to Repeated.	A valid numeric value. The default is 100.

TABLE 6 Inbound Schedules—Service Schedule *(Continued)*

Name	Description	Required Value
At Fixed Rate	Used in conjunction with the Repeated setting for the Schedule Type parameter and the Timer Service type of Scheduler. Options are: ■ **True**: Denotes a fixed rate. Each execution is scheduled relative to the scheduled time of the initial execution. If an execution is delayed for any reason, two or more executions will occur in rapid succession to return to the preset execution schedule. Overall, the frequency of executions will be exactly the reciprocal of the specified period. ■ **False**: Denotes a fixed delay. Each execution is scheduled relative to the actual execution time of the previous execution. If an execution is delayed for any reason, subsequent executions are delayed as well. Overall, the frequency of executions will generally be lower than the reciprocal of the specified period.	Select True or False. The default is False.

Connectivity Map Outbound Adapter General Settings

The following table lists and describes the Outbound Adapter General Settings.

TABLE 7 Outbound Adapter—General Settings

Name	Description	Required Value
Scope of State	Defines the scope of the State object, which is an OTD sub-node. The valid options for this parameter are: ■ **Connection Level**: The State has the same life cycle as the connection. ■ **Resource Adapter Level**: The State has the same life cycle as the resource adapter. The life terminates when the resource adapter is recycled. ■ **OTD Level**: The State has the same life cycle as the OTD object. This scope represents the life cycle of the State. The life terminates when the collaboration finishes.	Connection Level, Resource Adapter Level, or OTD Level. The default is Connection Level.

Connectivity Map Outbound Adapter SNA Settings

The following table lists and describes the Outbound Adapter SNA Settings.

TABLE 8 Outbound Adapter—SNA Settings

Name	Description	Required Value
Packet Size	The number of bytes per packet of data. This number also determines the size of the buffers.	A valid numeric value. The default is 1024.
Timeout	Specifies the milliseconds of pause before receiving a response from a server following a sent request.	A valid numeric value. The default is 1000.
Initialize Conversation	Specifies how the adapter will establish a SNA conversation. Options are: ■ **True**: The adapter will initialize SNA conversations as an invoking TP. ■ **False**: The adapter will accept SNA conversations as an invokable TP.	Select True or False. The default is True.
Deallocation Type	Specifies the type of deallocation required at the end of a conversation when a shutdown is issued. Please refer to your SNA documentation for more information.	Select one of the following four options: ■ 0 - SYNC_LEVEL. ■ 1 - FLUSH ■ 2 - CONFIRM ■ 3 - ABEND The default is 0 - SYNC_LEVEL.
Synchronization Level	Specifies the synchronization level parameter (CM_SYNC_LEVEL). Please refer to your SNA documentation for more information.	Select one of the following two options: ■ 0 - None ■ 1 - Confirm. The default is 0 - None.
Custom Handshake Class Name	Defines your SNA handshake logic (see Appendix B to deploy a custom handshake class).	A fully qualified class name such as `com.abc.MyClass`. The class must implement the interface `com.stc.connector.snalu62.api.snaCustomerHandshake`. No value (leaving the property blank) indicates that no SNA conversation handshake logic is defined. Instead, a built-in standard handshake logic is used.

Connectivity Map Outbound Adapter Connection Establishment

The following table lists and describes the Outbound Adapter Connection Establishment properties.

TABLE 9 Outbound Adapter—Connection Establishment

Name	Description	Required Value
Connection Mode	Specifies how or when a connection will become available. Options are: ■ **Automatic**: The adapter will establish a SNA conversation automatically. ■ **Manual**: The SNA conversation will become available to you only when you manually call the OTD function startConversation() from the Collaboration; the conversation will become unavailable when you call the OTD function endConversation(). **Note –** The OTD functions `startConversation()` and `endConversation()` are expected for Manual mode only. Automatic mode does not allow you to call them explicitly.	Select Automatic or Manual. The default is Automatic.
Max Connection Retry	Specifies the maximum number of retries to establish a connection upon failure to acquire one.	A valid numeric value. The default is 3.
Retry Connection Interval	Specifies the milliseconds of pause before each attempt to reaccess the SNA LU62 destination. This setting is used in conjunction with the Max Connection Retry setting. For example, In the event that the adapter cannot connect to the SNA destination, the adapter will try to reconnect three times in 30 second intervals when the Connection Retries value is 3 and the Connection Retry Interval is 30000.	A valid numeric value. The default is 30000.

TABLE 9 Outbound Adapter—Connection Establishment *(Continued)*

Name	Description	Required Value
Always Create New Connection	Specifies whether to ALWAYS attempt to create a new connection for a connection establishment request. Options are: ■ **True**: The adapter will always attempt to create a new connection without trying to match connection. ■ **False**: The adapter will attempt to match an existing connection.	Select True or False. The default is False.
Auto Reconnect Upon Matching Failure	Specifies whether or not to make an attempt to reconnect automatically after getting a matched connection from a container. Options are: ■ **True**: The adapter will discard the invalid matched connection and will attempt to establish another connection automatically. ■ **False**: The adapter will not attempt to establish a new connection automatically. Instead, control will be deferred to your business rules which will detect this type of failure and perform the desired operations accordingly.	Select True or False. The default is True.
Auto Disconnect Connection	Specifies whether the adapter disconnects automatically after the work on the connection is completed. Options are: ■ **True**: The adapter connection will be disconnected and it will not be re-used. ■ **False**: The connection will be left for reuse.	Select True or False. The default is False.

Configuring CICS Adapter Connectivity Map Properties

The CICS Adapter configuration parameters, accessed from the Connectivity Map, are organized into the following sections:

- "CICS Connector" on page 17
- "CICS Client" on page 18
- "Connection Mode" on page 20

CICS Connector

The Connector section of the CICS Connectivity Map properties contains the top-level parameters displayed in this table:

TABLE 10 Connectivity Map - Connector Properties

Name	Description	Required Value
Type	Specifies the connector type.	Enter CICS. The value always defaults to CICS for CICS connections.
Connection Transport	Specifies the underlying connection transport used by the CICS Adapter to send requests to and get responses from a CICS region.	Select one of the following underlying connection transports: ■ CICS Listener ■ CICS Transaction Gateway (specifies the IBM CICS Transaction Gateway) CICS Listener is the default.
Class	Specifies the class name of the CICS Client connector object.	The (class) package name for the CICS Client connector object. The default is `com.stc.adapters.cics.CicsClientConnector`.
Property.Tag	Specifies the data source identity. This parameter is required by the current `EBobConnectorFactory`.	The data source package name.

CICS Client

The CICS Client section of the CICS Connectivity Map properties contains the top-level parameters displayed in the following table.

TABLE 11 CICS Client Connectivity Map Properties

Name	Description	Required Value
ECI call type	Specifies whether the ECI call type is Synchronous. Synchronous calls wait for the transaction to complete, then return the contents of the COMMAREA. Only Synchronous calls are supported.	Synchronous is the configured default.
CICS Program	Specifies the CICS program to be run on the server. Maximum length is eight characters.	A CICS program name, eight characters or less.

TABLE 11 CICS Client Connectivity Map Properties *(Continued)*

Name	Description	Required Value
CICS TransId	CTG specific. Specifies the ID of a CICS transaction. Maximum length is four characters. Attributes are dependent upon the value set for `Use TransId as ECI_TPN`, as follows: ■ If `EciTPN` is set to false, the value of the transid is stored in EIBTRNID for the duration of the LINK to the program specified in the Program parameter. The called program runs under the mirror transaction `CPMI`, but is linked to under the Transid transaction name. This name is available to the called program for querying the transaction ID. Some servers use the transaction ID to determine security and performance attributes for the called program. ■ If `EciTPN` is set to true, the Transid will be interpreted as the `ECI_TPN` transid, a transaction that will be used in the server to process the ECI request. This transaction must be defined in the server as a CICS mirror transaction. If the ECI request is extended, this parameter has a meaning only for the first request.	A CICS transaction ID of four characters or less.
Use TransId as ECI_TPN	This is specific to CTG, and specifies whether the TransId is interpreted as ECI_TPN or if the called program runs under the default mirror transaction CPMI. Options are: ■ **True**: Indicates that the TransId is interpreted as ECI_TPN. ■ **False**: Indicates that the called program runs under the default mirror transaction CPMI, and is linked to under the TransId (if present).	Select True or False. False is the default.
COMMAREA Length	Specifies the length (in bytes) of the COMMAREA passed to the ECI.	A number indicating the byte length of the COMMAREA. **Note –** When using the CICS Transaction Gateway transport, data sent to CICS must be padded with spaces, if necessary, to match the full size of the commarea.

TABLE 11 CICS Client Connectivity Map Properties *(Continued)*

Name	Description	Required Value
ECI extend mode	Specifies whether a logical unit of work is terminated at the end of a call.	Select Yes or No. Yes indicates that the work unit is terminated at the end of a call. The default is No.
ECI LUW Token	CTG specific. Specifies an integer used to identify the logical unit of work (LUW) to which a call belongs. This must be set to 0 (zero) at the start of an LUW, even if the LUW is to be extended. The ECI updates the value upon the first (or only) call of the LUW. If the LUW is to be extended, this value is used as input to any subsequent calls associated with the same LUW. If the return code is not ECI_NO_ERROR, and a call is ending or continuing an existing LUW, then this field is used to report the state of the LUW as follows: ■ A code of 0 (zero) indicates that the LUW has ended and its updates have been backed out. ■ Any non-zero code indicates the current input value. It also indicates that the LUW is continuing, and updates are still pending. See the **Logical units of work in ECI** table in the **CICS Transaction Gateway: Programming Guide** for more information.	An integer used to identify the ECI logical unit of work.
Encoding	Specifies the canonical name for the encoding set.	The canonical name for any encoding set supported by JRE 1.1.8 (contained in rt.jar and i18n.jar). Examples are ASCII and Cp500 (EBCDIC). When running the CICS Adapter on a z/OS platform, set the Encoding value to Cp500.

Connection Mode

The **Connection Mode** section of the CICS Connectivity Map properties contains the top-level parameters displayed in the following table.

