Configuring Java CAPS Environment Components for Communications Adapters

Sun microsystems

Contents

Configuring Java CAPS Environment Components for Communications Adapters

The adapter environment configuration properties contain parameters that define how the adapter connects to and interacts with other Java CAPS components within the environment. The environment properties are accessed from the NetBeans IDE Services window. The following sections provide instructions on how to configure Java CAPS component environment properties and lists the environment properties for the various communications adapters.

What You Need to Know

This topic provides information you should know to start configuring the environment properties "Using the Environment Properties Editor" on page 8.

What You Need to Do

These topics provide configuration information used to set the communications adapter environment properties.

- "Configuring IMS Adapter Environment Properties" on page 72.
- "Configuring LDAP Adapter Properties" on page 83.
- "Configuring the SNA Adapter Environment Properties" on page 87.

Using the Environment Properties Editor

The Adapter Environment Configuration properties contain parameters that define how the adapter connects to and interacts with other Java CAPS components within the Environment. The Environment properties are accessed from the NetBeans IDE Services window.

▼ To Configure the Environment Properties

1   From the NetBeans Services window, expand the CAPS Environment node.

2   Expand the Environment created for your project and locate the External System for your specific adapter.

3   Right-click the External System and select Properties from the popup menu. The Environment Configuration Properties window appears.
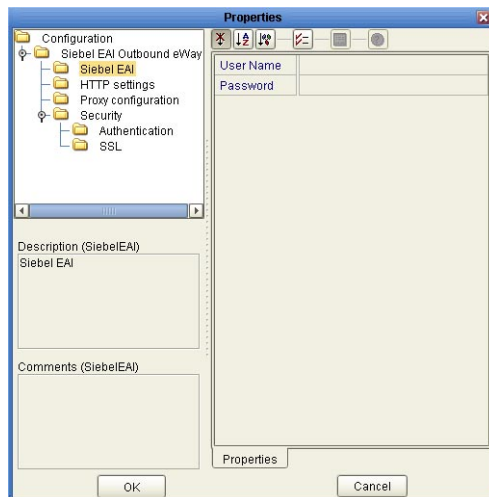
FIGURE 1   Adapter Environment Configuration Properties Editor

4   From the Properties Editor, click on any folder to display the default configuration properties for that section.

5 **Click on any property field to make it editable.**

6 **Once you have finished modifying the properties, click** OK **to save your changes and close the editor.**

Configuring Batch Adapter Batch Inbound Environment Properties

This topic explains the configuration parameters for the BatchInbound Adapter (OTD), accessed from the Environment Explorer.

MDB Settings (BatchInbound Environment)

The MDB Settings section of the BatchInbound Environment properties contains the top-level parameters displayed in the following table.

TABLE 1 Connectivity Map - BatchInbound- Settings

Name	Description	Required Value
Max Pool Size	Specifies the maximum size of the MDB (Message Driven Bean) pool.	An integer indicating the maximum MDB pool size. The configured default is **1000**.

Configuring TCP/IP Adapter Environment Properties

The TCP/IP Adapter Environment properties are organized into the following sections:

- "TCPIP Server (Inbound) Adapter - General Inbound Settings" on page 9.
- "TCPIP Server (Inbound) Adapter - TCPIP Inbound Settings" on page 10.
- "TCPIP Server (Inbound) Adapter - MDB Pool Settings" on page 11.
- "TCPIP Client (Outbound) Adapter - General Outbound Settings" on page 13.
- "TCPIP Client (Outbound) Adapter - TCPIP Outbound Settings" on page 14.
- "TCPIP Client (Outbound) Adapter - Connection Pool Settings" on page 14.

TCPIP Server (Inbound) Adapter - General Inbound Settings

The General Inbound Settings properties represents general TCPIP Inbound configuration information. The TCPIPServer (Inbound) Adapter - General Inbound Settings properties contain the top-level parameters displayed in the following table.

TABLE 2 TCPIPServer (Inbound) Adapter - General Inbound Settings Properties

Name	Description	Required Value
Persistence State File Location	Specifies the directory location (a local folder name) where the state files, used to persist the state value, are stored. This property is required when the Scope Of State is set to Persistence.	The file and path. The default value is **C:/temp/tcpipinbound/state** (depending on the environment settings).

TCPIP Server (Inbound) Adapter - TCPIP Inbound Settings

Specifies the Java Socket and ServerSocket options. For more information, refer to the *JDK Javadoc*.

The TCPIPServer (Inbound) Adapter - TCPIP Inbound Settings properties contain the top-level parameters displayed in the following table.

TABLE 3 TCPIP Server (Inbound) Adapter - MDB Properties

Name	Description	Required Value
Host	Specifies the host name or IP address used to establish a TCPIP connection. This parameter is only used when Connection Type is Client.	A TCP/IP host name or IP address.
ServerPort	Specifies the port number of the TCP/IP destination. This is dependent upon the specified Connection Type. If the value for Connection Type is: ■ **Server**: the ServerPort value is set to the port number on the local host. ■ **Client**: the ServerPort value is set to the port number of the external host.	An integer between 0 and 65535, indicating the port number of the TCP/IP destination. The port number of the TCP/IP destination. The default is **8888**. **Note –** TCP/IP server connects binds to random port, if the port number is 0.

TABLE 3 TCPIP Server (Inbound) Adapter - MDB Properties   *(Continued)*

Name	Description	Required Value
Backlog	Specifies the maximum length of the queue when creating the ServerSocket. The maximum queue length for incoming connection indications (a request to connect) is set to the backlog parameter. If a connection indication arrives when the queue is full, the connection is refused. **Note –** This parameter is only used when Connection Type is set to Server.	An integer indicting the queue length for incoming connections. The configured default value is **50**.

TCPIP Server (Inbound) Adapter - MDB Pool Settings

Specific to the MDB bean pool of Sun Java™ System Application Server and Sun Enterprise Service Bus only. The parameter settings in this section are applied to `sun-ejb-jar.xml`.

The TCPIPServer (Inbound) Adapter - MDB Pool Settings properties contain the top-level parameters displayed in the following table.

TABLE 4 TCPIPServer (Inbound) Adapter - MDB Pool Settings Properties

Name	Description	Required Value
Steady Pool Size	Specifies the minimum number of MDB beans to be maintained. When the value is set to a value that is greater than 0, the container not only pre-populates the MDB bean pool with the specified number, but also attempts to ensure that there are always this many MDB beans in the free pool. This ensures that there are enough MDB beans in the "ready to serve" state to process user requests. This parameter does not guarantee that more than the "steady-pool-size" MDB instances will not exist at a given time. It only governs the number of instances that are pooled over a long period of time. For example, suppose an idle stateless session container has a fully-populated pool with a steady-pool-size of 10. If 20 concurrent requests arrive for the MDB bean component, the container creates 10 additional instances to satisfy the burst of requests. This prevents the container from blocking any of the incoming requests. However, if the activity dies down to 10 or fewer concurrent requests, the additional 10 instances are discarded.	An integer indicating the minimum number of Message Driven Beans (MDBs) to be maintained. The configured default is **10**.
Max Pool Size	Specifies the maximum number of MDB beans in the pool.	An integer indicating the maximum number of MDB beans in the pool. A value of **0** means the pool is unbounded. The configured default is **60**.

TABLE 4   TCPIPServer (Inbound) Adapter - MDB Pool Settings Properties   *(Continued)*

Name	Description	Required Value
Pool Idle Timeout In Seconds	Specifies the interval at which the "*remove expired MDBs*" thread runs. This thread periodically removes unused MDB beans with expired timeouts. This provides a hint to the server, and allows the user to specify the maximum amount of time that an MDB bean instance can remain idle in the pool. After this period of time, the pool can remove this bean. When Pool Idle Timeout In Seconds is set to a value greater than 0, the removes or destroys any MDB bean instance that is idle for this specified duration. A value of 0 specifies that idle MDB beans can remain in the pool indefinitely.	An integer indicting the Pool Idle Timeout in seconds for unused MDBs. A value of 0 specifies that idle MDBs can remain in the pool indefinitely. The configured default value is **600**.

TCPIP Client (Outbound) Adapter - General Outbound Settings

The General Outbound Settings properties represents general TCPIP outbound configuration information. The TCPIP Client (Outbound) Adapter - General Outbound Settings properties contain the top-level parameters displayed in the following table.

TABLE 5   TCPIPClient (Outbound) Adapter - General Outbound Settings Properties

Name	Description	Required Value
Persistence State File Location	Specifies the File Location (a local folder name). This property is required when the Scope Of State value is set to is Persistence. This file is used to store the state files which are used to persist the state value.	The file and path. The default value is **/temp/tcpipoutbound/state**.

TCPIP Client (Outbound) Adapter - TCPIP Outbound Settings

The TCPIP Outbound Settings properties represents general TCPIP outbound configuration information. The TCPIP Client (Outbound) Adapter - TCPIP Outbound Settings properties contain the top-level parameters displayed in the following table.

TABLE 6 TCPIPClient (Outbound) Adapter - TCPIP Outbound Settings Properties

Name	Description	Required Value
Host	Specifies the host name or IP address used to establish a TCPIP connection. This parameter is only used when the Connection Type is set to Client.	A TCP/IP host name or IP address. The configured default is **localhost**.
ServerPort	Specifies the port number of the TCP/IP destination. This is dependent upon the specified Connection Type. If the value for Connection Type is: ■ **Server**: The ServerPort value is set to the port number on the local host. ■ **Client**: The ServerPort value is set to the port number of the external host.	The port number of the TCP/IP destination. The default is **7777**.
Backlog	Specifies the maximum length of the queue when creating the ServerSocket. The maximum queue length for incoming connection indications (a request to connect) is set to the backlog parameter. If a connection indication arrives when the queue is full, the connection is refused. **Note** – This parameter is only used when Connection Type is set to Server.	An integer indicting the queue length for incoming connections. The configured default value is **50**.

TCPIP Client (Outbound) Adapter - Connection Pool Settings

Specific to the RA connection pool of Sun Java System Application Server or the Sun Enterprise Service Bus only. The parameter settings in this section are applied to sun-ra.xml.

The TCPIPClient (Outbound) Adapter - Connection Pool Settings properties contain the top-level parameters displayed in the following table.

TABLE 7 TCPIPClient (Outbound) Adapter - Connection Pool Settings Properties

Name	Description	Required Value
Steady Pool Size	Specifies the minimum number of RA connections to be maintained. When the value is set to a value that is greater than 0, the container not only pre-populates the RA connection pool with the specified number, but also attempts to ensure that there are always this many RA connections in the free pool. This ensures that there are enough RA connections in the "ready to serve" state to process user requests. For example, suppose an idle stateless session container has a fully-populated pool with a steady-pool-size of 10. If 20 concurrent requests arrive for the RA Connection component, the container creates 10 additional instances to satisfy the burst of requests. This prevents the container from blocking any of the incoming requests. However, if the activity dies down to 10 or fewer concurrent requests, the additional 10 instances are discarded.	An integer indicating the minimum number of RA connections to be maintained. The configured default is **1**.
Max Pool Size	Specifies the maximum number of RA connections in the pool.	An integer indicating the maximum number of RA connections in the pool. A value of **0** indicates that the pool is unbounded. The configured default is **32**.
Pool Idle Timeout In Seconds	Specifies the interval at which the "*remove expired RA connections*" thread runs. This thread periodically removes unused RA connections with expired timeouts. This provides a hint to the server, and allows you to specify the maximum amount of time that an RA connection instance can remain idle in the pool. After this period of time, the pool can remove this bean. When Pool Idle Timeout In Seconds is set to a value greater than 0, the container removes or destroys any RA connection instance that is idle for this specified duration. A value of 0 specifies that idle RA connections can remain in the pool indefinitely.	An integer indicting the Pool Idle Timeout in seconds for unused RA connections. A value of 0 specifies that idle RA connections can remain in the pool indefinitely. The configured default value is **600**.

Configuring TCP/IP HL7 Adapter Environment Explorer Properties

The TCP/IP HL7 Adapter configuration parameters accessed from the Environment Explorer tree apply to both the inbound and outbound Adapters. The TCP/IP Adapter's Environment properties are organized into the following sections:

HL7 Inbound Adapter - TCPIP Inbound Settings

The HL7 Inbound Adapter - TCPIP Inbound Settings present the java Socket and ServerSocket options. For more information, please refer *JDK javadoc*.

This section of the TCP/IP HL7 Adapter Environment properties contains the top-level parameters displayed in the following table.

TABLE 8 Environment - HL7 Inbound Adapter - TCPIP Inbound Settings

Name	Description	Required Value
Host	Specifies the host name or IP address used to establish a TCP/IP connection. This parameter is only used when the Connection Type is set to Client.	The host name or IP address to used to establish a TCPIP connection.
ServerPort	Specifies the port number of the TCP/IP destination. This is dependent on the value set for Connection Type, ■ If Connection Type is set to Server, it indicates the port number on the local host. ■ If Connection Type is set to Client, it indicates the port number of the external host.	An integer between **0 and 65535**, indicating the port number of the TCP/IP destination. The configured default is **8888**.
Backlog	Specifies the maximum length of the queue when creating the ServerSocket. The maximum queue length for the incoming connection indications (request to connect) is set to the Backlog parameter. When a connection indication arrives and the queue is full, the connection is refused. This parameter is only used when Connection Type is set to Server.	An integer indicating the maximum length of the queue when creating the ServerSocket. The configured default is **50**.