TABLE 12 Connectivity Map Properties - Connection Mode Section

Name	Description	Required Value
CICS Connection Mode	Specifies whether a physical connection is established when an external connection is instantiated. The options are, **Automatic:** Establishes a physical connection when an external connection is instantiated. **Manual:**Does not automatically establish a physical connection when an external connection is instantiated. If a physical connection is not automatically established, a physical connection must be established from the Collaboration (for example, by calling the `connect()` method).	Select Automatic or Manual. The configured default is Automatic.

e-Mail Inbound Adapter Connectivity Map Properties

The e-Mail Adapter configuration parameters, accessed from the Connectivity Map, are contained in the section "Polling Setting" on page 21.

Note – Some e-Mail adapter properties can also be set from your Collaboration. Properties set from the Collaboration override the corresponding properties in the adapter's configuration file. Any properties that are not set from the Collaboration retain their configured default settings.

Polling Setting

The Polling Setting section of the e-Mail adapter Connectivity Map properties contains the top-level parameter displayed in the following table.

TABLE 13 Connectivity Map - Polling Setting

Name	Description	Required Value
Polling Interval	Specifies the interval (in milliseconds) at which the email source file is polled for new incoming email messages.	A number indicating the polling interval in milliseconds. The configured default is 5000 (5 seconds).

File Adapter Inbound Connectivity Map Properties

The inbound File Adapter configuration properties, accessed from the Connectivity Map, are in the File Adapter Inbound section.

Parameter Settings — File Adapter Inbound

The Parameter Settings section of the inbound File Adapter Connectivity Map properties contains the top level parameters displayed in the following table.

TABLE 14 Inbound File Adapter Connectivity Map Properties - Parameter Settings

Name	Description	Required Value
Input file name	Specifies the file mask for input data files.	A file mask. The default is `input*.txt`. You can provide only an extension with an asterisk (), for example, `*.txt`, to allow all files with that extension. An input file's extension is renamed to `.~in` after it is picked up. **Note –** If a file with the same name as the rename name exists, for example, `input1.txt.~in` , picking up the input file fails. You must ensure no files with the rename names exist before the files are processed by the inbound file poller.
Polling interval	The number of milliseconds the adapter waits between poll attempts of the input directory. The polling interval and the MDB pool size can be "tuned" based on the expected volume and frequency of incoming messages.	An integer; the acceptable range is an integer from 2 to 99999, inclusive, and the default is 5000 (5000 milliseconds or 5 seconds). It is not advised to enter a value less than 5 seconds. **Caution –** There is no error message if you enter an incorrect value.
Input type	Specifies the type of input file, for example, bytes.	A file type. The default is Bytes, the only valid value.

TABLE 14 Inbound File Adapter Connectivity Map Properties - Parameter Settings *(Continued)*

Name	Description	Required Value
Remove EOL	Specifies whether the adapter excludes the terminating End-Of-Line (EOL) characters from records (messages) sent to a subscriber. This property only applies if the `Multiple records per file` property is set to true. ■ **True**: Enables the feature. ■ **False**: Disables the feature.	Select True or False. The default is False.
Multiple records per file	Specifies whether multiple records (messages) are obtained per file. If this property is set to **True**, multiple records (messages) are generated per line, up to the number specified in the `Maximum bytes per record` property. Any data exceeding the maximum bytes per record size is sent in subsequent messages.	Select True or False. The default is False.
Maximum bytes per record	Specifies the maximum number of bytes per record (message) sent to a subscriber. This property only applies if the `Multiple records per file` property is set to True.	An integer; the acceptable range is an integer from 2 to 99999, inclusive, and the default is 4096. **Caution –** There is no error message if you enter an incorrect value.
Encoding	Specifies the valid encoding names. For more information on Java encoding, refer to the following site: `http://download.oracle.com/javase/1.3/docs/api/java/lang/package-summary.html`	The encoding names.
SerialMode	Specifies whether messages will be handled in serial mode (one after the other) or concurrently. ■ **True:** Indicates that messages are handled in serial mode. ■ **False:** Indicates that messages handled concurrently.	Select True or False. The configured default is True.

File Adapter Outbound Connectivity Map Properties

The Outbound File Adapter configuration properties, accessed from the Connectivity Map, are included in the Parameter Settings section.

Parameter Settings

The Parameter Settings section of the outbound File Adapter Connectivity Map properties contains the top level parameters displayed in the following table.

TABLE 15 Outbound File Adapter Connectivity Map Properties - Parameter Settings

Name	Description	Required Value
Output file name	Specifies the file mask for output data files. The default value contains `%d` in the file name, which is a counter and increments for each new file. Instead of `%d`, you can use any other printf style that takes an integer or long value. For example, you can specify `1%d` or `%012d` In cases where the `Multiple records per file` property is set to false: ■ If no output file exists at the time of the first execution, a new output file is created for each record. ■ If an output file already exists at the time of the first execution, messages are appended to it. **Note –** For more information regarding the printf feature, see the appropriate C language documentation.	An appropriate file name. The default is `output%d.dat`.
Add EOL	Specifies whether the system adds an end-of-line character to each record the adapter sends to the output file. ■ **True**: Indicates that the system will add an EOL to each record. ■ **False**: Indicates otherwise.	Select True or False. The configured default is False.
Multiple records per file	Specifies whether multiple records (messages) can be written to the output file. New messages are appended to the output file. ■ **True**: Indicates that the output file will contain multiple records (messages). ■ **False**: Indicates that each output file contains a single record (message).	Select True or False. The configured default is True.
Encoding	Specifies the valid encoding names. For all valid encodings, please refer to the following site: `http://download.oracle.com/javase/1.3/docs/api/java/lang/package-summary.html`	The encoding names.

HTTPS Adapter Connectivity Map Properties

The HTTPS Adapter Connectivity Map consists of the following categories:

HTTPS Adapter Connectivity Map Properties

The HTTPS Adapter Properties include parameters used by the external system.

TABLE 16 HTTP Adapter—HTTP Settings

Name	Description	Required Value
Allow Cookies	Specifies whether cookies sent from servers are allowed to be stored and sent on subsequent requests. If cookies are not allowed, sessions are not supported.	Select True or False. The default is True.
Accept Type	The default Accept type header value to include when sending a request to the server.	A string. For example: text/html, text/plain, text/xml, and so on. The default is text/*.

HTTPS Server Adapter Connectivity Map Properties

The HTTPS Server Adapter Properties include parameters used by the external system.

TABLE 17 HTTP Server Adapter—HTTP Server External Configuration

Name	Description	Required Value
servlet-url	Specifies the last path component of the HTTPS server servlet URL. The client uses this URL value to access the server. The property value must be the servlet name (for example, `HttpServerServlet`). An example of a valid servlet URL is `http://localhost:18001/Deployment1_servlet_HttpServerServlet/HttpServerServlet`, where, the URL value comprises several components as follows: ■ **App Server**: The name of the machine on which your current Application Server is running. ■ **18001**: The port number (in this case, the Application Server port number). ■ **Deployment1_servlet_HttpServerServlet**: The name of your current Project's Deployment Profile concatenated with **_servlet_HttpServerServlet**. ■ **HttpServerServlet**: The servlet name (equivalent to the **servlet_url** property). **Note –** Set the port number based on the Oracle Enterprise Service Bus properties. By default, it is 18001, but it can be modified. Set the Enterprise Service Bus properties in the Environment from the Services window. The `servlet-url` property does not support LDAP values.	A valid URL.

IMS Adapter Connectivity Map Properties

The following topics list and describe the IMS Adapter's configuration properties, accessed from the Connectivity Map.

- "Connector — IMS Adapter Outbound" on page 26
- "Connection Mode — IMS Adapter Outbound" on page 27

Connector — IMS Adapter Outbound

The Connector section contains the top level parameters contained in this table:

TABLE 18 IMS Adapter connector Parameter Settings

Name	Description	Required Value
Type	Specifies the connector type.	Connection type is IMSClientETD, by default for IMSClientETD connections.
Class	Specifies the class name of the ETD connector object.	A valid package name. The default is `com.stc.adapters.ims.IMSClientETDConnector`

Connection Mode — IMS Adapter Outbound

The Connection Mode section contains the top level parameters contained in this table:

TABLE 19 IMS Adapter Connection Mode Parameter Settings

Name	Description	Required Value
IMS Connection Mode	Specifies how a connection with the external system is established and closed. ■ **Automatic**: Indicates that the connection is automatically established when the collaboration is started, and it keeps the connection alive as needed. OnDemand indicates that the connection will be established on demand, as business rules requiring a connection to the external system are performed. The connection will be closed after the methods are completed. ■ **Manual**: Indicates that the user will explicitly call the connect and disconnect connection methods in their collaboration as business rules. Default is Automatic.	Either a Manual or Automatic setting. Default setting is Manual.

LDAP Adapter Connectivity Map Properties

The LDAP Adapter configuration parameters, accessed from the Connectivity Map, are organized into the following sections:

Connector Section Properties

The LDAP Adapter Connector Section Properties include the following parameters.

TABLE 20 LDAP Adapter— Connector Settings

Name	Description	Required Value
Connector Type	Lists the type of connector	The default is LDAP Connector.
Connector Class	Lists the Connector class.	The default connector class is `com.stc.connector.ldapadapter.LDAPadapterConnection`.

Connection Section Properties

The LDAP Adapter Connection Section Properties allow you to define the connection to the LDAP system.

TABLE 21 LDAP Adapter— Connection Settings

Name	Description	Required Value
Authentication	The authentication to be used (none or simple). Select the desired authentication as follows: ■ **None:** No authentication, that is, an anonymous log-on. If you use this setting, ensure that the LDAP server supports anonymous login. ■ **Simple:** Authentication is based on a user name and password. You must provide the user name and password in the appropriate fields (Principal and Credentials).	Select none or simple. The default is none.
Credentials	The credentials needed when using an authentication mechanism other than anonymous login (authentication = none).	The appropriate credentials, in the form of a valid password.