HL7 Inbound Adapter - MDB Pool Settings

The HL7 Inbound Adapter - MDB Pool Settings are specific to the MDB bean pool of Sun Java System Application Server or Sun Runtime Server only. The parameter settings in this section will go into `sun-ejb-jar.xml`.

This section of the TCP/IP HL7 Adapter Environment properties contains the top-level parameters displayed in the following table.

TABLE 9   Environment - HL7 Inbound Adapter - TCPIP Inbound Settings

Name	Description	Required Value
Steady Pool Size	Specifies the minimum number of MDB beans to be maintained. When the value is set to a number greater than 0 (zero), the container not only pre-populates the MDB bean pool with the specified number, but also attempts to ensure that there is always this many MDB beans in the free pool. This ensures that there are enough MDB beans in the ready-to-serve state to process user requests. This parameter does not necessarily guarantee that no more than steady-pool-size instances exist at a given time. It only governs the number of instances that are pooled over a long period of time. For example, suppose an idle stateless session container has a fully-populated pool with a steady-pool-size of 10. If 20 concurrent requests arrive for the MDB bean component, the container creates 10 additional instances to satisfy the burst of requests. The advantage of this is that it prevents the container from blocking any of the incoming requests. However, if the activity dies down to 10 or fewer concurrent requests, the additional 10 instances are discarded.	An integer indication the minimum number of MDB beans to be maintained. The configured default is **10**.
Max Pool Size	Specifies the maximum number of MDB beans in the pool.	An integer indication the maximum number of MDB beans in the pool. A value of **0** (zero) indicates that the pool is unbounded. The configured default is **60**.

TABLE 9 Environment - HL7 Inbound Adapter - TCPIP Inbound Settings   *(Continued)*

Name	Description	Required Value
Pool Idle Timeout in Seconds	Specifies the maximum amount of time (in seconds) that an MDB bean instance can remain idle in the pool. When an MDB has exceeded the configured Pool Idle Timeout, a timer thread removes the unused MDB bean. This property defines the interval at which this thread runs. A value greater than 0 indicates that the container removes or destroys any MDB bean instance that is idle at this specified duration. A value of 0 (zero) specifies that idle MDB beans can remain in the pool indefinitely.	The maximum amount of time (in seconds) that an MDB bean instance can remain idle in the pool. The configured default is **600** (10 minutes).

HL7 Inbound Adapter - Sequence Number Protocol

The HL7 Inbound Adapter - Sequence Number Protocol provides configuration settings used to help prevent duplication of data.

This section of the TCP/IP HL7 Adapter Environment properties contains the top-level parameters displayed in the following table.

TABLE 10 Environment - HL7 Inbound Adapter - Sequence Number Protocol

Name	Description	Required Value

Sequence Number File Location	Specifies the location of the sequence number file (a local directory). This is required when the Sequence Number Protocol is enabled. The sequence number file is a non-volatile directory that stores the sequence number files that are used to persist the HL7 sequence number. This unique base file name is automatically generated according to Project/Collaboration information.
	
	 For the Inbound Adapter the file names are created as follows:
	
	 `<Project name> + <deployment name> + <external application node name>  + <Collaboration name> + .seqno`
	
	 For example:
	
	 `prjHL7Inbound_dpIn_eaHL7Inbound_ jcdHL7inbound1.seqno`
		The path and directory where the sequence number file is located.
		
		 The default setting is:
		
		 `/temp/hl7inbound/seq`

HL7 Outbound Adapter - TCPIP Outbound Settings

Presents the java Socket options. For more information, please refer *JDK javadoc*.

This section of the TCP/IP HL7 Adapter Environment properties contains the top-level parameters displayed in the following table.

TABLE 11 Environment - HL7 Outbound Adapter - TCPIP Outbound Settings

Name	Description	Required Value
Host	Specifies the host name or IP address used to establish a TCPIP connection. This parameter is only used when the Connection Type is set to Client.	The host name or IP address used to establish a TCPIP connection. The configured default is **localhost**.
ServerPort	Specifies the port number of the TCP/IP destination. This is dependent on the value set for Connection Type. ■ If Connection Type is set to Server, it indicates the port number on the local host. ■ If Connection Type is set to Client, it indicates the port number of the external host.	An integer between **0 and 65535**, indicating the port number of the TCP/IP destination. The configured default is **7777**.
Backlog	Specifies the maximum length of the queue when creating the ServerSocket. The maximum queue length for the incoming connection indications (request to connect) is set to the Backlog parameter. When a connection indication arrives and the queue is full, the connection is refused. This parameter is only used when Connection Type is set to Server.	An integer indicating the maximum length of the queue when creating the ServerSocket. The configured default is **50**.

HL7 Outbound Adapter - Connection Pool Settings

The HL7 Outbound Adapter - Connection Pool Settings are specific to the RA connection pool of Sun Java System Application Server (Sun One Application Server) or Sun Runtime Server only. The parameter settings in this section will go into `sun-ra.xml`.

This section of the TCP/IP HL7 Adapter Environment properties contains the top-level parameters displayed in the following table.

TABLE 12 Environment - HL7 Outbound Adapter - Connection Pool Settings

Name	Description	Required Value
Steady Pool Size	Specifies the minimum number of RA connections to be maintained. When the value is set to a number greater than 0 (zero), the container not only pre-populates the RA connection pool with the specified number, but also attempts to ensure that there is always this many RA connections in the free pool. This ensures that there are enough RA connections in the ready-to-serve state to process user requests. This parameter does not necessarily guarantee that no more than steady-pool-size instances exist at a given time. It only governs the number of instances that are pooled over a long period of time. For example, suppose an idle stateless session container has a fully-populated pool with a steady-pool-size of 10. If 20 concurrent requests arrive for the RA connection component, the container creates 10 additional instances to satisfy the burst of requests. The advantage of this is that it prevents the container from blocking any of the incoming requests. However, if the activity dies down to 10 or fewer concurrent requests, the additional 10 instances are discarded.	An integer indication the minimum number of RA connections to be maintained. The configured default is **1**.
Max Pool Size	Specifies the maximum number of RA connections in the pool.	An integer indication the maximum number of RA connections in the pool. A value of **0** (zero) indicates that the pool is unbounded. The configured default is **32**.
Pool Idle Timeout in Seconds	Specifies the maximum amount of time (in seconds) that an RA connections instance can remain idle in the pool. When an RA connection has exceeded the configured Pool Idle Timeout, a timer thread removes the unused RA connection. This property defines the interval at which this thread runs. A value greater than **0** indicates that the container removes or destroys any RA connection instance that is idle at this specified duration. A value of 0 (zero) specifies that idle RA connections can remain in the pool indefinitely.	The maximum amount of time (in seconds) that an RA connections instance can remain idle in the pool. The configured default is **300** (5 minutes).

HL7 Outbound Adapter - Sequence Number Protocol

The HL7 Outbound Adapter - Sequence Number Protocol provides configuration settings used to help prevent duplication of data.

This section of the TCP/IP HL7 Adapter Environment properties contains the top-level parameters displayed in the following table.

TABLE 13 Environment - HL7 Outbound Adapter - Sequence Number Protocol

Name	Description	Required Value

Sequence Number File Location	Specifies the location of the sequence number file (a local directory). This is required when the Sequence Number Protocol is enabled. The sequence number file is a non-volatile directory that stores the sequence number files that are used to persist the HL7 sequence number. This unique base file name is automatically generated according to Project/Collaboration information.
	
	For the Outbound Adapter the file names are created as follows:
	
	`<Project name> + <deployment name> + <collab name> + <externalapplication node name> + .seqno`
	
	For example:
	
	`prjHL7Outbound_dpOut_jcolHL7Outbound_`
	`eaHL7Outbound.seqno`
		The path and directory where the sequence number file is located.
		
		The default setting is:
		
		`/temp/hl7outbound/seq`

Configuring EMail Adapter Environment Properties

The EMail Adapter configuration parameters, accessed from the EMail Adapter External System in the Environment Explorer tree, are organized into the following sections:

Note – Some EMail 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.

Inbound Email Adapter ⇒ Connection Settings

The Inbound Email Adapter ⇒ Connection Settings section of the EMail Adapter Environment properties contains the top-level parameters displayed in the following table.

TABLE 14 Environment - Inbound Email Adapter ⇒ Connection Settings

Name	Description	Required Value
Host Receive	Specifies the host name of the server used to receive messages. This is required for "receiving" Adapter connections. This is also required for "sending" Adapter connections when the SessionAuth parameter is set to **Yes** (for POP3 login).	The host name of the server used to receive messages.
Port Receive	Specifies the port number used to connect when receiving Email messages. This is required for "receiving" Adapter connections. This is also required for "sending" Adapter connections when the SessionAuth parameter is set to **Yes** (for POP3 login).	The port number used to connect when receiving Email messages. This is a number between **1** and **65535**. The configured default is **110**.
User Receive	Specifies the user name used when receiving messages. This is required for "receiving" Adapter connections. This is also required for "sending" Adapter connections when the SessionAuth parameter is set to **Yes** (for POP3 login).	The valid user login name used when receiving Email messages.
Password Receive	Specifies the password used when receiving messages. This is required for "receiving" Adapter connections. This is also required for "sending" Adapter connections when the SessionAuth parameter is set to **Yes** (for POP3 login).	The user password used when receiving messages.

Inbound Email Adapter ⇒ SSL

The Inbound Email Adapter ⇒ SSL section of the EMail Adapter Environment properties contains the top-level parameters displayed in the following table.

TABLE 15   Environment - Inbound Email Adapter ⇒ SSL

Name	Description	Required Value
Receive SSL Protocol	Specifies the SSL protocol to use when establishing an SSL connection with the server.	Select the appropriate SSL protocol. The options are: ■ No SSL ■ TLS ■ TLSv1 ■ SSLv3 ■ SSLv2 ■ SSL The configured default is **No SSL**.
X509 Algorithm Name	Specifies the X509 algorithm name to use for the trust and key manager factories.	An X509 algorithm name. The configured default is **SunX509**.

Inbound Email Adapter ⇒ SSL ⇒ CACerts

The Inbound Email Adapter ⇒ SSL ⇒ CACerts section of the EMail Adapter Environment properties contains the top-level parameters displayed in the following table.

TABLE 16   Environment - Inbound Email Adapter ⇒ SSL ⇒ CACerts

Name	Description	Required Value
TrustStore type	Specifies the type of truststore used for CA certificate management when establishing SSL connections.	The trustStore type. The configured default is **JKS**.
TrustStore	Specifies a truststore used for CA certificate management to establish SSL connections. A truststore file is a key database file that contains the public keys for a target server.	The truststore used for CA certificate management.
TrustStore password	Specifies the password for accessing the truststore used for CA certificate management when establishing SSL connections.	The truststore password.

Inbound Email Adapter ⇒ MDB Settings

The Inbound Email Adapter ⇒ MDB Settings section of the EMail Adapter Environment properties contains the top-level parameters displayed in the following table.

TABLE 17 Environment - Inbound Email Adapter ⟹ MDB Settings

Name	Description	Required Value
Max Pool Size	Specifies the maximum pool size. This controls the number of concurrent sessions.	An integer indicating the maximum pool size. The configured default is **10**.

Outbound Email Adapter ⟹ Connection Settings ⟹ Send SMTP

The Outbound Email Adapter ⟹ Connection Settings ⟹ Send SMTP section of the EMail Adapter Environment properties contains the top-level parameters displayed in the following table.

TABLE 18 Environment - Outbound Email Adapter ⟹ Connection Settings ⟹ Send SMTP

Name	Description	Required Value
Host Send	Specifies the host name of the server used to send messages. This is required for the "sending" Adapter connection.	The host name of the server used to send messages.
Port Send	Specifies the port number to connect to when sending messages. This is required for "sending" Adapter connections.	An integer indicating the port number. The configured default is **25**.
User Send	Specifies the user name used when sending messages. This is required for "sending" Adapter connections.	The login name used to access the sending host server.
Password Send	Specifies the password used when sending messages. This is required for "sending" Adapter connections.	The user password used to access the sending host server.
Text encoding	Specifies the encoding used for body and header text. Available encoding options are: ■ **ASCII**: for ASCII text. ■ **iso-8859-1**: Latin 1 (Western Europe) text. ■ **iso-2022-jp**: Japanese character text. ■ **ISO2022CN**: Chinese character text. ■ **ISO2022CN_GB**: Simplified Chinese character text. **ISO2022KR**: Korean character text.	Select one of the following: ■ ASCII ■ iso-8859-1 ■ iso-2022-jp ■ ISO2022CN ■ ISO2022CN_GB ■ ISO2022KR

(Continued)

Name	Description	Required Value
Header encoding	Specifies the encoding used for the header. Available encoding options are: ■ **B**: Identical to the "BASE64" encoding defined by RFC 1341. ■ **Q**: Designed to allow text containing mostly ASCII characters to be deciphered by an ASCII terminal without decoding. Q encoding is similar to "Quoted-Printable" content- transfer-encoding defined in RFC 1341. "Q" encoding is recommended for use with most Latin character sets, while "B" encoding is recommended for all others.	Select one of the following: ■ B ■ Q

Outbound Email Adapter ⇒ Connection Settings ⇒ Receive POP3

The Outbound Email Adapter ⇒ Connection Settings ⇒ Receive POP3 section of the EMail Adapter Environment properties contains the top-level parameters displayed in the following table.