TABLE 21 LDAP Adapter— Connection Settings *(Continued)*

Name	Description	Required Value
InitialContext Factory	The factory to be used for creating the initial context for the LDAP server. By default the LDAP service provider provided by Oracle, as part of the Java Software Developers' Kit (SDK), is used.	A valid Java factory name; the default is: `com.sun.jndi.ldap.LdapCtxFactory`. It is recommended that you do not change this value unless you want to use an LDAP service provider other than the one provided by Oracle.
Principal	The principal needed when using an authentication mechanism other than anonymous login (authentication = none).	The fully qualified Distinguished Name (DN) of the user, for example: `CN=Administrator,CN=Users, DC=stc,dc=com`
ProviderURL	The URL of the LDAP Server.	A valid URL with the protocol as **ldap**.

Referrals Section Properties

The LDAP Adapter Referrals Section Properties allow you to enter LDAP referral information.

TABLE 22 LDAP Adapter— Referrals Settings

Name	Description	Required Value
Credentials File	The credentials file to be used when following any referrals in the directory. The credentials file is created using the RCF command-line utility.	A valid file and path name available to the Service Bus.
Follow	An indicator of whether referrals returned by an LDAP server must be followed. ■ **Yes**: Follow referrals. ■ **No**: Referrals are not followed.	Select Yes or No. The default is Yes. Enter the desired value as follows:

Additional Referrals Section Notes

A referral is an entity used to redirect a client's request to another server. A referral contains the names and locations of other objects. It is sent by the server to indicate that the information the client has requested can be found at another location (or locations), possibly at another server or several servers.

When you execute a search operation, you may encounter a referral entry, which is just a pointer to where that information can be found. The pointer is usually in a form similar to the **Provider URL** configuration of the Adapter.

It consists of the following components:

- Host name
- Port number
- Context name (optional)

You have the following options when you encounter a referral:

- **Ignore**: Ignore the referral.
- **Follow**: Follow the referral, that is, connect to the referred system and continue the search operation.
- **Throw**: Throw a referral exception, which can be caught by the client and action taken as needed.

With the LDAP Adapter, you have the following properties you must set to work with referrals:

- **Credentials File**: Enter a fully qualified path to a file. This file must contain the appropriate referral credentials information (this file has to be generated using the RCF command line utility as explained later in this section).
- **Follow**: It is either Yes or No. Default is Yes.

The scenarios shown in the following table can arise depending on the properties provided for the referrals and the behavior of the Adapter, as explained for each of these scenarios.

TABLE 23 Referral Scenarios

Follow Setting	Credentials File	Adapter Operation
Follow is set to **Yes**.	The credentials file is not provided.	The Adapter uses the original credentials (user name and password) provided for the initial server and tries to connect to the referred system. The connection may fail if the referred system does not have the same credentials.
	The credentials file is provided and has the credentials entry for the referred host.	The connection to the initial server is configured to throw `LdapReferralException` when a referral is encountered which is subsequently caught by Adapter. The Adapter then establishes the connection to the referred system using the credentials information provided in the credentials file.
	The credentials file provided does **not** have the credentials entry for the referred host.	The connection to the initial server is configured to throw `LdapReferralException` when a referral is encountered, which is subsequently caught by the Adapter. The Adapter then establishes the connection to the referred system using an anonymous login. The connection may fail if the referred system does not allow an anonymous login.
Follow is set to **No**.	There is no credentials file.	Referrals are not followed, that is, the Adapter ignores any referral.

To create a credentials file, you can use the Referral Credentials File (RCF) command-line utility.

Note – Running the RCF utility on the command line without any parameters displays how to use the utility.

▼ To Create a Credentials File Using the RCF Utility

1 The file to be used for the RCF utility are located at the following locations:

netbeans_home`\usrdir\modules\ext\ldapadapter\stcldap13.jar`

or

```
<netbeans_home>\usrdir\modules\ext\ldapadapter\
stcldap14.jar
```

2 Copy and paste one of the above files to a folder and run the utility from this folder as follows:

netbeans_home\jdk\bin\java -cp ./stcldap13.jar
com.stc.connector.ldapadapter.utils.RCFUtil

The following menu displays:

```
C:\temp>java -cp ./stcldap13.jar
com.stc.connector.ldapadapter.utils.RCFUtil

Please specify the operation.

---+ RCFUtil +---

Interactive command line utility for creating and managing
file(s) containing credentials information to follow LDAP
referrals. File(s) generated can be used by the Java LDAP Adapter
for following referrals that required credentials different
from those used to create the connection to the initial LDAP
server.

Usage : java com.stc.connector.ldapadapter.utils.RCFUtilOPTIONS       -- <filename>

OPTIONS:

--create Create a new referral credentials file.
--add Add an entry to the referral credentials file.
--list Print a list of entries in the referral credentials file.
--remove Remove an entry from the referral credentials file.
--modify Modify an entry in the referral credentials file.
--decrypt When displaying credentials, decrypt the credentials.
--username <username>  Specify the username; if not specified,
it'll be prompted.
--password <password>  Specify the password; if not specified,
it'll be prompted.
--help Print this usage.

filename:

The full path to the referral credentials file.
```

3 To create a new referral file called `samplercf.txt`, enter the following parameters on the command line:

netbeans_home\jdk\bin\java -cp ./stcldap13.jar
com.stc.connector.ldapadapter.utils.RCFUtil --create -- samplercf.txt

This action requests a user name and password. Enter the user name and password. This user name and password is for protecting the file itself, because the file contains sensitive credential information about other LDAP servers. For example:

```
      C:\temp>c:\JavaCAPS6\netbeans\jdk\bin\java -cp .\stcldap13.jar
com.stc.connector.ldapadapter.utils.RCFUtil
      --create -- samplercf.txt
      Creating file samplercf.txt...
      Enter username >> test
      Enter password >> test
      File created!
```

A message "File created!" appears. The file name here is `samplercf.txt`. The extension does not matter.

▼ To Add Credentials Information To the File

1. **To add LDAP Server connection info to a referral file called `samplercf.txt`, enter the following parameters on the command line:**

   ```
   netbeans_home\jdk\bin\java -cp ./stcldap13.jar
   com.stc.connector.ldapadapter.utils.RCFUtil --add --
    samplercf.txt
   ```

2. **Username and Password are required to access the file. Provide the user name and password given for creating the file previously.**

3. **When the following prompts appear, enter the following information, as indicated:**

4. **Prompts for the host name: Enter the host name.**

5. **Prompts for the port number: Enter the LDAP port number.**

6. **Prompts for the principal: Enter the fully qualified DN of the user.**

7. **Prompts for the password: Enter the password for the DN specified previously.**

   For example:

   ```
   C:\temp>c:\JavaCAPS6\netbeans\jdk\bin\java -cp .\stcldap13.jar
    com.stc.connector.ldapadapter.utils.RCFUtil --add --
    samplercf.txt
   Adding a referral credentials entry...
   Enter username >> test
   Enter password >> test
   Enter LDAP Host >> localhost.stc.com
   Enter LDAP Port >> 389
   Enter the Principal >> cn=Manager,dc=stc,dc=com
   Enter the Credentials >> secret

   Done.
   ```

▼ To View the Contents of the Credentials File

1. **To view LDAP Server connection info in a referral file called samplercf.txt, enter the following parameters on the command line:**

```
<netbeans_home>\jdk\bin\java -cp ./stcldap13.jar
com.stc.connector.ldapadapter.utils.RCFUtil --list --
 samplercf.txt
```

2. **Username and Password are required to access the file. Provide the user name and password given for creating the file previously.**

3. **The entries in the file are listed as shown in the following single-entry example:**

```
1> localhost.stc.com | 389 | cn=Manager,dc=stc,dc=com | l/
ZRt1cfNKc=
```

4. **The password is encrypted. To display the password in its decrypted form add --decrypt to the previous command. The output is as follows:**

```
1> localhost.stc.com | 389 | cn=Manager,dc=stc,dc=com | secret
```

For example:

```
C:\temp>c:\JavaCAPS6\netbeans\jdk\bin\java -cp .\stcldap13.jar
 com.stc.connector.ldapadapter.utils.RCFUtil --list --
 samplercf.txt
Listing entries in the referral credentials file...
Enter username >> test
Enter password >> test
1> localhost.stc.com | 389 | cn=Manager,dc=stc,dc=com | l/
ZRt1cfNKc=

C:\temp>c:\JavaCAPS6\netbeans\jdk\bin\java -cp .\stcldap13.jar
 com.stc.connector.ldapadapter.utils.RCFUtil --list --decrypt --
 samplercf.txt
Listing entries in the referral credentials file...
Enter username >> test
Enter password >> test
1> localhost.stc.com | 389 | cn=Manager,dc=stc,dc=com | secret
```

Other operations, such as removing a credential entry and modifying a credential entry for an entry, can be done using the RCF utility in the same way.

The following example shows the content of a credentials file, `samplercf.txt`, with explanatory comments:

```
###This properties file was generated by
#com.stc.connector.ldapadapter.utils.RCFUtil.
#Do NOT modify this file "by hand" if you don't understand the
 nature
#or format of this file. Use the utility to create and
#manage this file.
#
#Tue Feb 14 17:49:17 PST 2006
```

```
password=P9He6eCUY6Q\=
localhost.stc.com\:389=test;P9He6eCUY6Q\=
username=test
#New credentials entry that was created.
```

Security/SSL Section Properties

The LDAP Adapter Security/SSL Section Properties are used to set the basic security features for SSL.

TABLE 24 LDAP Adapter— Security/SSL Settings

Name	Description	Required Value

JSSE Provider Class	The fully qualified name of the JSSE provider class.	The name of a valid JSSE provider class; the default is:
		`com.sun.net.ssl.internal.ssl.Provider`
		If you are running the application server on AIX, specify:
		`com.ibm.jsse.IBMJSSEProvider`

KeyStore	The default KeyStore file. The keystore is used for key/certificate management when establishing SSL connections.	A valid package location. There is no default value. It is recommended to use *c:\JavaCAPS*`\appserver\is\domains` *MyDomain* `\config\keystore.jks` where *c:\JavaCAPS* is the directory where Java CAPS is installed and *MyDomain* is the name of your domain.
KeyStore password	The default KeyStore password. The password is used to access the KeyStore used for key/certificate management when establishing SSL connections; there is no default.	A valid KeyStore password. There is no default value.
KeyStore type	The default KeyStore type. The keystore type is used for key/certificate management when establishing SSL connections. If the KeyStore type is not specified, the default KeyStore type, JKS, is used.	A valid KeyStore type.

TABLE 24 LDAP Adapter— Security/SSL Settings *(Continued)*

Name	Description	Required Value
KeyStore username	The user name for accessing the keystore used for key/certificate management when establishing SSL connections. **Note –** If the keystore type is PKCS12 or JKS, the keystore user name property is not used. PKCS12 and JKS keystore types require passwords for access but do not require user names. If you enter a value for this property, it is ignored for PKCS12 and JKS.	A valid KeyStore user name.
SSL Connection Type	The type of SSL connection to be used. Enter the desired value as follows: ■ **None:** No SSL, simple plain connection. ■ **Enable SSL:** SSL is enabled. All communication to the LDAP server uses a secure communication channel. **Note –** If you are using the Enable SSL option, the ProviderURL property must point to a secure LDAP port (the default is 636). For additional information on required values for this property, see **SSL Connection Type**.	Select None, Enable SSL, or TLS On Demand.
SSL Protocol	The SSL protocol to use when establishing an SSL connection with the LDAP server. See your JSSE documentation for information on your Application Server's platform.	Select TLS, TLSv1, SSLv3, SSLv2, or SSL.
TrustStore	Specifies the default TrustStore. The TrustStore is used for CA certificate management when establishing SSL connections.	A valid TrustStore file; there is no default.
TrustStore password	Allows you to specify the default TrustStore password. The password is for accessing the TrustStore used for CA certificate management when establishing SSL connections.	A valid TrustStore password; there is no default.
TrustStore type	Allows you to specify the TrustStore type of the TrustStore used for CA certificate management when establishing an SSL connection. If the TrustStore type is not specified, the default TrustStore type, JKS, is used.	A valid TrustStore type.

TABLE 24 LDAP Adapter— Security/SSL Settings *(Continued)*

Name	Description	Required Value
Verify hostname	Determines whether the host name verification is done on the server certificate during the SSL handshake. You can use this property to enforce strict checking of the server host name in the request URL and the host name in the received server certificate.	Select True or False. The default is False. For additional information on required values for this property, see **Verify hostname**.
X509 Algorithm Name	Specifies the X509 algorithm name to use for the trust and key manager factories.	The name of a valid X509 algorithm; the default is SunX509. If you are running the application server on AIX, specify IbmX509.

Additional Security/SSL Property Notes

Listed are the additional notes for the following Security/SSL section properties:

- "SSL Connection Type" on page 37.
- "Verify Hostname" on page 38.

SSL Connection Type

Make sure that the SSL properties, including security certificate installation, port number, and so on, are set correctly for the current LDAP server.

Transport Layer Security (TLS) is a protocol that guarantees privacy and data integrity between client/server applications communicating over the Internet. The TLS operation for this Adapter supports both secure and nonsecure communication on the same connection.

However, some LDAP servers are required to start on a configured nonsecure port and cannot start on a secure port. For details, see the appropriate documentation for the LDAP server.

- **TLS on Demand**: A feature of LDAP version 3 (**StartTLS** extended operation), which is supported in Java SDK version 1.4 and later. Selecting this option allows you to establish an SSL connection on demand programmatically.

  Note – If you are using the **TLS on Demand** option, the `ProviderURL` property must point to a nonsecure LDAP port (the default is 389).

  After selecting this option, whenever secure communication is required, you must place any method call to the LDAP server between startTLS and stopTLS calls, which can be accessed through the LDAP OTD.

In the following example, the call to `performAddEntry` goes through a secure communication channel, but the call to `performRename` goes through a nonsecure plain-communication channel:

```
startTLS();
performAddEntry();
stopTLS();

performRename();
```

Make sure that the TLS settings (in addition to the SSL settings) are configured correctly for the current LDAP server.

Note – Using the stopTLS method may cause unexpected behavior with some LDAP servers. You may need to remove the use of this method in your Collaboration Definitions. For details, see the appropriate documentation for the LDAP server.

Active Directory does not release the context, when you iteratively add a single attribute with multiple values using TLS connection. But, with the workaround of starting the TLS, adding the attribute operations and then stopping the TLS will release the context.

For information on how to use this feature with the LDAP OTD, see **TLSExtension Node**.

Verify Hostname

Under some circumstances, you can get different Java exceptions, depending on whether you set this property to True or False. This section explains what causes these exceptions.

For example, suppose the host name in the URL is `localhost`, and the host name in the server certificate is `localhost.stc.com`. Then, the following conditions apply:

- If `Verify hostname` is set to False:

  Host name checking between the requested URL and the server certificate is turned *off*.

  You can use an incomplete domain host name, for example, `https://localhost:444`, or a complete domain host name, for example, `https://localhost.stc.com:444`, and get a positive response in each case.

- If `Verify hostname` is set to True:

  Host name checking between the requested URL and the server certificate is turned *on*.

  Note – If you use an incomplete domain host name, for example, `https://localhost:444`, you can get the exception java.io.IOException: HTTPS hostname wrong.

  You must use a complete domain host name, for example, `https://localhost.stc.com:444`.

Note – If the Java SDK version used by the Application Server and the corresponding Application Server property setting do not match, you can get the exception `java.lang.ClassCastException`.

MSMQ Adapter Inbound Connectivity Map Properties

Inbound MSMQ Adapter configuration information is organized into the following topics:

MSMQ Adapter Inbound Connectivity Map Properties

When use one of these properties, for example `Format Name`, to identify a queue, recommend that you leave the value for the other two properties blank (in this case, you would leave the `MSMQ Queue Alias` and `MSMQ Queue Name` properties blank). This ensures that only the queue identification method you specify is used.

The MSMQ Configuration section of the inbound MSMQ Connectivity Map properties contains the top-level parameters displayed in the following table.

TABLE 25 Connectivity Map - Inbound - MSMQ Configuration

Name	Description	Required Value
MSMQ Queue Alias	Specifies the queue alias. A queue alias associates an ADs path and a user-defined string (alias) with a public, private, or direct single-element format name. To send or receive messages using a queue alias, MSMQ and the Active Directory service must be installed on the same computer as the Application Server.	The queue alias. See "Identifying an MSMQ Queue" on page 41 for more information.

TABLE 25 Connectivity Map - Inbound - MSMQ Configuration *(Continued)*

Name	Description	Required Value
MSMQ Format Name	Specifies the format name of the queue. The format name is a string that uniquely identifies a queue by providing some connection details and the queue's path. Different types of format names can be used to specify the way messages are routed, the type of destination, and the type of operation for which the queue is being opened. Configure your MQ security attributes to accept TCP and HTTP protocol. ■ For public queues use the following: DIRECT=TCP: *IPAddress**QueueName* DIRECT=OS: *ComputerName**QueueName* DIRECT=HTTP: *//Host/msmq/QueueName* ■ For private queues use the following: DIRECT=TCP: *IPAddress**private$*\ *QueueName* DIRECT=*OS:ComputerName*\ *private$**QueueName* Configure your MQ security attributes to accept TCP and HTTP protocol. For examples of both public and private MSMQ Format Names, see "Pre Transfer (BatchFTP Connectivity Map)" in *Oracle Java CAPS Adapter for Batch User's Guide*. **Note –** If the *ComputerName* or *Host* name contains more than 15 characters, MSMQ truncates the name. In this case, you must use the truncated ComputerName. Refer to the queue Properties, General tab to see the specific ComputerName for your system. See "Identifying an MSMQ Queue" on page 41 for more information.	The public or private format name .

TABLE 25 Connectivity Map - Inbound - MSMQ Configuration *(Continued)*

Name	Description	Required Value
MSMQ Queue Name	Specifies the Microsoft Message Queue Name.	The string name that identifies the queue. This is associated with the host name, so the value is entered as *queuename*, where *queuename* is the queue name. See "Identifying an MSMQ Queue" on page 41 for more information.
MSMQ Share Mode	Specifies the MSMQ share mode (DENY_NONE or DENY_RECEIVE_SHARE).	Select DENY_NONE or DENY_RECEIVE_SHARE. DENY_NONE is the configured default.
MSMQ Access Mode	Specifies the MSMQ Access Mode. Only RECEIVE_ACCESS is supported for inbound mode.	RECEIVE_ACCESS
MSMQ Receive Interval	Specifies the polling interval or frequency at which the adapter checks the queue for incoming messages (in milliseconds).	A number indicating the polling interval in milliseconds. The configured default is 5000 (5 seconds).
MSMQ Receive Action Code	Specifies the MSMQ Receive Action code. Only ACTION_RECEIVE is supported for inbound mode.	ACTION_RECEIVE
MSMQ Transaction Type	Specifies the Microsoft Message Transaction Type.	The string name that identifies the queue. This is associated with the host name, so the value is entered as *queuename*, where *queuename* is the queue name.

Identifying an MSMQ Queue

The MSMQ Adapter identifies an MSMQ Queue using the first available value of these three properties, in the following order:

1. MSMQ Queue Alias
2. MSMQ Format Name
3. MSMQ Queue Name

MSMQ Format Name and Host Name

The format name is a string that uniquely identifies a queue using connection details and the queue's path. Different types of format names can be used to specify how messages are routed, the type of destination, and the type of operation for which the queue is being opened.

The public or private format name property value is entered in the following manner:

For **public** queues use the following:

- DIRECT=TCP:*IPAddress**QueueName*

  for example: DIRECT=TCP:192.168.100.100\testmsmq

  - DIRECT=OS:*ComputerName**QueueName*

    for example: DIRECT=OS:mypc-gx600\testmsmq

  - DIRECT=HTTP://*Host*/msmq/*QueueName*

    for example: DIRECT=HTTP://mypc-gx600/msmq/testmsmq

    For **private** queues use the following:

  - DIRECT=TCP:*IPAddress*\private$*QueueName*for example: DIRECT=TCP:192.168.100.100\private$\privateqnxa

  - DIRECT=OS:*ComputerName*\private$*QueueName*

    for example: DIRECT=OS:mypc-gx600\private$\privateqnxa

    Configure your MQ security attributes to accept TCP and HTTP protocol. If you specify an `MSMQ Format Name`, the `MSMQ Queue Name` and `MSMQ Queue Alias` properties must be left blank.