TABLE 19 Environment - Outbound Email Adapter ⇒ Connection Settings ⇒ Receive POP3

Name	Description	Required Value
Host Receive	Specifies the host name of the server used to receive messages. This is required for "receiving" Adapter connections. This is also required for "sending" Adapter connections when the SessionAuth parameter is set to Yes (for POP3 login).	The host name of the server used to receive messages.
Port Receive	Specifies the port number to connect to when receiving messages. This is required for "receiving" Adapter connections. This is also required for "sending" Adapter connection when the Session Authentication parameter is set to Yes (for POP3 login).	An integer indicating the port number used to connect with the receiving host server. The configured default is **110**.
User Receive	Specifies the user name used when receiving messages. This is required for "receiving" Adapter connections. This is also required for "sending" Adapter connections when the Session Authentication parameter is set to Yes (for POP3 login).	The login name used to access the receiving host server.

TABLE 19   Environment - Outbound Email Adapter ⇒ Connection Settings ⇒ Receive POP3 *(Continued)*

Name	Description	Required Value
Password Receive	Specifies the password used when receiving messages. This is required for "receiving" Adapter connections. This is also required for "sending" Adapter connections when the Session Authentication parameter is set to Yes (for POP3 login).	The user password used to access the receiving host server.
Session Authentica-tion	Determines whether a POP3 session authentication is performed before attempting an SMTP connection. This is required by some e-mail services. Set the value to Yes only when necessary. Yes requires that settings for Host Receive, Port Receive, User Receive, and Password Receive are entered for sending the Adapter connection.	Select **Yes** or **No**. **Yes** indicates that POP3 session authentication will be performed before attempting an SMTP connection. The configured default is **No**.

Outbound Email Adapter ⇒ SSL

The Outbound Email Adapter ⇒ SSL section of the EMail Adapter Environment properties contains the top-level parameters displayed in the following table.

TABLE 20   Environment - Outbound Email Adapter ⇒ SSL

Name	Description	Required Value
Send SSL Protocol	Specifies the SSL protocol to use when establishing an SSL connection with the SMTP server.	Select the appropriate SSL protocol. The options are: ■ No SSL ■ TLS ■ TLSv1 ■ SSLv3 ■ SSLv2 ■ SSL The configured default is **No SSL**.

TABLE 20 Environment - Outbound Email Adapter ⇒ SSL   *(Continued)*

Name	Description	Required Value
Receive SSL Protocol	Specifies the SSL protocol to use when establishing an SSL connection with the server.	Select the appropriate SSL protocol. The options are: ■ No SSL ■ TLS ■ TLSv1 ■ SSLv3 ■ SSLv2 ■ SSL The configured default is **No SSL**.
X509 Algorithm Name	Specifies the X509 algorithm name to use for the trust and key manager factories.	An X509 algorithm name. The configured default is **SunX509**.

Outbound Email Adapter ⇒ SSL ⇒ CACerts

The Outbound Email Adapter ⇒ SSL ⇒ CACerts section of the EMail Adapter Environment properties contains the top-level parameters displayed in the following table.

TABLE 21 Environment - Outbound Email Adapter ⇒ SSL ⇒ CACerts

Name	Description	Required Value
TrustStore type	Specifies the type of truststore used for CA certificate management when establishing SSL connections.	The trustStore type. The configured default is **JKS**.
TrustStore	Specifies a truststore used for CA certificate management to establish SSL connections.	The truststore used for CA certificate management.
TrustStore password	Specifies the password used to access the truststore used for CA certificate management when establishing SSL connections.	The truststore password.

Configuring File Adapter Environment Properties

The File Adapter configuration parameters, accessed from the Environment Explorer, are organized into the following sections:

- "Inbound File Adapter - Parameter Settings" on page 28.
- "Inbound File Adapter - MDB Settings" on page 28.
- "Outbound File Adapter - Parameter Settings" on page 28.

Inbound File Adapter - Parameter Settings

The Inbound File Adapter - Parameter Settings section of the File Adapter Environment properties contains the top level parameters displayed in the following table.

TABLE 22   File Adapter Environment Properties - Inbound File Adapter - Parameter Settings

Name	Description	Required Value
Directory	Specifies the folder or directory that the Adapter polls for input files.	A folder or directory name. You must use the absolute path; specify paths using the (forward) / slash mark. The configured default is `C:/temp`.

Inbound File Adapter - MDB Settings

The Inbound File Adapter - MDB Settings section of the File Adapter Environment properties contains the top level parameters displayed in the following table.

TABLE 23   File Adapter Environment Properties - Inbound File Adapter - MDB Settings

Name	Description	Required Value
Max Pool Size	Specifies the maximum number of Message Driven Beans instantiated at any one point for message handling. The polling interval and the MDB pool size (Max Pool Size) can be "tuned" based on the expected volume and frequency of incoming messages. A more frequent polling interval, due to a large number of new messages, will trigger the creation of new threads. The number of threads that can be processed, however, is limited by the MDB pool size. When the maximum MDB pool limit is reached, incoming threads are blocked. Increasing the MDB pool size allows more resource adapter threads to send data to the Collaboration. In most cases, the default MDB pool size of 1000 is sufficient.	An integer indicating the maximum MDB pool size. The default configuration is **1000**.

Outbound File Adapter - Parameter Settings

The Outbound File Adapter - Parameter Settings section of the File Adapter Environment properties contains the top level parameters displayed in the following table.

TABLE 24 File Adapter Environment Properties - Outbound File Adapter - Parameter Settings

Name	Description	Required Value
Directory	Specifies the directory to which output files are written.	The absolute path and directory name.

Configuring MSMQ Adapter Environment Properties

The MSMQ Adapter Environment properties are organized into the following sections:

Inbound MSMQ Adapter — MSMQ Environment

The Inbound MSMQ Adapter — MSMQ Environment section of the MSMQ Adapter Environment properties contains the top-level properties displayed in the following table.

TABLE 25 Environment - Inbound MSMQ Adapter — MSMQ Environment

Name	Description	Required Value
MSMQ Host Name	Specifies the Microsoft Message Queue host name.	The Microsoft Message Queue host name. Avoid using an IP address or "localhost" for the local server. These may not work in your MSMQ Environment. **Note –** 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.

Inbound MSMQ Adapter — MDB Settings

The Inbound MSMQ Adapter — MDB Settings section of the MSMQ Adapter Environment properties contains the top-level properties displayed in the following table.

TABLE 26   Environment - Inbound MSMQ Adapter — MDB Settings

Name	Description	Required Value
Max Pool Size	Specifies the maximum number of physical connections the pool can contain. A value of **0** (zero) indicates that there is no maximum.	An integer indicating the maximum pool size. The configured default is **1000**.

Inbound MSMQ Adapter — Connection Retry Settings

The Inbound MSMQ Adapter — Connection Retry Settings section of the MSMQ Adapter Environment properties contains the top-level properties displayed in the following table.

TABLE 27   Environment - Inbound MSMQ Adapter — Connection Retry Settings

Name	Description	Required Value
Connec tion Retry Count	Specifies the maximum number of attempts made to connect to the destination queue manager or queue. If the queue manager or queue cannot be accessed for any reason, this setting specifies how many reattempts are made to complete the processing.	An integer indicating the maximum number of connection attempts. The configured default is **0**.
Connec tion Retry Interval	Specifies the amount of time (in milliseconds) between attempts to connect to the destination queue manager or queue. This is the pause between each reattempt to access the destination queue manager or queue. Used in conjunction with the **Connection Retry Count** setting.	An integer indicating the wait time in milliseconds between connection attempts. The configured default is **1000**.

Outbound MSMQ Adapter — MSMQ Environment

The Outbound MSMQ Adapter — MSMQ Environment section of the MSMQ Adapter Environment properties contains the top-level properties displayed in the following table.

TABLE 28 Environment - Outbound MSMQ Adapter — MSMQ Environment

Name	Description	Required Value
MSMQ Host Name	Specifies the Microsoft Message Queue host name.	The Microsoft Message Queue host name. Avoid using an IP address or "localhost" for the local server. These may not work in your MSMQ Environment. **Note** – 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.

Outbound MSMQ Adapter — Connection Retry Settings

The Outbound MSMQ Adapter — Connection Retry Settings section of the MSMQ Adapter Environment properties contains the top-level properties displayed in the following table.

TABLE 29 Environment - Outbound MSMQ Adapter — Connection Retry Settings

Name	Description	Required Value
Connection Retry Count	Specifies the maximum number of attempts made to connect to the destination queue manager or queue. If the queue manager or queue cannot be accessed for any reason, this setting specifies how many reattempts are made to complete the processing.	An integer indicating the maximum number of connection attempts. The configured default is **0**.
Connection Retry Interval	Specifies the amount of time (in milliseconds) between attempts to connect to the destination queue manager or queue. This is the pause between each reattempt to access the destination queue manager or queue. Used in conjunction with the Connection Retry Count setting.	An integer indicating the wait time in milliseconds between connection attempts. The configured default is **1000**.

Configuring Batch Adapter BatchFTP Environment Properties

This topic describes the configuration properties for the BatchFTP OTD accessed from the Environment Explorer tree.

The BatchFTP Environment Explorer properties include the following sections:

- "SOCKS (BatchFTP Environment)" on page 32.
- "FTP (BatchFTP Environment)" on page 33.
- "General Settings (BatchFTP Environment)" on page 35.
- "SSH Tunneling (BatchFTP Environment)" on page 36.
- "Connection Pool Settings (BatchFTP Environment)" on page 38.
- "Connection Retry Settings (BatchFTP Environment)" on page 39.

> ⚠ **Caution –** Several of these configuration options allow you to use regular expressions. This advanced feature is useful but must be used carefully. An improperly formed regular expression can cause the creation of undesired data or even the loss of data. You must have a clear understanding of regular-expression syntax and construction before attempting to use this feature. It is recommended that you test such configurations thoroughly before moving them to production.

SOCKS (BatchFTP Environment)

This topic provides information for configuring the SOCKS properties (accessed from the Environment Explorer). The BatchFTP Adapter supports the negotiation methods, No-authentication and User/password.

The SOCKS section of the BatchFTP Environment properties contains the top-level parameters displayed in the following table.

TABLE 30 Environment - BatchFTP - SOCKS

Name	Description	Required Value
Socks Host Name	Specifies the SOCKS server (host) name. If you are communicating with a SOCKS server enter the SOCKS server name in this parameter.	The name of the SOCKS server.
Socks Server Port	Specifies the port number to use on the SOCKS server, when connecting to it.	An integer from **1** to **65,535**. The configured default is **1080**.

TABLE 30 Environment - BatchFTP - SOCKS *(Continued)*

Name	Description	Required Value
Socks User Name	Specifies the user name to use (together with the password specified under the Socks Password parameter) for authentication with a SOCKS5 server, if necessary. This parameter is used for the user/password negotiation method.	A valid SOCKS5 user name.
Socks Password	Specifies the password to use (together with the user name specified under the Socks User Name parameter) for authentication with a SOCKS5 server, if necessary. This parameter is used for the user/password negotiation method. **Note –** The corresponding Java accessors are `getSocksPassword()`, `setSocksPassword(java.lang.String p)` and `setSocksEncryptedPassword(java.lang.String p)`.	A valid SOCKS5 password.

FTP (BatchFTP Environment)

The FTP section of the BatchFTP Environment properties contains the top-level parameters displayed in the following table.

TABLE 31   Environment - BatchFTP - FTP

Name	Description	Required Value
Host Name	Specifies the name of the external system that the Adapter connects to. ■ If the parameter SSH Tunneling Enabled under the SSH Tunneling configuration settings is set to Yes, the parameters Host Name and Server Port, under the FTP settings, are ignored. In this case, the FTP host name is determined by an SSH option, according to the following model: **ssh -L ListenPort:FtpServerHost:FtpServerPort SSHServer** In the previous example, the FTP feature communicates with the FTP server *FtpServerHost:FtpServerPort* using an existing SSH tunnel. See *SSH Tunneling* (BatchFTP Connectivity Map) for details. ■ If the parameter Socks Enabled under the SOCKS configuration parameters is set to Yes, the host name under the FTP configuration could fail to resolve some names, for example, localhost or 127.0.0.1 correctly. Use real IP or machine names to represent the hosts. See *SOCKS* (BatchFTP Connectivity Map) for details.	The host name.
Server Port	Specifies the port number to use on the FTP server when connecting to it. If the parameter SSH Tunneling Enabled under the SSH Tunneling configuration is set to Yes, the parameters Host Name and Server Port under the FTP configuration are ignored. In this case, the FTP server port number is determined by an SSH option, according to the following model: **ssh -L ListenPort:FtpServerHost:FtpServerPort SSHServer** In the previous example, the FTP feature communicates with the FTP server *FtpServerHost:FtpServerPort* using an existing SSH tunnel. See *SSH Tunneling* (BatchFTP Connectivity Map) for details.	The server port number.
User Name	Specifies the user name used to log onto the external system, when required.	A user name that provides access to the external system.