    To find the Computer Name, right-click My Computer and click Manage. From the Computer Management dialog box, select Computer Management ⇒ Services and Applications ⇒ Message Queueing ⇒ Public Queues. Right-click your queue and select Properties. From the General tab of the Properties dialog box, you can see the computer name and queue name used by MSMQ.

Note – The Computer Name and Host Name is limited to 15 characters by MSMQ. If the Host Name contains more than 15 characters, MSMQ will truncate the name. In this case, you must use the truncated Host Name. Refer to the queue Properties, General tab to see the specific Host Name for your system.

MSMQ Adapter Outbound Connectivity Map Properties

Outbound MSMQ Adapter configuration information is organized into the following topics:

MSMQ Adapter Outbound Connectivity Map Properties

When use one of these properties, for example Format Name, to identify a queue, recommend that you leave the value for the other two properties blank (in this case, you would leave the MSMQ Queue Alias and MSMQ Queue Name properties blank). This ensures that only the queue identification method you specify is used.

The MSMQ Configuration section of the outbound MSMQ Adapter Connectivity Map properties contains the top-level properties displayed in the following table.

TABLE 26 Connectivity Map - Outbound - MSMQ Configuration

Name	Description	Required Value
MSMQ Queue Alias	Specifies the queue alias. A queue alias associates an ADs path and a user-defined string (alias) with a public, private, or direct single-element format name. To send or receive messages using a queue alias, MSMQ and the Active Directory service must be installed on the same computer as the Application Server.	The queue alias. See "SOCKS (BatchFTP Connectivity Map)" in *Oracle Java CAPS Adapter for Batch User's Guide* for more information.

TABLE 26 Connectivity Map - Outbound - MSMQ Configuration *(Continued)*

Name	Description	Required Value
MSMQ Format Name	Specifies the format name of the queue. The format name is a string that uniquely identifies a queue using connection details and the queue's path. Different types of format names can be used to specify how messages are routed, the type of destination, and the type of operation for which the queue is being opened. Configure your MQ security attributes to accept TCP and HTTP protocol. ■ For **public** queues use the following: DIRECT=TCP: *IPAddress**QueueName* DIRECT=OS: *ComputerName**QueueName* DIRECT=HTTP: *//Host/msmq/QueueName* ■ For **private** queues use the following: DIRECT=TCP: *IPAddress*\ *private$**QueueName* DIRECT=OS: *ComputerName**private$*\ *QueueName* Configure your MQ security attributes to accept TCP and HTTP protocol. For examples of both public and private MSMQ Format Names, see "Pre Transfer (BatchFTP Connectivity Map)" in *Oracle Java CAPS Adapter for Batch User's Guide*. If the *ComputerName* or *Host* name contains more than 15 characters, MSMQ will truncate the name. In this case, you must use the truncated ComputerName. refer to the queue Properties, General tab to see the specific ComputerName for your system. See "SOCKS (BatchFTP Connectivity Map)" in *Oracle Java CAPS Adapter for Batch User's Guide* for more information.	The public or private format name.

TABLE 26 Connectivity Map - Outbound - MSMQ Configuration *(Continued)*

Name	Description	Required Value
MSMQ Queue Name	Specifies the Microsoft Message queue name. See "SOCKS (BatchFTP Connectivity Map)" in *Oracle Java CAPS Adapter for Batch User's Guide* for more information.	The string name that identifies the queue. This is associated with the host name, so the value is entered as `host/queue`, where *host* is the host name and *queue* is the queue name.
MSMQ Transaction Type	Specifies the outbound transaction type. The options are: ■ **MQ_NO_TRANSACTION**: Select this value when sending messages to a non-transactional queue. ■ **MQ_XA_TRANSACTION**: Select this value when sending messages to an XA-transactional queue. ■ **MQ_SINGLE_MESSAGE**: Select this value when sending messages to a transactional queue. This value does not apply to or affect inbound messages from a transactional queue.	Select MQ_NO_TRANSACTION, MQ_XA_TRANSACTION, or MQ_SINGLE_MESSAGE as the transaction type. Use the default value, MQ_NO_TRANSACTION. For more information, see the MSMQ user documentation.
MSMQ Share Mode	Specifies the MSMQ share mode as one of the following: ■ **DENY_NONE**: Grants everyone full access to send, peek at, or retrieve messages from the queue while it is open. ■ **DENY_RECEIVE_SHARE**: You alone can peek at or retrieve messages until the queue is closed. Other applications are still able to send messages to the queue, but they may not delete messages while you have the queue open.	Select DENY_NONE or DENY_RECEIVE_SHARE. DENY_NONE is the configured default. An error message occurs if you or anyone tries to open the queue with RECEIVE_ACCESS or PEEK_ACCESS after Message Queuing opens the queue. If you attempt to open a queue with DENY_RECEIVE_SHARE when the queue is already open with RECEIVE_ACCESS or PEEK_ACCESS, the call will fail.
Message Priority	Specifies the message priority level for all messages. Priorities range from level 0 (low priority) to 7 (high priority). Messages are stored in descending order in the queue, with higher priority messages first. Priority is only valid when `MSMQ Access Mode` is set to SEND_ACCESS .	A number between 0 and 7 indicating the message priority. The configured default is 3.

TABLE 26 Connectivity Map - Outbound - MSMQ Configuration *(Continued)*

Name	Description	Required Value
MSMQ Receive Action Code	Specifies the MSMQ receive action code as one of the following: ■ **ACTION_RECEIVE**: Reads the message at the current cursor location and removes it from the queue. ■ **ACTION_PEEK_CURRENT**: "Peeks" for last message in the queue. You can use this functionality when there are several messages coming in and going out of queue, and you want to peek (look) and get the latest message.	Select ACTION_RECEIVE or ACTION_PEEK_CURRENT. ACTION_RECEIVE is the configured default.
MSMQ Access Mode	Specifies whether Message Queuing opens the queue with peek, send, or receive access. ■ **RECEIVE_ACCESS**: Allows messages to be retrieved from a queue when they are read. This is used when the receiving application opens a queue to remove messages. ■ **SEND_ACCESS**: Allows messages to be sent to a queue. This is used when the sending application opens a queue to send messages. ■ **PEEK_ACCESS**: Allows messages to be read from a queue without removing them. This is used when the receiving application opens a queue to read message.	Select RECEIVE_ACCESS, SEND_ACCESS, or PEEK_ACCESS. You must set Access Mode to SEND ACCESS to use Message Priority.
Connection Mode	Specifies whether a physical connection is established when an external connection is instantiated. The options are: ■ **Automatic:** Establishes a physical connection when an external connection is instantiated. ■ **Manual:** Does not connect to the external system on startup, and instead expects the user to initiate the connection from the Collaboration (for example, by calling the `connect()` method).	Select Automatic or Manual (dynamic). The configured default is Automatic.

Identifying an MSMQ Queue

The MSMQ Adapter identifies an MSMQ Queue using the first available value of these three properties, in the following order:

1. MSMQ Queue Alias

2. MSMQ Format Name
3. MSMQ Queue Name

TCP/IP Adapter Inbound Connectivity Map Properties

The inbound property settings determine the adapter's behavior for input operations. The TCP/IP inbound adapter configuration parameters, accessed from the Connectivity Map, are organized into the following sections:

General Inbound Settings — TCP/IP Adapter Inbound

The General Inbound Settings properties provide the dedicated session mode and maximum data size message settings for the server. The following table lists and describes the top-level parameters.

TABLE 27 Connectivity Map - General Inbound Settings

Name	Description	Required Value
Max Data Size	Allows you to define the maximum size of the data that the programs can hold internally.	The valid range is from 1 to 2 GB (the maximum value of the Java integer). The configured default is 2147483647.

TABLE 27 Connectivity Map - General Inbound Settings *(Continued)*

Name	Description	Required Value
Scope Of State	Defines the scope of State object, which is an OTD node. The options for this parameter are: ■ **Resource Adapter Level**: The State has the same life cycle as the resource adapter. ■ **Persistence**: The State is persisted in the storage media like file or DB (**Persistence State File Location** must be specified if this option is selected). ■ **Connection Level**: The State has the same life cycle as the connection. ■ **OTD Level**: The State has the same life cycle as the OTD object. This scope represents the life cycle of the State.	Select one of the following: ■ Resource Adapter Level ■ Connection Level ■ OTD Level The configured default is Resource Adapter Level.
Dedicated Session Mode	Allows you to enable or disable the adapter's Dedicated Session Mode. When the Dedicated Session Mode is enabled in a server, the current client's request can exclusively hold the server port to which it connects. For example, if this property is enabled, and the client is connected to a server, it only serves the client until the work is completed, and the session is disconnected. If another client tries to connect to the server during this time, it cannot until the session is completed.	Select True or False. True indicates that Dedicated Session Mode is enabled. The configured default is False.

TCPIP Inbound Settings — TCP/IP Adapter Inbound

The **TCPIP Inbound Settings** properties provide the basic TCP/IP values for the server. The TCP/IP Inbound Settings properties contain the top-level parameters as displayed in the table.

TABLE 28 Connectivity Map - TCPIP Inbound Settings

Name	Description	Required Value
Connection Type	Specifies how the adapter establishes the TCP/IP connection: ■ **Client**: The adapter connects to an external server (host/port) to establish the connection. The adapter is in active mode. ■ **Server**: The adapter waits/listens on a certain port for an incoming connection request from an external client. Once the request is received, the adapter accepts the request and establishes the connection. The adapter is in passive mode.	Select Client or Server. Server is the default setting. Unless you specifically require Client mode, leave this value as the default: Server.
ServerSO Timeout	Allows you to set or get the server SO_TIMEOUT value, in milliseconds.	The server's SO_TIMEOUT value is in milliseconds. The default value is 10000 milliseconds (10 seconds).
Server Socket Factory Implementation Class Name	Enter the name of the Java class that implements the server socket factory. This class is used to create the server socket. If you have provided your own server socket implementation, enter the name of the Java class that contains this implementation. The factory implementation class must implement the following interface: `com.stc.connector.tcpip.model.factory.TCPIPSocketFactory`	A valid Java class name; the default is: `com.stc.connector.tcpip.model.factory.TCPIPSocketFactoryImpl`
Keep Alive	Specifies whether the server's SO_KEEPALIVE option is enabled or disabled. It is used for the accepted client socket. **Note –** For some properties, the server socket itself does not have direct properties settings associated with it. Instead, the properties map have direct properties settings associated to the accepted client socket.	Select True or False. True indicates that the server SO_KEEPALIVE option is enabled. The configured default is True.
Receive Buffer Size	Allows you to set or get the value of the server's SO_RCVBUF option for the current socket, that is, the buffer size used by the operating system for input on this socket. It is used for the accepted client socket.	A number indicating the receive buffer size. The configured default is 8192.
Send Buffer Size	Allows you to set or get the value of the server's SO_SNDBUF option for the current socket, that is, the buffer size used by the operating system for output on this socket. It is used for the accepted client socket.	A number indicating the send buffer size. The configured default is 8192.