TABLE 31 Environment - BatchFTP - FTP   *(Continued)*

Name	Description	Required Value
Password	If a password is required to log on to an external system, enter the password that corresponds to the user name. The corresponding Java accessor methods are `getPassword()`, `setPassword()`, and `setEncryptedPassword()`.	The password.
User Defined Heuristics Configuration File	Specifies the name and location of the user defined FTP heuristics configuration file. The format of the files content must be in the same form as that of the `FTPHeuristics.cfg` file. See **To Create a Custom Heuristics Configuration File** in *Configuring Java CAPS Project Components for Communication Adapters* for more details. This property works in conjunction with the User Defined Directory Listing Style property. **Note –** The BatchFTP OTD will generate an exception if a selected User Defined Directory Listing Style or the User Defined Heuristics Configuration File path is not defined correctly. If a User Defined Directory Listing Style is specified, a corresponding value must also be provided for the User Defined Heuristics Configuration File property.	The location and name of the user defined FTP heuristics configuration file on the local host.

General Settings (BatchFTP Environment)

The General Settings section of the BatchFTP Environment properties contains the top-level parameters displayed in the following table.

TABLE 32   Environment - BatchFTP - General Settings

Name	Description	Required Value
State Persistence Base Location	Specifies a working directory for storing intermediary results. The options are: ■ Leave value blank: BatchFTP will use a default folder as the working directory. ■ Specify a path to a local file system folder with read/write permissions. **Note –** See *Deploying an EAR File* if you are deploying the project to another application server.	A working directory with read/write permissions, or leave blank (no value) to accept a default directory.
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**.

SSH Tunneling (BatchFTP Environment)

This topic provides information for configuring the SSH Tunneling properties (accessed from the Environment Explorer). If Secure FTP (FTP over SSH or FTP over SSL) is required, use the Secure FTP OTDs (BatchFTPOverSSL, BatchSFTP, and BatchSCP). SSH Tunneling is supported for compatibility purposes.

The SSH Tunneling section of the BatchFTP Environment properties contains the top-level parameters displayed in the following table.

TABLE 33 Environment - BatchFTP - SSH Tunneling

Name	Description	Required Value
SSH Listen Host	Specifies the name of the host where the SSH support software runs, and to which the host listens. This parameter is required when and only when SSH Channel Established is set to Yes. The reason for this is, if you choose No this Listen Host will always be localhost because the SSH client will always be started from localhost. For optimum security, it is recommended that you use localhost as your choice.	The SSH listen host name. The configured default is **localhost.**
	The connection to the corresponding port number on this host is forwarded to the FTP server through an SSH-secure channel.	
	On the listen host, the SSH support software must be configured and started with the Port-Forwarding option.	
	The FTP command connection is forwarded through the secure tunnel. The corresponding SSH command uses the following model:	
	ssh -L ListenPort:FtpServerHost:FtpServerPort -o BatchMode=yes SSHServer	
	If this host name is not localhost, the data transport between the local host and the SSH listen host is not secure. Also, your SSH support software should be configured to allow connections to other hosts (for some SSH clients, it is an option -g).	
	For example, on an SSH listen host, you could issue a command, such as:	
	ssh -L 4567:apple:21 -o BatchMode=yes apple	
	or	
	ssh -L 5678:orange: 21 -o BatchMode=yes apple	
	Regardless, the transport between the SSH listen host and the FTP server is still secure.	

TABLE 33   Environment - BatchFTP - SSH Tunneling   *(Continued)*

Name	Description	Required Value
SSH Listen Port	Specifies the port number that the SSH-tunneling support software uses to check for incoming connections. This port number can be any unused port number on the SSH listen host. The connection to this port is forwarded to the FTP server through an SSH-secure channel. This parameter is required and it must be exactly same as the `ListenPort` value in the SSH command you issue either inside or outside the Sun Enterprise Service Bus system. The corresponding SSH command line uses the following model: `ssh -L ListenPort:FtpServerHost:FtpServerPort -o BatchMode=yes SSHServer Required Values`	An integer from **1** to **65535**.
SSH User Name	Specifies an SSH user name. This parameter can be required when the setting for the SSH Channel Established parameter is No. This parameter is required only if the SSH support software is started from within the Adapter (refer to the corresponding SSH command line). Even then, it is only required if your SSH implementation executes in an interactive way that requires you to enter a user name. Again, this requirement depends on how you specify the SSH command line and how your SSH environment is configured.	The SSH user name.
SSH Password	Specifies an SSH password corresponding to the user name entered under SSH User Name. This parameter can be required only when the setting for the SSH Channel Established parameter is No. For more information, see *SSH User Name*.	The SSH password.

Connection Pool Settings (BatchFTP Environment)

The Connection Pool Settings section of the BatchFTP Environment properties contains the top-level parameters displayed in the following table.

TABLE 34 Environment - BatchFTP - Connection Pool Settings

Name	Description	Required Value
Steady pool size	Specifies the minimum number of physical EIS connections that the pool keeps available at all times.	An integer indicating the maximum number of connections available at all times. A value of **0** (zero) indicates that there are no physical connections in the pool and that new connections are created as needed. The configured default is **2**.
Maximum pool size	Specifies the maximum number of physical EIS connections the pool contains.	An integer indicating the maximum number of connections allowed. A value of 0 (zero) indicates that there is no maximum. The default value is **10**.
Max Idle Timeout In Second	Specifies the maximum idle timeout (in seconds). This is a hint to the server. A timer thread periodically removes unused connections. This parameter defines the interval at which this thread runs. This thread removes unused connections after the specified idle time expires. It allows the user to specify the maximum amount of time that a connection can remain in the pool.	An integer indicating the maximum idle timeout in seconds. When this is set to a number greater than 0 (zero), the container removes or destroys any connections that are idle for the specified duration. A value of 0 specifies that idle connections can remain in the pool indefinitely. The configured default is **300** (5 minutes).

Connection Retry Settings (BatchFTP Environment)

The Connection Retry Settings section The Connection Pool Settings section of the BatchFTP Environment properties contains the top-level parameters displayed in the following table.

TABLE 35 Environment - BatchFTP - Connection Retry Settings

Name	Description	Required Value
Connection Retries	Specifies the number of retries to establish a connection upon failure to acquire a connection.	An integer indicating the maximum number of retries to establish a connection upon failure to acquire a connection. The Configured default value is **0**.

TABLE 35   Environment - BatchFTP - Connection Retry Settings   *(Continued)*

Name	Description	Required Value
Connection Retry Interval	Specifies the length of time (in milliseconds) between each reattempt to access the destination file. This is used in conjunction with the Connection Retries setting.	An integer indicating length of the pause (in milliseconds). The configured default value is **1000** (1 second).

Configuring Batch Adapter BatchFTPOverSSL Environment Properties

This topic describes the configuration properties for the BatchFTPOverSSL OTD accessed from the Environment Explorer.

The BatchFTPOverSSL Adapter Environment properties include the following sections:

- "FTP and SSL Settings (BatchFTPOverSSL Environment)" on page 40.
- "Firewall Settings (BatchFTPOverSSL Environment)" on page 41.
- "General Settings (BatchFTPOverSSL Environment)" on page 42.
- "Connection Pool Settings (BatchFTPOverSSL Environment)" on page 43.
- "Connection Retry Settings (BatchFTPOverSSL Environment)" on page 44.

FTP and SSL Settings (BatchFTPOverSSL Environment)

The FTP and SSL Settings section of the BatchFTPOverSSL Environment properties contains the top-level parameters displayed in the following table.

TABLE 36   Environment - BatchFTPOverSSL - FTP and SSL Settings

Name	Description	Required Value
FTP Host	Specifies the FTP server host name or IP address.	The FTP server host name or IP address. The configured default is **localhost**.
Explicit port for FTP over SSL	Specifies the FTP port for explicit SSL. default is 21 (data port 20).	The FTP port number for explicit SSL. The configured default is **21** (data port 20).

TABLE 36 Environment - BatchFTPOverSSL - FTP and SSL Settings *(Continued)*

Name	Description	Required Value
Implicit port for FTP over SSL	Specifies the FTP port for implicit SSL.	The FTP port for implicit SSL. The configured default is **990** (data port 989).
User ID	Specifies the user login for FTP server.	The user login name for FTP server.
Password	Specifies the password for the FTP server user login.	The password for the FTP server user login.
Key Store Location	Specifies the path to the keystore that contains the trusted CA certificates required for server authentication.	The fully qualified path to the keystore file.
Key Store Password	Specifies the password to access the keystore file.	The password for the keystore.
Key Store Type	Specifies the keystore format type. Selections include JKS or other. **Note –** JKS is currently the only supported keystore type. If "other" is selected as the value, an exception is thrown when the OTD is initialized, with the error message, "Unknown type key store".	Select **JKS** is currently the only supported keystore type, and the configured default.

Firewall Settings (BatchFTPOverSSL Environment)

The Firewall Settings section of the BatchFTPOverSSL Environment properties contains the top-level parameters displayed in the following table.

TABLE 37 Environment - BatchFTPOverSSL - Firewall Settings

Name	Description	Required Value
Firewall Host	Specifies the proxy server host name or IP.	The proxy server host name or IP address.
Firewall Port	Specifies the proxy server port.	The proxy server port number.
User ID	Specifies the user login on the proxy server.	User login ID.
Password	Specifies the password for the user login.	The user password.

General Settings (BatchFTPOverSSL Environment)

The General Settings section of the BatchFTPOverSSL Environment properties contains the top-level parameters displayed in the following table.

TABLE 38   Environment - BatchFTPOverSSL - General Settings

Name	Description	Required Value
Connection Mode	Specifies whether a physical connection is established when an external connection is instantiated. 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). If BatchFTPOverSSL Connection Mode is set to **Automatic**, the following Environment parameters must be set with valid values for the FTP and SSL Settings: ■ **FTP Host** ■ **Explicit port for FTP over SSL** (this must be set if the BatchFTPOverSSL Connectivity Map property, Secure Mode is set to Explicit SSL) ■ **Implicit port for FTP over SSL** (this must be set if the BatchFTPOverSSL Connectivity Map property, Secure Mode is set to Implicit SSL) ■ **User ID** ■ **Password**	Select **Automatic** or **Manual**. The default is **Automatic**.

TABLE 38 Environment - BatchFTPOverSSL - General Settings   *(Continued)*

Name	Description	Required Value
	■ **Key Store Location** (this must be set if the BatchFTPOverSSL Connectivity Map property, Require Server Authentication is set to Yes, or if the BatchFTPOverSSL Connectivity Map property, Distinguished Name for User contains a value, indicating that client authentication is required) ■ **Key Store Password** (this must be set if Key Store Location is set) Also, if a firewall is enabled and the Connectivity Map property, **Firewall Settings ⇒ Use Firewall**, is set to Yes, the following Environment parameters for Firewall Settings must be set with valid values: ■ **Firewall Host** ■ **Firewall Port** ■ **User** ■ **Password**	
Temp Dir	Specifies a working directory for storing intermediary results. Options include: ■ **Leave value blank**: BatchFTPOverSSL will use a default location as the temporary directory. ■ Specify a path to a local file system folder with read/write permissions. **Note –** See *Deploying an EAR File*, if you are deploying the project to another application server.	A working directory with read/write permissions, or leave blank (no value) to accept a default directory.

Connection Pool Settings (BatchFTPOverSSL Environment)

The Connection Pool Settings section of the BatchFTPOverSSL Environment properties contains the top-level parameters displayed in the following table.

TABLE 39 Environment - BatchFTPOverSSL - Connection Pool Settings

Name	Description	Required Value
Steady pool size	Specifies the minimum number of physical connections the pool keeps available at all times.	An integer indicating the maximum number of connections available at all times. A value of **0** (zero) indicates that there are no physical connections in the pool and that new connections are created as needed. The configured default is **2**.
Maximum pool size	Specifies the maximum number of physical connections the pool contains. **0 (zero)** indicates that there is no maximum.	An integer indicating the maximum number of connections allowed. A value of **0** (zero) indicates that there is no maximum. The configured default is **10**.
Max Idle Timeout In Seconds	Specifies the maximum idle timeout (in seconds). This is a hint to the server. A timer thread periodically removes unused connections. This parameter defines the interval at which this thread runs. This thread removes unused connections after the specified idle time expires. It allows the user to specify the amount of time a connection can remain idle in the pool.	An integer indicating the maximum idle timeout in seconds. When this is set to a number greater than **0**, the container removes or destroys any connections that are idle for the specified duration. A value of **0** specifies that idle connections can remain in the pool indefinitely. The configured default is **300** (5 minutes).

Connection Retry Settings (BatchFTPOverSSL Environment)

The Connection Retry Settings section of the BatchFTPOverSSL Environment Map properties contains the top-level parameters displayed in the following table.

TABLE 40 Environment - BatchFTPOverSSL - Connection Retry Settings

Name	Description	Required Value
Connection Retries	Specifies the number of retries to establish a connection upon failure to acquire a connection.	An integer indicating the maximum number of retries to establish a connection upon failure to acquire a connection. The default value is **0**.
Connection Retry Interval	Specifies the length of the pause (in milliseconds) between each reattempt to access the destination file. Used in conjunction with the Connection Retries setting.	An integer indicating length of the pause (in milliseconds). The default value is **1000** (1 second).

Configuring Batch Adapter BatchSCP Environment Properties

This topic describes the configuration properties for the BatchSCP OTD, accessed from the Environment Explorer.

The BatchSCP Adapter Environment properties include the following sections:

- "SSH Settings (BatchSCP Environment)" on page 45.
- "Firewall Settings (BatchSCP Environment)" on page 46.
- "General Settings (BatchSCP Environment)" on page 47.
- "Connection Pool Settings (BatchSCP Environment)" on page 48.
- "Connection Retry Settings (BatchSCP Environment)" on page 49.

SSH Settings (BatchSCP Environment)

The SSH Settings section of the BatchSCP Environment properties contains the top-level parameters displayed in the following table.

TABLE 41 Environment - BatchSCP - SSH Settings

Name	Description	Required Value
SSH Host	Specifies the host name or IP address of the SSH server.	The host name or IP address of the SSH server.
SSH Port	Specifies the port number of the SSH server.	The port number of the SSH server.
User	Specifies the user login name for the SSH server.	A user login name for the SSH server.

TABLE 41   Environment - BatchSCP - SSH Settings   *(Continued)*

Name	Description	Required Value
Password	Specifies a login password for the user.	A login password.
Key File	Specifies the folder (path) that holds the private key for client authentication.	The folder (path) that holds the private key for client authentication.
Key File Password	Specifies the password used to protect the key file (key pair).	The Key File password.
Server Public Key	Specifies the folder (path) that holds the public key used by the SSH server. This key is generated on the server and sent to the client through a, safe channel, and stored on the client machine for host key verification. See your specific *SSH server* documentation for more information.	The folder (path) that contains the server public key for host key verification.
Server Name For Host Key Verification	Specifies the full domain name of the SSH server used for host key verification.	The expected SSH host name when doing host key verification.
Preferred Public Key Algorithm	Specifies the preferred public key algorithm for SSH authentication. The options are DSA and RSA.	Select **DSA** or **RSA**. The configured default is **DSA**.

Firewall Settings (BatchSCP Environment)

The Firewall Settings section of the BatchSCP Environment properties contains the top-level parameters displayed in the following table.

TABLE 42   Environment - BatchSCP - Firewall Settings

Name	Description	Required Value
Firewall Host	Specifies the proxy server host name or IP address.	The proxy server host name or IP address. The configured default is **localhost**.
Firewall Port	Specifies the proxy server port number.	The proxy server port number. The configured default is **1080**.
Password	Specifies a login password for the proxy server.	A password for the proxy server login ID.

TABLE 42 Environment - BatchSCP - Firewall Settings *(Continued)*

Name	Description	Required Value
User	Specifies a user login ID for the proxy server.	A user login ID for the proxy server.

General Settings (BatchSCP Environment)

The General Settings section of the BatchSCP Environment properties contains the top-level parameters displayed in the following table.

TABLE 43 Environment - BatchSCP - General Settings

Name	Description	Required Value
Connection Mode	Specifies whether a physical connection is established when an external connection is instantiated. 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). If BatchSCP Connection Mode is set to Automatic, the following *SCP Settings* Environment parameters must be set with valid values: ■ **SSH Host** ■ **SSH Port** ■ **User**	Select **Automatic** or **Manual**. The configured default is **Automatic**.

TABLE 43   Environment - BatchSCP - General Settings   *(Continued)*

Name	Description	Required Value
	■ **Password**	
	■ **Key File** (this must be set if the BatchSCP Connectivity Map property, Authentication Type is set to HOSTBASED or PUBLICKEY)	
	■ **Key File Password** (required by the Key File property)	
	■ **Server Public Key** (this must be set if the BatchSCP Connectivity Map property, Do Host Key Verification is set to Yes)	
	■ **Server Name for Host Key Verification** (this must be set if the BatchSCP Connectivity Map property, Do Host Key Verification is set to Yes)	
	Also, if a firewall is enabled and the Connectivity Map property, **Firewall Settings ⇒ Use Firewall**, is set to Yes , the following Environment parameters must be set with valid values: ■ **Firewall Host** ■ **Firewall Port** ■ **User** ■ **Password**	

Connection Pool Settings (BatchSCP Environment)

The Connection Pool Settings section of the BatchSCP Environment properties contains the top-level parameters displayed in the following table.

TABLE 44   Environment - BatchSCP - Connection Pool Settings

Name	Description	Required Value
Steady pool size	Specifies the minimum number of physical connections the pool keeps available at all times.	An integer indicating the maximum number of connections available at all times. A value of **0** (zero) indicates that there are no physical connections in the pool and that new connections are created as needed. The configured default is **2**.

TABLE 44   Environment - BatchSCP - Connection Pool Settings   *(Continued)*

Name	Description	Required Value
Maximum pool size	Specifies the maximum number of physical connections the pool contains. 0 (zero) indicates that there is no maximum.	An integer indicating the maximum number of connections allowed. A value of **0** (zero) indicates that there is no maximum. The configured default is **10**.
Max Idle Timeout In Seconds	Specifies the maximum idle timeout (in seconds). This is a hint to the server. A timer thread periodically removes unused connections. This parameter defines the interval at which this thread runs. This thread removes unused connections after the specified idle time expires. It allows the user to specify the amount of time a connection can remain idle in the pool.	An integer indicating the maximum idle timeout in seconds. When this is set to a number greater than **0**, the container removes or destroys any connections that are idle for the specified duration. A value of **0** specifies that idle connections can remain in the pool indefinitely. The configured default is **300** (5 minutes).

Connection Retry Settings (BatchSCP Environment)

The General Settings section of the BatchSCP Environment Map properties contains the top-level parameters displayed in the following table.

TABLE 45   Environment - BatchSCP - Connection Retry Settings

Name	Description	Required Value
Connection Retries	Specifies the number of retries to establish a connection upon failure to acquire a connection.	An integer indicating the maximum number of retries to establish a connection upon failure to acquire a connection. The default value is **0**.
Connection Retry Interval	Specifies the length of the pause (in milliseconds) between each reattempt to access the destination file. Used in conjunction with the Connection Retries setting.	An integer indicating length of the pause (in milliseconds). The default value is **1000** (1 second).

Configuring Batch Adapter BatchSFTP Environment Properties

This topic describes the configuration properties for the BatchSFTP OTD, accessed from the Environment Explorer.

The BatchSFTP Adapter Environment properties include the following sections:

- "SFTP Settings (BatchSFTP Environment)" on page 50.
- "Firewall Settings (BatchSFTP Environment)" on page 51.
- "General Settings (BatchSFTP Environment)" on page 51.
- "Connection Pool Settings (BatchSFTP Environment)" on page 53.
- "Connection Retry Settings (BatchSFTP Environment)" on page 54.

SFTP Settings (BatchSFTP Environment)

The SFTP Settings section of the BatchSFTP Environment properties contains the top-level parameters displayed in the following table.

TABLE 46 Environment - BatchSFTP - SFTP Settings

Name	Description	Required Value
SSH Host	Specifies the host name or IP address of the SSH server.	The host name or IP address of the SSH server.
SSH Port	Specifies the port number of the SSH server.	The port number of the SSH server.
User ID	Specifies the user login name for the SSH server.	A user login name for the SSH server.
Password	Specifies a login password for the user.	A password.
Key File	Specifies the folder (path) that holds the private key for client authentication of type PUBLIC KEY or HOST BASED.	The folder (path) that holds the private key for client authentication.
Key File Password	Specifies the password used to protect the key file.	The Key File password.
Server Public Key	Specifies the folder (path) that holds the public key used by the SSH server. This key is generated on the server and sent to the client through a, safe channel, and stored on the client machine for host key verification.	The the folder (path) that contains the server public key for host key verification.
Server Name For Host Key Verification	Specifies the full domain name of the SSH server used for host key verification.	The expected SSH host name when doing host key verification.

TABLE 46 Environment - BatchSFTP - SFTP Settings   *(Continued)*

Name	Description	Required Value
Preferred Public Key Algorithm	Specifies the preferred public key algorithm for SSH authentication. The options are DSA and RSA	Select **DSA** or **RSA**. The configured default is **DSA**.

Firewall Settings (BatchSFTP Environment)

The Firewall Settings section of the BatchSFTP Environment properties contains the top-level parameters displayed in the following table.

TABLE 47 Environment - BatchSFTP - Firewall Settings

Name	Description	Required Value
Firewall Host	Specifies the proxy server host name or IP address.	The proxy server host name or IP address. The configured default is **localhost**.
Firewall Port	Specifies the proxy server port number.	The proxy server port number. The configured default is **1080**.
User ID	Specifies a user login ID for the proxy server.	A user login ID for the proxy server.
Password	Specifies a login password for the proxy server.	A password for the proxy server login ID.

General Settings (BatchSFTP Environment)

The General Settings section of the BatchSFTP Environment properties contains the top-level parameters displayed in the following table.

TABLE 48 Environment - BatchSFTP - General Settings

Name	Description	Required Value
Connection Mode	Specifies whether a physical connection is established when an external connection is instantiated. Options: ■ **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). If BatchSFTP Connection Mode is set to **Automatic**, the following SFTP Settings Environment parameters must be set with valid values: ■ **SSH Host** ■ **SSH Port** ■ **User** ■ **Password** ■ **Key File** (this must be set if the BatchSFTP Connectivity Map property, Authentication Type , is set to HOSTBASED or PUBLICKEY) ■ **Key File Password** (required by the Key File property)	Select **Automatic** or **Manual**. The configured default is **Automatic**.

TABLE 48 Environment - BatchSFTP - General Settings   *(Continued)*

Name	Description	Required Value
	■ **Server Public Key** (this must be set if the BatchSFTP Connectivity Map property, Do Host Key Verification is set to Yes) ■ **Server Name for Host Key Verification** (this must be set if the BatchSFTP Connectivity Map property, SFTP Settings ⇒ Do Host Key Verification is set to Yes) Also, if a firewall is enabled and the Connectivity Map property, **Firewall Settings ⇒ Use Firewall**, is set to Yes, the following Environment parameters must be set with valid values: ■ **Firewall Host** ■ **Firewall Port** ■ **User** ■ **Password**	

Connection Pool Settings (BatchSFTP Environment)

The Connection Pool Settings section of the BatchSFTP Environment properties contains the top-level parameters displayed in the following table.

TABLE 49 Environment - BatchSFTP - Connection Pool Settings

Name	Description	Required Value
Steady pool size	Specifies the minimum number of physical connections the pool keeps available at all times.	An integer indicating the maximum number of connections available at all times. A value of **0** (zero) indicates that there are no physical connections in the pool and that new connections are created as needed. The configured default is **2**.
Maximum pool size	Specifies the maximum number of physical connections the pool contains. 0 (zero) indicates that there is no maximum.	An integer indicating the maximum number of connections allowed. A value of **0** (zero) indicates that there is no maximum. The configured default is **10**.

TABLE 49   Environment - BatchSFTP - Connection Pool Settings   *(Continued)*

Name	Description	Required Value
Max Idle Timeout In Seconds	Specifies the maximum idle timeout (in seconds). This is a hint to the server. A timer thread periodically removes unused connections. This parameter defines the interval at which this thread runs. This thread removes unused connections after the specified idle time expires. It allows the user to specify the amount of time a connection can remain idle in the pool.	An integer indicating the maximum idle timeout in seconds. When this is set to a number greater than **0**, the container removes or destroys any connections that are idle for the specified duration. A value of **0** specifies that idle connections can remain in the pool indefinitely. The configured default is **300** (5 minutes).

Connection Retry Settings (BatchSFTP Environment)

The General Settings section of the BatchSFTP Environment Map properties contains the top-level parameters displayed in the following table.

TABLE 50   Environment - BatchSFTP - Connection Retry Settings

Name	Description	Required Value
Connection Retries	Specifies the number of retries to establish a connection upon failure to acquire a connection.	An integer indicating the maximum number of retries to establish a connection upon failure to acquire a connection. The default value is **0**.
Connection Retry Interval	Specifies the length of the pause (in milliseconds) between each reattempt to access the destination file. Used in conjunction with the Connection Retries setting.	An integer indicating length of the pause (in milliseconds). The default value is **1000** (1 second).

Configuring Batch Adapter BatchLocalFile Environment Properties

This topic explains the properties for the BatchLocalFile OTD, accessed from the Environment Explorer.

The BatchLocalFile properties include the following sections:

- "General Settings (BatchLocalFile Environment)" on page 55.
- "Connection Pool Settings (BatchLocalFile Environment)" on page 55.

General Settings (BatchLocalFile Environment)

The General Settings section of the BatchLocalFile Environment properties contains the top-level parameters displayed in the following table.

TABLE 51 Environment - BatchLocalFile - General Settings

Name	Description	Required Value
State Persistence Base Location	Specifies a working directory for storing intermediary results. Options: ■ Leave value blank: BatchLocalFile will use a default folder as the working directory. ■ Specify a path to a local file system folder with read/write permissions. **Note –** See *Deploying an EAR File* if you are deploying the project to another application server.	A working directory with read/write permissions, or leave blank (no value) to accept a default directory.

Connection Pool Settings (BatchLocalFile Environment)

The Connection Pool Settings section of the BatchLocalFile Environment properties contains the top-level parameters displayed in the following table.

TABLE 52 Environment - BatchLocalFile - Connection Pool Settings

Name	Description	Required Value
Steady pool size	Specifies the minimum number of physical connections the pool keeps available at all times.	An integer indicating the maximum number of connections available at all times. A value of **0** (zero) indicates that there are no physical connections in the pool and that new connections are created as needed. The configured default is **2**.