TABLE 28 Connectivity Map - TCPIP Inbound Settings *(Continued)*

Name	Description	Required Value
SoLinger	Specifies whether the server's SO_LINGER option is enabled or disabled; used for the accepted client socket.	Select True or False. True enables the SO_LINGER option. The configured default value is True.
SoLinger Timeout	Specifies the server's linger timeout in seconds. The maximum timeout value is platform specific. The setting only affects the socket close; used for the accepted client socket.	The linger timeout in seconds. The configured default is 30 seconds, indicating that the SO_LINGER option is disabled.
SoTimeout	Allows you to set or get the value of the server's SO_TIMEOUT value, in milliseconds. Used for the accepted client socket. A timeout of 0 (zero) is an infinite timeout. If you specify this value, the adapter goes into an infinite read. If this action happens, it is recorded in the adapter's log file.	The SO_TIMEOUT value in milliseconds. The configured default value is 10000 milliseconds (10 seconds).
TcpNoDelay	Specifies whether the server's TCP_NODELAY option (that is, Nagle's algorithm) is enabled or disabled. Used for the accepted client socket.	Select True or False. True enables the TCP_NODELAY option. The configured default value is False.

TCPIP Inbound Settings - Server Port Binding — TCP/IP Adapter Inbound

The **Server Port Binding** section defines the configuration parameters used for controlling the server port binding. The following table lists and describes the TCP/IP Inbound Settings — Server Port Binding properties.

TABLE 29 Connectivity Map - TCPIP Inbound Settings - Server Port Binding

Name	Description	Required Value
Max Binding Retry	Specifies the maximum number of times the adapter attempts to bind to the specified TCP/IP port on the localhost.	An integer indicating the number of bind attempts. The configured default is 3.

TABLE 29 Connectivity Map - TCPIP Inbound Settings - Server Port Binding *(Continued)*

Name	Description	Required Value
Retry Binding Interval	Specifies the amount of time (in milliseconds) the adapter waits between attempts to bind to the specified TCP/IP port on the localhost.	An integer indicating the amount of time in milliseconds that the adapter waits between attempts. The configured default is 30000 (30 seconds).

TCPIP Inbound Settings - Client Connection Establishment — TCP/IP Adapter Inbound

The Client Connection Establishment properties define some of the configuration parameters used for controlling the connection establishment. This section is used only when the Connection Type is set as Client.

The following table lists and describes the TCP/IP inbound adapter Connectivity Map properties.

TABLE 30 Connectivity Map - TCPIP Inbound Settings - Client Connection Establishment

Name	Description	Required Value
Time to Wait Before Attempting Connection	Specifies the amount of time (in milliseconds) the adapter waits before attempting to connect to the external system.	A number indicating the amount of time (in milliseconds) the adapter waits before attempting to connect to an external system. The configured default is 30000 (30 seconds).

TCPIP Inbound Settings - Inbound Connection Management — TCP/IP Adapter Inbound

The Inbound Connection Management properties define the parameters used for inbound Server Connection Management. For example, the connection pool and the life cycle of the accepted connection.

The following table lists and describes the TCP/IP HL7 inbound adapter Connectivity Map properties.

TABLE 31 Connectivity Map - TCPIP Inbound Settings - Inbound Connection Management

Name	Description	Required Value
Max Connection Pool Size	Specifies the maximum number of concurrent connections allowed for the specific listener/monitor which is listening or monitoring a specified TCP/IP port. This represents the capability or availability of this server's services. Each connect-request from a client gains one concurrent connection. This parameter also represents the maximum number of clients who can concurrently connect to this server's services, and get served by the specific listener/monitor at the same time.	A number indicating the maximum number of concurrent connections available from a listener/monitor for a specific TCP/IP port. 0 indicates that there is no limit. The configured default is 50.
Scope Of Connection	Specifies the scope of the accepted connection which is used by the adapter. The two options are: ■ **Resource Adapter Level**: The resource adapter will close the connection upon receiving a closure request, so the connection may **keep alive** during multiple executions of the Collaboration. ■ **Collaboration Level**: The connection is closed once the Collaboration has been executed, so the connection has the same life cycle as the Collaboration.	Select Resource Adapter Level or Collaboration Level. The configured default value is Resource Adapter Level.
Close Notification	Specifies the close notification value. When the server receives a notification with content that matches this parameter's value, the server safely closes the connection and cancels any corresponding schedules.	A String indicating the trigger value that notifies the server to close the connection. The configured default is QUIT.
Idle Timeout	Specifies the length of time (in milliseconds) for inactivity of the requestor (client). The adapter attempts to detect activity on client side (the other side of the connection). If no client activity (no i/o request comes over the connection from the client) for a specified time period, then the connection is closed from the server side to release the resource. The value is in milliseconds.	An integer that indicates the amount of time (in milliseconds) for inactivity of the requestor (client) before the connection is closed from the server side to release the resource. A value of **0** disables IdleTimeout. The configured default is 60000 (1 minute).

TCPIP Inbound Schedules - Listener Schedule — TCP/IP Adapter Inbound

This section configures the scheduler used by the inbound TCP/IP Server. The server waits for a new client connection establishment request. These parameters are used to configure the listener/monitor that listens on the specified port.

Two J2EE schedulers are available (see Scheduler):

- **Timer Service**: This scheduler is configured using the At Fixed Rate, Delay, and Period properties.
- **Work Manager**: Available for J2EE (JCA 1.5 and above). This scheduler is configured using the Delay and Period properties.

Both schedulers provide the functionality required by the inbound TCP/IP Server.

The following table lists and describes the TCP/IP HL7 inbound adapter Connectivity Map properties.

TABLE 32 Connectivity Map - TCPIP Inbound Schedules - Listener Schedule

Name	Description	Required Value
Scheduler	Specifies the scheduler type for this inbound communication. There are two options: ■ **Timer Service**: The scheduler is configured using the At Fixed Rate, Delay and Period properties. ■ **Work Manager**: The task is scheduled through the J2EE Work Manager. Work Manager is supported by J2EE (JCA 1.5 and above). This scheduler is configured using the Delay and Period properties	Select Timer Service or Work Manager. If your container doesn't support JCA Work Manager, select Timer Service.
Schedule Type	This property configuration, though visible from the Properties Editor, is disabled. The only available schedule type is Repeated, indicating that the task is scheduled for repeated execution at regular intervals defined by the Period property in this section (see Period).	This property is disabled.
Delay	Applies to both the Timer Service or the Work Manager. Specifies, in milliseconds, the length of delay time before the task is executed.	An integer indicating the amount of time before the task is executed, in milliseconds (1000 milliseconds is equal to 1 second).

TABLE 32 Connectivity Map - TCPIP Inbound Schedules - Listener Schedule *(Continued)*

Name	Description	Required Value
Period	Specifies the regular interval, in milliseconds, between successive repeated task executions. This is used for the Repeated Schedule Type. See **Schedule Type**. Applies to both the Timer Service or the Work Manager.	An integer indicating the amount of time between successive task executions, in milliseconds. Enter a positive integer. The configured default is 100. Lowering this value may increase the number of transactions per second.
At Fixed Rate	Specific to the Timer Service configuration only. Specifies whether a Fixed-Rate execution or Fixed-Delay execution is used. ■ **Fixed-Rate**: A fixed-rate execution means that each execution is scheduled relative to the scheduled time of the initial execution. If an execution is delayed for any reason (such as garbage collection or other background activity), two or more executions will occur in rapid succession to catch up. In the long run, the frequency of execution will be exactly the reciprocal of the specified period (assuming the system clock underlying Object.wait(long) is accurate). ■ **Fixed-Delay**: A fixed-delay execution means that each execution is scheduled relative to the actual time of the previous execution. If an execution is delayed for any reason (such as garbage collection or other background activity), subsequent executions will be delayed as well. As a result, the frequency of execution will generally be slightly lower than the reciprocal of the specified period, assuming the system clock underlying Object.wait(long) is accurate.	Select True or False. True indicates that a fixed-rate execution is used. False indicates that a fixed-delay execution is used.

TCPIP Inbound Settings - Service Schedule — TCP/IP Adapter Inbound

This section configures the scheduler used by the TCP/IP Server that executes the business tasks (Collaboration Rules) over the existing connection. This scheduler affects the actual Business Rules defined by the user.

Two J2EE schedulers are available (see **Scheduler**):

- **Timer Service**: This scheduler is configured using the At Fixed Rate, Delay, Period, and Schedule Type properties.
- **Work Manager**: Available for J2EE (JCA 1.5 and above). This scheduler is configured using the Delay, Period, and Schedule Type properties.

Both schedulers provide the functionality required by the inbound TCP/IP Server.

The following table lists and describes the TCP/IP inbound adapter Connectivity Map properties.