TABLE 52   Environment - BatchLocalFile - Connection Pool Settings   *(Continued)*

Name	Description	Required Value
Maximum pool size	Specifies the maximum number of physical connections the pool contains. 0 (zero) indicates that there is no maximum.	An integer indicating the maximum number of connections allowed. A value of 0 (zero) indicates that there is no maximum. The configured default is **10**.
Max Idle Timeout In Seconds	Specifies the maximum idle timeout (in seconds) This is a hint to the server. A timer thread periodically removes unused connections. This parameter defines the interval at which this thread runs. This thread removes unused connections after the specified idle time expires. It allows the user to specify the amount of time a connection can remain idle in the pool.	An integer indicating the maximum idle timeout in seconds. When this is set to a number greater than **0**, the container removes or destroys any connections that are idle for the specified duration. A value of **0** specifies that idle connections can remain in the pool indefinitely. The configured default is **300** (5 minutes).

Configuring Batch Adapter BatchRecord Environment Properties

Connection Pool Settings (BatchRecord Environment)

The Connection Pool Settings section of the BatchRecord Environment properties contains the top-level parameters displayed in the following table.

TABLE 53 Environment - BatchRecord - Connection Pool Settings

Name	Description	Required Value
Steady pool size	Specifies the minimum number of physical connections the pool keeps available at all times.	An integer indicating the maximum number of connections available at all times. A value of **0** (zero) indicates that there are no physical connections in the pool and that new connections are created as needed. The configured default is **2**.
Maximum pool size	Specifies the maximum number of physical connections the pool contains. 0 (zero) indicates that there is no maximum.	An integer indicating the maximum number of connections allowed. A value of **0** (zero) indicates that there is no maximum. The configured default is **10**.
Max Idle Timeout In Seconds	Specifies the maximum idle timeout (in seconds). This is a hint to the server. A timer thread periodically removes unused connections. This parameter defines the interval at which this thread runs. This thread removes unused connections after the specified idle time expires. It allows the user to specify the amount of time a connection can remain idle in the pool.	An integer indicating the maximum idle timeout in seconds. When this is set to a number greater than **0**, the container removes or destroys any connections that are idle for the specified duration. A value of **0** specifies that idle connections can remain in the pool indefinitely. The configured default is **300** (5 minutes).

Configuring CICS Adapter Environment Properties

The CICS Adapter configuration parameters, accessed from the Environment Explorer tree, are organized into the following sections:

- "Sun CICS Listener" on page 58.
- "CICS Gateway" on page 60.
- "CICS Client" on page 61.
- "Tracing" on page 61.
- "Connection Retry Settings" on page 63.
- "Connection Pool Settings" on page 63.

Sun CICS Listener

The Sun CICS Listener section of the CICS Environment properties contains the top-level parameters displayed in the following table.

TABLE 54   Environment Properties - Sun CICS Listener Section

Name	Description	Required Value
Host	Specifies the name of the mainframe host you are connecting.	Enter CICS. The value always defaults to **CICS** for CICS connections.
Port	Specifies the TCP/IP port where the Sun CICS Listener (and the CICS Listener) is listening. This is the port to which the CICS Adapter will connect.	The TCP/IP port to which Sun CICS Listener is listening.
Sun CICS Listener TransId	Specifies the Sun CICS Listener TransId on the mainframe host. This is the CICS Transaction that the Sun CICS Listener is installed under.	The valid TransId of the Sun CICS Listener. The default is **STCL**.
Start Type	Specifies the startup type. This can be either IC for CICS interval control, or TD for CICS transient data. This is the CICS Startup type for the program being executed.	Select **IC** or **TD** . The default value is **IC**.
Start Delay	Specifies the hours, minutes and seconds (interval of time) to delay starting the transaction program (TP) on the CICS server for the IC Start Type. This field is optional, but must specify all 6 digits if used.	A 6 digit integer. All 6 digits must be given if this is specified (for example, 123456). The default value is **000000**.
Listener Timeout	Specifies the amount of time (in milliseconds) for the Sun CICS Listener to wait for the next incoming transaction program request from the CICS Adapter.	A number indicating the Listener timeout in milliseconds (for example, 120000 milliseconds equals 2 minutes).
TP Timeout	Specifies the amount of time the CICS Adapter will wait for the Sun CICS Listener to return results for a current transaction program request.	A number indicating the TP Timeout in milliseconds (for example, 120000 milliseconds equals 2 minutes).
Polling Rate	Specifies the polling rate. This is the number of times the Sun CICS Listener queries the current TCP connection for incoming traffic before issuing an EXEC CICS DELAY for one second.	An integer indicating the Sun CICS Listener polling rate.

TABLE 54 Environment Properties - Sun CICS Listener Section *(Continued)*

Name	Description	Required Value
Transport Timeout	Specifies the timeout used by both the local and host side for send or receive.	A number indicating the Transport Timeout in milliseconds (for example, 5000 milliseconds equals 5 seconds).
COMMAREA Padding Character	Specifies the EBCDIC code for the character used by the SBYND listener to pad the COMMAREA at the CICS server when the actual length of the payload in the COMMAREA is shorter than the length given by CommAreaLength. The default value is hexadecimal 40 - EBCDIC space.	A character value coded in Hexadecimal. For example: 40 for Blanks, 00 for Low Values, FF for High Values, and so forth. The default value is **40**.
SendBufSize	Specifies the Send Buffer Size (in bytes) for the underlying socket.	A number indicating the Buffer Size in bytes (for example, 2048 bytes equals 2 KB).
ReceiveBufSize	Specifies the Receive Buffer Size (in bytes) for the underlying socket (provided as a hint).	A number indicating the Receive Buffer Size in bytes (for example, 10240 bytes equals 10 kilobytes).
NoDelay	Specifies whether the system can delay connections or requests. Generally, True (no delay) is required for high-volume or critical transactions. In cases of low-volume and noncritical transactions, you can use False. (Specifies whether to disable Nagle's Algorithm.)	Enter **True** or **False**. **True** is the default.

TABLE 54 Environment Properties - Sun CICS Listener Section   *(Continued)*

Name	Description	Required Value
KeepAlive	Specifies whether to enable the socket's SO_KEEPALIVE option when it creates a socket connection to the CICS listener. SO_KEEPALIVE maintains active connections by enabling periodic transmission of messages (if this is supported by the protocol). If the connected socket fails to respond to these messages, the connection is broken and the processes writing to that socket are notified with an ENETRESET errno. This option takes an int value in the optval argument. This is a BOOL option.	Enter **True** to enable SO_KEEPALIVE, or **False** to disable the option. **True** is the configured default.
	The socket's SO_KEEPALIVE option is used to enable pinging of the connection to the peer during connection to keep the connection "alive". This is used to prevent connections from going idle and timing out.	
	SO_KEEPALIVE periodically sends a message to the connection socket of the peer to ensure that the connection is still "alive" (active). One of three responses is expected:	
	1. The peer responds with the expected ACK. The application is not notified (since everything is OK). TCP will send another probe following another 2 hours of inactivity.	
	2. The peer responds with an RST, which tells the local TCP that the peer host has crashed and rebooted. The socket is closed.	
	3. The peer fails to return a response. The socket is closed. The purpose of this option is to detect whether the peer host has crashed.	

CICS Gateway

The CICS Gateway section of the CICS Environment properties contains the top-level parameters displayed in the following table.

TABLE 55 Environment Properties - CICS Gateway Section

Name	Description	Required Value
URL	Specifies the remote or local Gateway to which you are connecting. **Note** – This parameter requires specific JAR files when using local: as the value. The default value **local:**, does not work with CTG running on z/OS. For CTG running on a z/OS system, the URL property value must be set to localhost or to the server name.	The remote or local Gateway node name or IP address. The configured default is **local**.
Port	Specifies the TCP/IP port where CTG is running.	An number indicating the TCP/IP port.
Server	Specifies a server to use from the servers listed in the CTG configuration.	The name of a server as specified in the CTG server list. If this value is left blank, the first server specified in the list is used by default.
SSL KeyRing Class	Specifies the classname of the SSL KeyRing class.	The full classname of the SSL KeyRing class.
SSL KeyRing Password	Specifies the password for the encrypted KeyRing class.	The password for the SSL KeyRing class.

CICS Client

The CICS Gateway section of the CICS Environment properties contains the top-level parameters displayed in the following table.

TABLE 56 Environment Properties - CICS Client Section

Name	Description	Required Value
CICS UserId	Specifies the ID of the CICS user. Maximum length is eight characters.	A CICS user ID that uses eight characters or less.
CICS Password	Specifies the password for the CICS user. Maximum length is eight characters.	A password of eight characters or less.

Tracing

The Tracing section of the CICS Environment properties contains the top-level parameters displayed in the following table.

TABLE 57   Environment Properties - Tracing Section

Name	Description	Required Value
Level	CTG specific. Specifies the level of trace information recorded available. The options are: **0: None**. No CICS Java client application tracing. **1: Standard**. Only the first 128 bytes of any data block (for example the COMMAREA, or network flows) are displayed by default. This trace level is equivalent to the Gateway trace set by the ctgstart -trace option. (This can also be set using the system property **gateway.T.trace=on**). **2: Full Debug**. Traces out the whole of any data blocks by default. The trace contains more information about CICS Transaction Gateway than the standard trace level. This trace level is equivalent to the Gateway debug trace set by the ctgstart -x option. (This can also set using the system property **gateway.T=on**). **3: Exception Stacks**. Traces most Java exceptions, including exception which are expected during normal operation of the CICS Transaction Gateway. No other tracing is written. This trace level is equivalent to the Gateway stack trace set by the ctgstart -stack option. (This can also set using the system property **gateway.T.stack=on**).	An integer from **0** to **3** that indicates the specified trace information level. The configured default is **0**.
Filename	CTG specific. Specifies a file location for writing the trace output. This is an alternative to the default output on stderr. Long filenames must be surrounded by quotation marks; for example, "trace output file.log". **Note –** The filename can also be set using the system property **gateway.T.setTFile=xxx**, where **xxx** is a filename.	The output file name.
Truncation Size	CTG specific. Specifies the maximum size of any data blocks written in the trace. **Note –** The truncation size can also be set using the system property **gateway.T.setTruncationSize=xxx**, where **xxx** is a number.	A number indicating the maximum data block size. A value of **0** indicates that no data blocks will be written in the trace. No value (leaving the property blank) indicates that no truncation size is specified.

TABLE 57   Environment Properties - Tracing Section   *(Continued)*

Name	Description	Required Value
Dump Offset	CTG specific. Specifies the offset from which the display of any data blocks will start. **Note –** The dump offset can also be set using the system property **gateway.T.setDumpOffset=xxx**, where **xxx** is a number.	CTG specific. Specifies the offset from which the display of any data blocks will start.
Timing	Specifies whether or not to display time-stamps in the trace.	Select **On** or **Off**. **On** indicates that the time-stamp is displayed in the trace. The default setting is **On**. **Note –** Timing can also set using the system property **gateway.T.timing=on**.

Connection Retry Settings

The Connection Retry Settings section of the CICS Environment properties contains the top-level parameters displayed in the following table.

TABLE 58   Environment Properties - Connection Retry Settings Section

Name	Description	Required Value
Maximum Retries	Specifies the maximum number of connection retries.	A number indicating the number of times the Adapter will try to establish a connection. The configured default is **5**.
Retry Interval [ms]	Specifies the number of milliseconds to wait between connection retries.	A number indicating the time (in milliseconds) that the Adapter waits between connection attempts. The configured default is **5000** (or 5 seconds).

Connection Pool Settings

The Connection Pool Settings section of the CICS Environment properties contains the top-level parameters displayed in the following table.

TABLE 59   Environment Properties - Connection Pool Settings Section

Name	Description	Required Value
Steady Pool Size	Specifies the initial and minimum number of connections to be maintained.	A number indicating the initial and minimum number of connections to be maintained. The configured default is **2**.
MaxPool Size	Specifies the maximum size of the connection to EIS.	A number indicating the maximum size of the connection to EIS. The configured default is **10**.

Configuring COM/DCOM Adapter Environment Properties

The DCOM section of the COM/DCOM Environment property contains the top-level parameter displayed in the following table.

Name	Description	Required Value
Server	Specifies the default server used when creating an instance of a DCOM component (that is, a remote server executable). This property is not required when using an in-process component (for example, a .dll).	The name of the server on which the DCOM component is to be created. If the name is not specified, then objects are created on the local host. **Note –** This property can also be configured dynamically from the Collaboration.

Configuring HTTPS Adapter Environment Properties

Adapter External System properties must be configured from within the Environment. Until you have successfully configured all Adapters for your Java CAPS project, your project cannot be properly executed or deployed. The following list identifies the HTTPS Adapter properties. There are four Environment Configuration categories that the HTTPS Adapter implements.

Property Categories Configured in the Application Server Environment

- "HTTP Settings" on page 65.
- "Proxy Configuration" on page 66.
- "Security" on page 67.
- "Connection Pool Settings" on page 72.

HTTP Settings

HTTP Settings includes the configuration parameters listed in the following table.

Caution – Calling the clear() method in the Collaboration Editor (Java) clears all properties in this HTTP Settings section. Once the properties have been cleared, you must manually rebuild the header and payload sections of the Request message in the Transformation Designer.

TABLE 60   Environment Configuration—HTTP Settings

Name	Description	Required Value
URL	Specifies the default URL to be used for establishing an HTTP or HTTPS connection. When a URL is not assigned to the HTTP OTD, the default value is used as the URL for both the GET and POST commands. See GET and POST Methods. If "https" protocol is specified, SSL must be enabled. See the *SSL properties table*.	A valid URL. You must include the full URL. For example, http://www.sun.com or http://google.yahoo.com/bin/query If using GET functionality, you can provide the properties, using encoded query string notation. For example (all on one line). http://www.ee.cornell.edu /cgi-bin/cgiwrap/~wes/ pq?FirstName= John&LastName=Doe **Note –** For international URLs, be sure the targeting URL supports the encoding used in this property. A list of the character encoding supported by the Java 2 platform is at the Sun Web site: http://java.sun.com
Content Type	The default Content type header value to include when sending a request to the server. If no value is specified, a default value of **application/x-www-form-urlencoded** is supplied by the Adapter. **Note –** A change of the configuration value will only alter the header value, and not the actual Content type. When necessary, you can undertake any conversion or transformation of data manually.	A valid string.

TABLE 60   Environment Configuration—HTTP Settings   *(Continued)*

Name	Description	Required Value
Encoding	The default encoding used when reading or writing textual data.	A valid entry. The default is ASCII.

Proxy Configuration

The properties in this section specify the information required for the Adapter to access the external systems through a proxy server.

Use the Proxy Configuration settings in the client HTTPS Environment properties, when setting the desired URL dynamically within a Collaboration (Java) or Business Process.

Note – It is a known behavior of the Java Virtual Machine (JVM) to bypass an invalid proxy server through a local connection. As a result, you may still get a response, even if the proxy setting is invalid. This false response only happens with an HTTP connection. An HTTPS connection ensures authenticated handshaking from the proxy.

The HTTPS Adapter client bypasses the proxy server when accessing local addresses. This contrasts a web browser's behavior where all requests are sent to a proxy even if they are local.

Proxy Configuration includes the configuration parameters listed in the following table.

TABLE 61   Environment Configuration—Proxy Configuration

Name	Description	Required Value
Proxy Host	Specifies the host name of the HTTP proxy. This specifies the HTTPS proxy host to which requests to an HTTP server or reception of data from an HTTP server may be delegated to a proxy. This sets the proxy port for secured HTTP connections.	A valid HTTPS proxy host name.
Proxy Port	Specifies the port of the HTTPS proxy. This specifies the HTTPS proxy port to which requests to an HTTP server or reception of data from an HTTP server may be delegated to a proxy. This sets the proxy port for secured HTTP connections.	A valid HTTPS proxy port. The default is **8080**.

TABLE 61 Environment Configuration—Proxy Configuration   *(Continued)*

Name	Description	Required Value
Proxy Username	Specifies the user name necessary for authentication to access the proxy server.	A valid user name. **Note –** The user name is required by URLs that require HTTP basic authentication to access the site. Be sure to enter a value for this property before you enter a value for the Proxy password properties.
Proxy Password	Specifies the password required for accessing the HTTPS proxy.	The appropriate password. **Note –** Be sure to enter a value for the Proxy username properties before entering this property.

An additional task to properly configure the Proxy properties is to edit the PropertyPermission utility of the `server.policy` file in the application server:

▼ To Edit the Property Permission Utility of the server.policy File

1   Navigate to

c:\JavaCAPS`\appserver\is\lib\install\templates\`

where, *c:\JavaCAPS* is the location of your installation.

2   Add the following syntax to the `server.policy` **file:**

```
permission java.util.PropertyPermission "*", "read,write";
```

3   For the permission changes to take place, you need to create a new domain.

See *Creating and Starting the Domain* to create a new domain.

Security

The Environment Configuration Security properties are used to perform HTTP authentication and SSL connections. They include the following configuration sections:

- Table 62.
- Table 63.

Authentication

Details for the Authentication settings used for HTTP authentication are detailed in the following table.

TABLE 62 Environment Configuration — Security, Authentication

Name	Description	Required Value
HTTP Username	Specifies the user name for authenticating the web site specified by the URL.	A valid user name. **Note** – Enter a value for this property before you enter a value for the HTTP password properties.
HTTP Password	Specifies the password used for authenticating the web site specified by the URL.	A valid password. **Note** – Be sure to enter a value for the HTTP username properties before entering this property.

SSL

Details for the SSL settings used for SSL connections are detailed in the following table.

TABLE 63 Environment Configuration — Security, SSL

Name	Description	Required Value
Protocol SSL	The SSL protocol to use when establishing an SSL connection with the server. If the protocol is not set by this method, the default protocol type, TLS (Sun JSSE), is used. If an SSL connection is not required, leave the default No SSL option.	If you are using the default Sun JSSE provider, choose one of the following settings: ■ TLSv1 ■ TLS ■ SSLv2 ■ SSLv3 ■ SSL If you are running the Sun Integration Server on AIX, choose or enter one of the following settings: ■ SSL-TLS ■ TLSv1 ■ TLS ■ SSLv3 ■ SSLv2 ■ SSL For details on these settings, see the appropriate *JSSE* documentation.
JSSE Provider Class	Specifies the fully qualified name of the JSSE provider class. For more information, see the Sun Java Web site at: http://java.sun.com. It is assumed that the provider class is in the runtime classpath.	The name of a valid JSSE provider class. The default is **com.sun.net.ssl.internal.ssl. Provider** If you are running the Sun Integration Server on AIX, specify **com.ibm.jsse.IBMJSSEProvider**
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 Sun Integration Server on AIX, specify **IbmX509**.
KeyStore Type	Specifies the default KeyStore type. The keystore type is used for key/certificate management when establishing an SSL connection. If the default KeyStore type is not set by this method, the default KeyStore type, JKS, is used.	

TABLE 63 Environment Configuration — Security, SSL   *(Continued)*

Name	Description	Required Value

KeyStore	Specifies 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 the Sun Java Composite Application Platform Suite is installed and *MyDomain* is the name of your domain.

KeyStore Username	The username for accessing the keystore used for key/certificate management when establishing SSL connections. **Note –** If the keystore type is PKCS12 or JKS, the keystore username properties 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.	
KeyStore Password	Specifies 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.	
TrustStore Type	The TrustStore type of the TrustStore used for CA certificate management when establishing SSL connections. If the TrustStore type is not set by this method, the default TrustStore type, **JKS**, is used.	A valid **TrustStore** type.

TrustStore	Specifies the default TrustStore. The TrustStore is used for CA certificate management when establishing SSL connections.	A valid **TrustStore** name. There is no default value. It is recommended to use
		
		`<c:\JavaCAPS>\appserver\is`
		`\domains<MyDomain>\config\`
		`cacerts.jks`
		
		where,
		
		c:\JavaCAPS is the directory where the Sun Java Composite Application Platform Suite is installed and *MyDomain* is the name of your domain.

TABLE 63 Environment Configuration — Security, SSL   *(Continued)*

Name	Description	Required Value
TrustStore Password	Specifies 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 value.

Additional SSL Section Notes

Following are additional notes related to the properties in the SSL section.

Verify Hostname

Description

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.

- **Required Values**.
- Select **True** or **False**.

  The default is **False**.

Additional information

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`.

Connection Pool Settings

Connection Pool Settings include the configuration parameters listed in the following table.

TABLE 64   Environment Configuration — Connection Pool Settings

Name	Description	Required Value
Steady Pool Size	Specifies the minimum number of physical connections the pool should keep available at all times. 0 (zero) indicates that there should be no physical connections in the pool and the new connections should be created as needed.	A valid numeric value. The default is **1**.
Maximum Pool Size	Specifies the maximum number of physical connections the pool should keep available at all times. 0 (zero) indicates that there is no maximum.	A valid numeric value. The default is **10**.
Maximum Idle Timeout	Specifies the number of seconds that a physical connection may remain unused before it is closed. 0 (zero) indicates that there is no limit.	A valid numeric value. The default is **300**.

Configuring IMS Adapter Environment Properties

The IMS Adapter configuration parameters, accessed from the Environment Explorer tree, are organized into the following sections:

- "TCP/IP Configuration" on page 72.
- "IRM Header" on page 73.
- "Serial Mode Settings" on page 81.
- "Connection Retry Settings" on page 82.
- "Connection Pool Settings" on page 82.

TCP/IP Configuration

The TCPIP Configuration section contains information for connecting to the Portal Infranet. This section contains the top level parameters, as displayed in the following table.

TABLE 65 Environment TCP/IP Configuration Settings

Name	Description	Required Value
Server	Specifies the name of the server host. This parameter is mandatory.	The server host name.
Port	Specifies the port that IMS Connect is listening on. This parameter is mandatory.	A number indicating the port on which IMS Connect is listening. The default is **7777**.

IRM Header

The IRM (IMS Request Message) Header section contains the top-level parameters displayed in the following table.

Note – For a full description of the IRM header, see *IBM's IMS Connect Guide and Reference (SC27-0946-00).*

TABLE 66 Environment IRM Header Settings

Name	Description	Required Value
IRM_LEN	Specifies the length of the IRM structure. The user written exits minimum size is 36. HWSIMSO0 and HWSSMPL1 have a minimum IRM length of 80.	An integer indicating valid IRM structure length. The configured default is **80**.
IRM_ID	Specifies the identifier (character string) of the user exit that is driven after the complete message is received. In a program, an exit is used to move from the called routine back to the calling routine. A routine can have more than one exit point, thus allowing termination based on various conditions. The following IDs are used by the IMS Connect-supplied user message exits: ■ *IRMREQ* (for HWSIMSO0) ■ *SAMPL1* (for HWSSMPL1)	The appropriate identifier character string. The configured default is ***SAMPL1***.

TABLE 66 Environment IRM Header Settings *(Continued)*

Name	Description	Required Value
IRM_TIMER	Specifies the time delay for the receive to the Datastore after an ACK or RESUME TPIPE. One of following three predefined timer options can be selected: ■ **.25 SEC**: .25 seconds. ■ **No_Wait**: Timer is not set and no delay occurs. ■ **Block**: The receive waits indefinitely. This setting is used to support the Auto option of the asynchronous output function. OR One of the following hex values can be entered as a timer value: ■ **X01 - X19**: Range from 0.01 to 0.25 second, 0.01 second increments. ■ **X19 - X28**: Range from 0.25 to 1 second, 0.05 second increments. ■ **X28 - X63**: Range from 1 to 60 second, 1 second increments. ■ **X63 - X9E**: Range from 1 to 70 minutes, 1 minute increments.	Select one of the three predefined options or enter a valid hex value. The configured default is **.25 SEC**. **Note** – The following hex values correspond to the three predefined choices in the drop-down menu: ■ X00 = Default - .25 secs ■ XE9 = No_Wait - Does not set the timer ■ XFF = Block
IRM_SOCT	Specifies the socket connection type. ■ **Transaction**: Transaction socket. The socket connection lasts across a single transaction. ■ **Persistent**: Persistent socket. The socket connection lasts across multiple transactions. ■ **Non_Persistent**: Non-persistent socket. The socket connection lasts for a single exchange consisting of one input and one output. Do not use Non_Persistent when implementing conversational transactions because this type causes multiple connects and disconnects.	Select one of the three options. The configured default is **Persistent**. **Note** – The default for this property was changed from the previous version.

TABLE 66 Environment IRM Header Settings   *(Continued)*

Name	Description	Required Value
IRM_CLIENTID	Specifies the name of the client ID (character string) to be used by IMS Connect. IMS Adapter supports both Serial and Parallel mode. ■ Serial mode is supported by specifying a ClientID. ■ Parallel mode is supported by specifying a ClientID with an *. **Note –** In each deployment, the ClientID must be unique.	The client ID to be used by IMS Connect.
IRM_F1 (MFS MOD Names)	Specifies whether the MFS Message Output Descriptor (MOD) is returned as part of the output. ■ **MFS**: The user requests that MFS MOD name be returned. ■ **NO_MFS**: The user requests that no MFS MOD name be returned. When MFS is specified, a Request Mod Message (RMM) is returned as the first structure of the output message. This structure contains an ID of *REQMOD* followed by the MFS MOD name. For details, see *IBM's IMS Connect Guide and Reference, (SC27-0946-00)*.	Select **MFS** or **NO_MFS**. The default is **NO_MFS**.
IRM_F2 (COMMIT MODE)	Specifies the Commit Mode. ■ **COMMIT_MODE_0** - Also known as commit-then-send. ■ **COMMIT_MODE_1** - Also known as send-then-commit. For a full description of the IRM header, see IBM's *IMS Connect Guide and Reference (SC27-0946-00)*.	Select **COMMIT_MODE_0** or **COMMIT_MODE_1**. The default is **COMMIT_MODE_1**. **Note –** The default for this property was changed from the previous version.

TABLE 66 Environment IRM Header Settings   *(Continued)*

Name	Description	Required Value
IRM_F3 (Sync Level)	Specifies whether the message is to be confirmed with an ACK for Commit Mode 1 processing. For Commit Mode 0, IRM_F3 must be set to SYNC_LEVEL_CONFIRM. ■ **SYNC_LEVEL_CONFIRM**: Must be used when the IRM_F2 parameter (commit mode) is set to COMMIT_MODE_0. ■ **SYNC_LEVEL_NONE**: No Sync level.	Select **SYNC_LEVEL_CONFIRM** or **SYNC_LEVEL_NONE**. If the IRM_F2 property is set to COMMIT_MODE_0, the Sync level must be set to SYNC_LEVEL_CONFIRM. The default is **SYNC_LEVEL_NONE**. **Note** – The default for this property was changed from the previous version.