TABLE 33 Connectivity Map - TCPIP Inbound Settings - Server Schedule

Name	Description	Required Value
Scheduler	Specifies the scheduler type for this inbound communication. There are two options: ■ **Timer Service**: This scheduler is configured using the At Fixed Rate, Delay, and Period properties. ■ **Work Manager**: The task is scheduled through the J2EE Work Manager. Work Manager is supported by J2EE (JCA 1.5 and above). This scheduler is configured using the Delay and Period properties.	Select Timer Service or Work Manager. If your container doesn't support JCA Work Manager, select Timer Service .
Schedule Type	Applies to both the Timer Service or the Work Manager. Specifies whether the task is scheduled to occur once or be repeated. ■ **OneTime**: The task will be scheduled for one-time execution. ■ **Repeated**: The task will be scheduled for repeated execution at regular intervals defined by the Period property in this section (see **Period**).	Select OneTime or Repeated.
Delay	Applies to both the Timer Service or the Work Manager. Specifies, in milliseconds, the length of delay time before the task is executed.	An integer indicating the amount of time, in milliseconds, before the task is executed (1000 milliseconds is equal to 1 second).
Period	Specifies the wait interval in milliseconds between successive repeated task executions. This is used for the Repeated Schedule Type (see **Schedule Type**). Applies to both the Timer Service or the Work Manager.	An integer indicating the amount of time, in milliseconds, between successive task executions (1000 milliseconds is equal to 1 second). Enter a positive integer. The configured default is 100. Lowering this value may increase the number of transactions per second.

TABLE 33 Connectivity Map - TCPIP Inbound Settings - Server Schedule *(Continued)*

Name	Description	Required Value
At Fixed Rate	Specific to the Timer Service configuration only. Specifies whether a Fixed-Rate execution or Fixed-Delay execution is used. This is used for the "Repeated" schedule type by the "Timer Service" scheduler. ■ **Fixed-Rate**: A fixed-rate execution means that each execution is scheduled relative to the scheduled time of the initial execution. If an execution is delayed for any reason (such as garbage collection or other background activity), two or more executions will occur in rapid succession to catch up. In the long run, the frequency of execution will be exactly the reciprocal of the specified period (assuming the system clock underlying Object.wait(long) is accurate). ■ **Fixed-Delay**: A fixed-delay execution means that each execution is scheduled relative to the actual time of the previous execution. If an execution is delayed for any reason (such as garbage collection or other background activity), subsequent executions will be delayed as well. As a result, the frequency of execution will generally be slightly lower than the reciprocal of the specified period (assuming the system clock underlying Object.wait(long) is accurate).	Select True or False. True indicates that a fixed-rate execution is used. False indicates that a fixed-delay execution is used.

TCPIP Inbound Settings - Envelope Message — TCP/IP Adapter Inbound

The following table lists and describes the Inbound TCP/IP adapter Connectivity Map properties.

TABLE 34 Connectivity Map - TCPIP Inbound Settings - Envelope Message

Name	Description	Required Value
Envelope Type	Specifies the envelope type. The envelope type defines where a message starts and stops.	Enter one of the following properties denoting the envelope type: ■ BeginEndMarked ■ EndMarked ■ FixedLength ■ LengthPrefixed ■ MarkedAndFixed ■ PerActiveConnection ■ Custom The default is BeginEndMarked.
	BeginEndMarked is supported by the properties `Bytes to Read`, `Ignore Until Char Value`, and `Store Until Char Value`.	
	EndMarked is supported by the property `Store Until Char Value`.	
	FixedLength is supported by the properties `Bytes to Read`.	
	LengthPrefixed is supported by the properties `Width of Length` and `Numeric Representation`.	
	MarkedAndFixed is supported by the properties `Bytes to Read`, `Ignore Until Char Value`, and `Store Until Value`.	
	PerActiveConnection is supported by the property `PerActiveConnection`.	
	Custom is supported by the properties `Custom Enveloped Class Name` and `Custom Defined Property`. **Note –** For all envelope types, except MarkedAndFixed, the data is just the payload. See MarkedAndFixed for an explanation of how the data is handled by that envelope type.	

TABLE 34 Connectivity Map - TCPIP Inbound Settings - Envelope Message *(Continued)*

Name	Description	Required Value

Custom Enveloped Class Name	Specifies the Java class name to be used when the `Envelope Type` property is set to Custom.
	
	If you are using a custom envelope you have created, using a Java Class, you can import the Java JAR file containing the class into any desired Collaboration, using the Collaboration Editor's file import feature.
	
	The class name should be a full qualified class name, such as `com.abc.MyClass`. The class must implement interfaces
	
	`com.stc.connector.tcpip.ext.msg.EnvelopedMsgReceiver`
	
	and
	
	`com.stc.connector.tcpip.ext.msg.EnvelopedMsgSender`
	
	For more details, see **Customized Enveloping**.
		A full Java class name.
		
		A full qualified class name, or None if Custom is not the Envelope Type.
		
		The configured default is None.

Customer Defined Property	Used when the Envelope Type value is set to Custom. Specifies a list of user-defined parameters. You can parse this information, such as delimiters, into your customized envelope message implementation.	A text string.
Bytes to Read	Used with the following Envelope Types: ■ **FixedLength** ■ **MarkedAndFixed** Specifies the number of bytes to read. It is assumed that all Events received by the adapter have the same length.	An integer indicating the number of bytes. The configured default is 1.
Width of Length	Used for Envelope Type value LengthPrefixed. Specifies the width of the envelope length. In other words, it dictates the number of digits to be used to represent the length field.	An integer, the range is 1 to 10. This property must be set to 2 for Network short and 4 for Network long. The configured default value is 1.

TABLE 34 Connectivity Map - TCPIP Inbound Settings - Envelope Message *(Continued)*

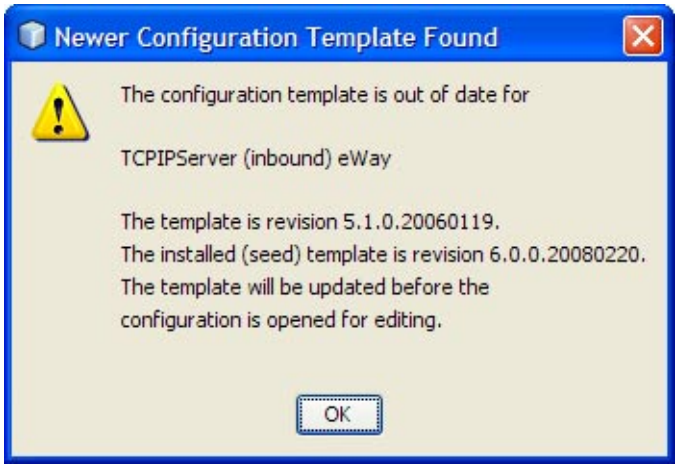
Name	Description	Required Value
Numeric Representation	Used for Envelope Type value LengthPrefixed. Specifies how the number representation of the prefixed length is expressed. This value is expressed in one of the following formats: ■ Decimal ■ Hexadecimal ■ Octal ■ Network Short ■ Network Long	Select one of the following: ■ Decimal ■ Hexadecimal ■ Octal ■ Network Short ■ Network Long The configured default is Decimal.
Ignore Until Char Value	Used for the Envelope Types BeginEndMarked and MarkedAndFixed. Specifies the value for the ignore-until (same as begin block) character. All incoming characters are ignored until this character is encountered.	A decimal ASCII number. The allowed range is 1 to 127. The configured default is 11.
Store Until Char Value	Used for Envelope Types BeginEndMarked, EndMarked, and MarkedAndFixed. Specifies the character in the End Block or Marker position of the envelope. All incoming characters are stored until this character is encountered.	A decimal ASCII number. The allowed range is 1 to 127. The configured default is 12.

Java CAPS 5.1.x to 6 Upgrade Procedure

There are new versions of the Configuration templates used in version 6. For previous 5.1.x projects that are imported or going through an "in-place upgrade" to the latest version, the Configuration template will be upgraded during design time or build time.

At design time when you open the Connectivity Map or Environment properties window, a warning window appears (as shown in the figure below), and the Configuration template automatically upgrades. You can now update the Environment properties with any necessary change, and run the project.

FIGURE 3 Configuration Template Warning Window



If you attempt to build a project without first opening either the Connectivity Map or Environment property window, code generation will automatically upgrade the Configuration template. Once this build-time upgrade scenario is complete, you will not see the warning window anymore.

TCP/IP Adapter Outbound Connectivity Map Properties

The outbound TCP/IP adapter properties determine the adapters behavior for output operations. The outbound TCP/IP Adapter Connectivity Map properties are organized into the following sections:

General Outbound Settings — TCP/IP Adapter Outbound

The General Outbound Settings properties provides a general TCP/IP outbound configuration information. The following table lists and describes the General Outbound Setting properties.

TABLE 35 Connectivity Map - General Outbound Settings

Name	Description	Required Value
Max Data Size	Specifies the maximum amount of data that the programs can hold internally.	The valid range is from 1 to 2147483647 bytes (2 GB — the maximum value of the Java integer). The configured default is 2147483647.
Scope Of State	Specifies the scope of State object, which is an OTD node. The options for this parameter are: ■ **Resource Adapter Level**: The State has the same life cycle as the resource adapter. ■ **Persistence**: The State is persisted in the storage media like file or DB (**Persistence State File Location** must be specified if this option is selected). ■ **Connection Level**: The State has the same life cycle as the connection. ■ **OTD Level**: The State has the same life cycle as the OTD object. This scope represents the life cycle of the State.	Select one of the following: ■ Resource Adapter Level ■ Connection Level ■ OTD Level The configured default is Resource Adapter Level.

TCPIP Outbound Settings — TCP/IP Adapter Outbound

The TCPIP Outbound Settings properties presents the Java Socket options. For more information, see the JDK Javadoc. The following table lists and describes the TCP/IP Outbound Settings properties.

Note – For complete information on options referred to by these base settings, such as SO_KEEPALIVE, see the appropriate Oracle Java documentation.

TABLE 36 Connectivity Map - TCPIP Outbound Settings

Name	Description	Required Value
Connection Type	Specifies how the adapter establishes the TCP/IP connection: ■ **Client**: The adapter connects to an external server (host/port) to establish the connection. The adapter is in active mode. ■ **Server**: The adapter waits/listens on a particular port for an incoming connection request from an external client. Once the request is received, the adapter accepts the request and establishes the connection. The adapter is in passive mode.	Select Client or Server. Server is the default setting. Unless you specifically require Server mode, leave this value as the default: Client.
ServerSO Timeout	Sets or gets the value of the SoTimeout for the ServerSocket, in milliseconds. Used for `ServerSocket.accept()`. When you set this option to a non-zero timeout, calling `accept()` for ServerSocket will block for only this period of time. If the timeout expires, a `java.net.SocketTimeoutException` (or `java.net.InterruptedIOException`) is thrown, though the ServerSocket remains valid. Enable this option prior to entering the blocking operation. This parameter is used only when the `Connection Type` is set as Server.	An integer that indicates the SoTimeout value in milliseconds. The default value is 60000 milliseconds (60 seconds). The timeout must be greater than 0 (zero). A timeout value of 0 is interpreted as an infinite timeout.
Keep Alive	Specifies whether the client's SO_KEEPALIVE option is enabled or disabled. True indicates that the server SO_KEEPALIVE option is enabled.	Select True or False. The configured default is True.
Receive Buffer Size	Allows you to set or get the value of the client SO_RCVBUF option for the current socket, that is, the buffer size used by the operating system for input on this socket. It sets a hint as to the size of the underlying buffers used by the platform for incoming network I/O. When used in **set**, this is a suggestion to the kernel, from the application, regarding the buffer sizes to use for the data that will be received over the socket. When used in **get**, this must return the size of the buffer actually used by the platform when receiving in data on this socket.	An integer indicating the receive buffer size. The configured default is 8192.

TABLE 36 Connectivity Map - TCPIP Outbound Settings *(Continued)*

Name	Description	Required Value
Send Buffer Size	Allows you to set or get the value of the client's SO_SNDBUF option for the current socket, that is, the buffer size used by the operating system for output on this socket.	A number indicating the send buffer size. The configured default is 8192.
SoLinger	Specifies whether the client's SO_LINGER option is enabled or disabled.	Select True or False. True enables the SO_LINGER option.
SoLinger Timeout	Specifies the client's linger timeout in seconds. The maximum timeout value is platform specific. The setting only affects the socket close.	The linger timeout in seconds. The configured default is 30 seconds, indicating that the SO_LINGER option is disabled.
SoTimeout	Allows you to set or get the value of the client's SO_TIMEOUT value, in milliseconds.	The SO_TIMEOUT value in milliseconds. The configured default value is 10000 milliseconds (10 seconds).
TcpNoDelay	Specifies whether the client's TCP_NODELAY option (that is, Nagle's algorithm) is enabled or disabled.	Select True or False. True enables the TCP_NODELAY option.
Socket Factory Implementation Class Name	Enter the name of the Java class that implements the client socket factory. This class is used to create the client socket. If you have provided your own client socket implementation, enter the name of the Java class that contains this implementation. The factory implementation class must implement the following interface: `com.stc.connector.tcpip.model.factory.TCPIPSocketFactory`	A valid Java class name; the default is: `com.stc.connector.tcpip.model.factory.TCPIPSocketFactoryImpl`

TCPIP Outbound Settings - Connection Establishment — TCP/IP Adapter Outbound

The Client Connection Establishment properties define some of the configuration parameters used to control the connection establishment.

The following table lists and describes the TCP/IP Outbound Adapter Connectivity Map properties.

Note – This section is used only when the Connection Type is set as Client.

TABLE 37 Connectivity Map - TCPIP Outbound Settings - Client Connection Establishment

Name	Description	Required Value
Time to Wait Before Attempting Connection	Specifies the length of time (in milliseconds) the adapter waits before attempting to connect to the external system.	A number indicating the amount of time (in milliseconds) the adapter waits before attempting to connect. The configured default is 0.
Always Create New Connection	Specifies whether the adapter always attempts to create a new connection when a connection establishment request is received. ■ **True** indicates that the adapter always attempts to create a new connection without attempting to match an existing connection. ■ **False** indicates that the adapter attempts to match an existing connection (managed by the container).	Select True or False. The configured default is False.
Auto Reconnect Upon Matching Failure	Specifies whether to attempt to re-connect automatically when the adapter gets a matching connection from a container, even though this connection is not valid due to various reasons: for example, the external side of the connection is closed/reset due to the external application's logic. This property only takes effect when the application server has an existing connection in its connection pool, not during an initial triggering when the pool is empty. ■ **True** indicates that the adapter discards the invalid matching connection and automatically attempts to reconnect using a new connection. ■ **False** indicates that the adapter does not automatically attempt to reconnect using a new connection: instead, a exception is thrown and the adapter raises the appropriate alert. The user must detect this type of failure and act appropriately.	Select True or False. The configured default is True.
Max Connection Retry	Specifies the maximum number of times the adapter attempts to connect to a specific external TCP/IP destination (host/port) before giving up.	An integer indicating the number of times the adapter attempts to connect.

TABLE 37 Connectivity Map - TCPIP Outbound Settings - Client Connection Establishment *(Continued)*

Name	Description	Required Value
Retry Connection Interval	Specifies the amount of time (in milliseconds) the adapter waits between attempts to connect to a specific external TCP/IP destination (host/port).	An integer indicating the amount of time (in milliseconds) the adapter waits between attempts to connect. The configured default is 30000 (or 30 seconds).

TCPIP Outbound Settings - Server Port Binding

The Server Port Binding section defines the configuration parameters used for controlling the server port binding. This parameter is used only when the `Connection Type` is set as Server. The following table lists and describes TCP/IP Outbound Settings — Server Port Binding properties.

TABLE 38 Connectivity Map - TCPIP Outbound Settings - Server Port Binding

Name	Description	Required Value
Max Binding Retry	Specifies the maximum number of times the adapter attempts to bind to the specified TCP/IP port on the localhost.	An integer indicating the number of bind attempts to the specified TCP/IP port on the localhost.
Retry Binding Interval	Specifies the amount of time (in milliseconds) the adapter waits between attempts to bind to the specified TCP/IP port on the localhost.	An integer indicating the amount of time in milliseconds that the adapter attempts to bind to the specified TCP/IP port. The configured default is 30000 (30 seconds).

TCPIP Outbound Settings - Envelope Message

These properties are the envelope message format settings for the Outbound adapter. These properties operate in the same way as those for the inbound adapter

This section explains the envelope message format properties for the server. These properties are all associated with TCP/IP enveloping. This section of the Outbound TCP/IP adapter Connectivity Map properties contains the top-level parameters as displayed in the table.

TABLE 39 Connectivity Map - TCPIP Outbound Settings - Envelope Message

Name	Description	Required Value
Envelope Type	Specifies the envelope type. The envelope type defines where a message starts and stops.	Enter one of the following properties denoting the envelope type: ■ BeginEndMarked ■ EndMarked ■ FixedLength ■ LengthPrefixed ■ MarkedAndFixed ■ PerActiveConnection ■ Custom The default is BeginEndMarked.
	BeginEndMarked is supported by the properties `Bytes to Read`, `Ignore Until Char Value`, and `Store Until Char Value`.	
	EndMarked is supported by the property `Store Until Char Value`.	
	FixedLength is supported by the property `Bytes to Read`.	
	LengthPrefixed is supported by the properties `Width of Length` and `Numeric Representation`.	
	MarkedAndFixed is supported by the property `Bytes to Read`, `Ignore Until Char Value`, and `Store Until Value`.	
	PerActiveConnection is supported by the property **PerActiveConnection**.	
	Custom is supported by the properties `Custom Enveloped Class Name` and `Custom Defined Property`. For optimum performance, use the method `receiveEnvelopedMsg()` with any enveloped messages. This method uses the envelope as its ending condition, while the other receiving methods, `receiveBytes()` and `receiveString()`, use a timeout as their ending condition. **Note –** For all envelope types, except MarkedAndFixed, the data is just the payload. See MarkedAndFixed for an explanation of how the data is handled by that envelope type.	

TABLE 39 Connectivity Map - TCPIP Outbound Settings - Envelope Message *(Continued)*

Name	Description	Required Value

Custom Enveloped Class Name	Specifies the Java class name to be used when the `Envelope Type` property is set to Custom.
	
	If you are using a custom envelope you have created, using a Java Class, you can import the Java JAR file containing the class into any desired Collaboration, using the Collaboration Editor's file import feature.
	
	The class name should be a full qualified class name, such as `com.abc.MyClass`. The class must implement interfaces
	
	`com.stc.connector.tcpip.ext.msg.EnvelopedMsgReceiver`
	
	and
	
	`com.stc.connector.tcpip.ext.msg.EnvelopedMsgSender`
	
	For more details, see **Customized Enveloping**.
		A full Java class name.
		
		A full qualified class name, or None if Custom is not the Envelope Type.
		
		The configured default is None.

Customer Defined Property	Used when the Envelope Type value is set to **Custom**. Specifies a list of user-defined parameters. You can parse this information, such as delimiters, into your customized envelope message implementation.	A text string.
Bytes to Read	Used with the following Envelope Types: ■ **FixedLength** ■ **MarkedAndFixed** Specifies the number of bytes to read. It is assumed that all Events received by the adapter have the same length.	An integer indicating the number of bytes. The configured default is 1.
Width of Length	Used for Envelope Type value LengthPrefixed. Specifies the width of the envelope length. In other words, it dictates the number of digits to be used to represent the length field.	An integer, the range is 1 to 10. This property must be set to 2 for Network short and 4 for Network long. The configured default value is 1.

TABLE 39 Connectivity Map - TCPIP Outbound Settings - Envelope Message *(Continued)*

Name	Description	Required Value
Numeric Representation	Used for Envelope Type value LengthPrefixed. Specifies how the number representation of the prefixed length is expressed. This value is expressed in one of the following formats: ■ Decimal ■ Hexadecimal ■ Octal ■ Network Short ■ Network Long	Select one of the following: ■ Decimal ■ Hexadecimal ■ Octal ■ Network Short ■ Network Long The configured default is Decimal.
Ignore Until Char Value	Used for the Envelope Types BeginEndMarked and MarkedAndFixed. Specifies the value for the ignore-until (same as begin block) character. All incoming characters are ignored until this character is encountered.	A decimal ASCII number. The allowed range is 1 to 127. The configured default is 11.
Store Until Char Value	Used for Envelope Types BeginEndMarked, EndMarked, and MarkedAndFixed. Specifies the character in the End Block or Marker position of the envelope. All incoming characters are stored until this character is encountered.	A decimal ASCII number. The allowed range is 1 to 127. The configured default is 12.