TABLE 66 Environment IRM Header Settings *(Continued)*

Name	Description	Required Value
IRM_F4 (ACK/NAK/ Response)	Specifies the ACK/NAK (positive/negative acknowledgement) response expression sent to IMS Connect and forwarded to IMS. The ACK/NAK/DEALLOCATE /RESUME [A/N/D/R] values must be sent to IMS Connect with no data element. ■ **NO_ACK**: No request for acknowledgment or deallocation. When a response mode transaction or conversational transaction is being sent to IMS Connect, IRM_F4 must be set to NO_ACK. ■ **ACK**: Positive acknowledgment, used in response to a message sent to the client where the SYNC level is set to CONFIRM (SYNC _LEVEL_CONFIRM). ■ **DEALLOCATE**: Deallocate connection. Used to terminate a conversation before the conversation is complete. ■ **NACK**: Negative acknowledgment. Used in response to a message sent to the client where the SYNC level is set to CONFIRM (SYNC _LEVEL_CONFIRM). ■ **RESUME**: Resume TPIPE. Used to request Asynchronous output data from IMS. Resume must execute on a transaction socket as COMMIT_MODE_0. ■ **SENDONLY**: Send only, used for a non-response transaction and for sending data to IMS. SENDONLY must execute as COMMIT_MODE_0.	Select one of the six options. The configured default is **NO_ACK**.

TABLE 66 Environment IRM Header Settings *(Continued)*

Name	Description	Required Value
IRM_F5 (Flow Control)	Specifies Flow Control properties. Sun recommends using the default value **No_Auto_Flow**. **Note – :** Contact Sun Support before using any value other than *No_Auto_Flow*. ■ **Client_Translation**: Translation is done by the client. ■ **Single_Message**: Returns only one message on receive following the resume TPIPE. ■ **No_Auto_Flow**: No message auto flow (see meaning for No_Auto_Flow_Out). ■ **Auto_Flow_Out**: Auto message flow. Returns all current messages, one at a time, and waits on the last receive for the next message for IRM_TIMER value. Set the IRM_TIMER high. Use this only for a dedicated output client. ■ **No_Auto_Flow_Out**: No message auto flow. Returns all current messages one at a time, and waits on the last receive for the next message for IRM_TIMER value. Set the IRM_TIMER low. Use this only for a dedicated output client. This value is similar to *Auto_Flow_Out*, as described above, except that the IRM_TIMER causes the last receive to terminate.	The recommended default setting is **No_Auto_Flow**.
IRM_TRNCOD	Specifies the default IMS transaction code.	A valid transaction code.
IRM_TRNCOD _SRC	Specifies where the transaction code is taken. ■ **CFG**: The transaction code is to be taken from the configuration file. ■ **MESSAGE**: the transaction code is the first 8 bytes of the message.	Select one of the two options. The configured default is **CFG**.
IRM_DESTID	Specifies the Datastore name (IMS destination ID). This field is required.	String-set. A Datastore name/IMS destination ID (character string).
IRM_LTERM	Specifies the IMS LTERM override name. This field can be set to a name or blank.	The appropriate LTERM name or blank.
IRM_RACF_ GRNAME	Specifies the RACF Group Name. The client must provide the RACF group name if RACF is to be used.	The appropriate RACF group name.

TABLE 66   Environment IRM Header Settings   *(Continued)*

Name	Description	Required Value
IRM_RACF_ USERID	Specifies the RACF User ID. The client must provide the RACF user ID if RACF is used.	A valid RACF user ID.
IRM_RACF_PW	Specifies the RACF PASSTICKET. The client must provide the RACF PASSTICKET, if RACF is to be used.	The appropriate RACF PASSTICKET.
IRM_HEADER_ ENCODING	Specifies the encoding of the IRM Header properties sent to IMS Connect. ■ Set the value to ISO-8859-1 if the message body is ASCII text. The IMS Connect *SAMPL1* user exit converts the data to EBCDIC. ■ Set the value to an EBCDIC code set, such as cp500, if the message is EBCDIC text or binary data. No data translation occurs.	ISO-8859-1 for ASCII transaction content, or an EBCDIC code, such as cp500, for EBCDIC transaction content.
SEND_DATA_ ENCODING	Specifies the encoding translation (if any) to apply to the message body sent to IMS Connect. ■ Set to NO TRANSLATION to send the message body to IMS Connect without translation, or when using the *SAMPL1* user exit when the IRM Headers and message body are in ASCII. ■ Set to an EBCDIC code, such as cp500, to translate the message body from ASCII to EBCDIC before sending to IMS Connect. ■ If the content is a double-byte character set such as Japanese, set to the EBCDIC code page for that language (for example, cp930 for Japanese).	Enter NO TRANSLATION or the appropriate code page as follows: ■ Enter NO TRANSLATION when using the *SAMPL1* user exit and IRM Headers and message content is in ASCII. ■ Enter an EBCDIC code, such as cp500, to translate ASCII message content to EBCDIC before sending it to IMS Connect. ■ For double-byte character sets, enter the appropriate code page for that language (for example, cp390 for Japanese).
REPLY_DATA_ ENCODING	Specifies the encoding of the message body received back from IMS Connect. Set to ISO-8859-1 if the message text is ASCII. ■ Set to an EBCDIC code, such as cp500, if the return message is EBCDIC and/or no content translation is needed. ■ If the content set is a double-byte character, such as Japanese, set the appropriate EBCDIC code page for that language (for example, cp930 for Japanese).	The appropriate code page. For ASCII transactions, enter ISO-8859-1. ■ For EBCDIC transactions, enter an EBCDIC code, such as cp500. ■ For double-byte character sets, enter the appropriate code page for that language (for example, cp390 for Japanese).

Configuring the Client ID for the IMS Adapter

The following topics describe the configuration of Client IDs for the IMS Adapter.

▼ To configure the IMS Adapter for Parallel Processing

In this mode, the IMS Adapter is configured to handle multiple requests simultaneously (parallel mode).

1 Set the Client ID in the IRM_Header section to a string which contains one or more trailing asterisks. For example, "SUN*".

The Adapter will generate the rest of the Client ID string filling it with randomly generated alphanumeric characters. The length of the Client ID is 8. If you use a static Client ID, it must be unique (across deployments) if the IMS external systems which are being used are configured to connect to the same IMS Connect.

2 Set the IRM_SOCT in the IRM_Header section to Persistent.

This allows the Adapter to retain the physical connection so that it can leverage the use of connection pooling as a resource adapter. If this is not set to Persistent and the Client ID is configured to use dynamic generation (that is, with an "*"), then a protocol error will occur.

No other IRM_SOCT type can be used in parallel mode; as noted a protocol error will result if Persistent is not used.

For the acknowledgement response expression (IRM_F4 - ACK/NAK Response), the following additional parameters must be set (in addition to the above):

a. Set the IRM_F2 (commit mode) to COMMIT_MODE_0.

b. Set the IRM_F3 (sync level) to SYNC_LEVEL_CONFIRM.

▼ To Configure the IMS Adapter for Serialized Processing

In this mode, the IMS Adapter is configured to handle one single request at a time. Multiple requests are serialized by the IMS Adapter through an internal locking mechanism.

● Set the Client ID in the IRM_Header section to a string which does NOT contain an asterisk. For example, "SUNIMS".

The Adapter will generate the rest of the Client ID string filling it with randomly generated alphanumeric characters. The length of the Client ID is 8. If you use a static Client ID, it must be unique (across deployments) if the IMS external systems which are being used are configured to connect to the same IMS Connect.

For the acknowledgement response expression (IRM_F4 - ACK/NAK Response), the following additional parameters must be set (in addition to the above):

a.   **Set the IRM_SOCT to Transaction.**

b.   **Set the IRM_F2 (commit mode) to COMMIT_MODE_0.**

c.   **Set the IRM_F3 (sync level) to SYNC_LEVEL_CONFIRM.**

Duplicate Client IDs

When sending an IMS Connect interaction on a given port, an error will occur when using a ClientID which is already in use on that port. This can happen when you are executing an interaction with a ClientID, which is the same as that used by another interaction that ended as a result of a socket timeout. If this new interaction is received by IMS Connect while IMS Connect is still waiting for a response from IMS for the original interaction that received the socket timeout, a duplicate ClientID error could occur.

This can also occur if the socket timeout being used for the original interaction is set to a value which is less than the timeout set by the IRM_TIMER or the IMS Connect default timeout (set in the HWSCFGxx member). IMS Connect is not aware that the original socket has been disconnected as a result of the socket timeout until it does a subsequent read on that socket. This means it would consider the original socket still active, even though that socket has already been disconnected from the client end. Once you get to this situation, you will receive DUPECLNT errors until the IRM_TIMER expires on the IMS Connect side.

Note – For a full discussion of Client ID and timer issues, refer to *IMS Connectivity in the On Demand Environment - A Practical Guide to IMS Connectivity* (IBM Publication SG24-6794-00).

Serial Mode Settings

The Serial Mode Settings section of the Outbound IMS Adapter Environment contains the top-level parameters displayed in the following table.

TABLE 67 Outbound IMS Adapter Environment - Serial Mode Settings

Name	Description	Required Value
Wait Timeout	When IRM_CLIENTID is static, which results in requests being serialized, multiple threads using the same Client ID will contend for a request lock. Threads contending for a request lock, being held by another thread, will wait until the pending request thread releases the lock. This parameter controls how long, in milliseconds, a request thread will wait for the lock.	An integer indicating the configured length of the time a thread will wait for the lock. The default is **6000** (milliseconds).

Connection Retry Settings

The Connection Retry Settings section of the Outbound IMS Environment contains the top-level parameters displayed in the following table.

TABLE 68 Outbound IMS Adapter Environment - Connection Retry Settings

Name	Description	Required Value
Maximum Retries	Specifies the number of retries to establish a connection with the IMS Adapter database upon a failure to acquire one.	An integer indicating the number of attempts allowed to establish a connection. The configured default is **5**.
Retry Interval [ms]	Specifies the configured length of the pause before each reattempt to access the destination file. This property is used in conjunction with the property Maximum Retries.	An integer indicating the configured length of the time (in milliseconds) before each reattempt to access the destination file. The configured default is **5000** (1 second).

Connection Pool Settings

The Connection Retry Settings section of the Outbound IMS Adapter Environment contains the top-level parameters displayed in the following table.

TABLE 69 Outbound IMS Adapter Environment - Connection Pool Settings

Name	Description	Required Value
Steady Pool Size	Specifies the initial and minimum number of connections to be maintained.	A number indicating the initial and minimum number of connections to be maintained. The configured default is **2**.
MaxPoolSize	The maximum number of physical connections the pool keeps available at all times. 0 (zero) indicates that there is no maximum.	A valid numeric value. The default is **10**.

Configuring LDAP Adapter Properties

The Adapter External System consists of the following properties categories.

-
-
-
-

Configuring Connection Section Properties

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

TABLE 70 LDAP Adapter— Connection Settings

Name	Description	Required Value
Authentication	Allows you to select 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 log-ons. ■ **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**.

TABLE 70 LDAP Adapter— Connection Settings   *(Continued)*

Name	Description	Required Value
Credentials	Allows you to enter the credentials needed when using an authentication mechanism other than anonymous log-in (authentication = **None**).	The appropriate credentials, in the form of a valid password.

InitialContext Factory	Allows you to enter the factory to be used for creating the initial context for the LDAP server. By default, the LDAP service provider provided by Sun, 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 Sun.

Principal	Allows you to specify the principal needed when using an authentication mechanism other than anonymous log-in (authentication = **None**).	The fully qualified Distinguished Name (DN) of the user, for example:
		
		`CN=Administrator,CN=Users,`
		`DC=stc,dc=com`

ProviderURL	Allows you to specify the URL of the LDAP Server.	A valid URL with the protocol as **ldap**.

Configuring the Security/SSL Section Properties

The LDAP Adapter Security/SSL Section Properties are used to set the basic security features for SSL. For more information on SSL Section properties, refer to **Additional Security/SSL Property Notes** in *Configuring Java CAPS Project Components for Communication Adapters*.

TABLE 71 LDAP Adapter— Security/SSL Settings

Name	Description	Required Value

JSSE Provider Class	Specifies the fully qualified name of the JSSE provider class. For more information, see the Sun Microsystems Java site at:
	
	http://java.sun.com/
		The name of a valid JSSE provider class; the default is:
		
		`com.sun.net.ssl.`
		`internal.ssl.Provider`
		
		If you are running the Integration Server on AIX, specify:
		
		`com.ibm.jsse.`
		`IBMJSSEProvider`

TABLE 71 LDAP Adapter— Security/SSL Settings   *(Continued)*

Name	Description	Required Value
KeyStore	Specifies 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.
KeyStore password	Specifies 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	Allows you to specify 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.
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	Allows you to specify the type of SSL connection to be used.	Select None, Enable SSL, or TLS On Demand. 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.

TABLE 71 LDAP Adapter— Security/SSL Settings   *(Continued)*

Name	Description	Required Value
SSL Protocol	The SSL protocol to use when establishing an SSL connection with the LDAP server.	Select one of the following: ■　TLS ■　TLSv1 ■　SSLv3 ■　SSLv2 ■　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 value.
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 value.
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.
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 Integration Server on AIX, specify **IbmX509**.

Configuring the Connection Retry Settings

The LDAP Adapter Connection Retry Settings properties include the following parameters:

TABLE 72   LDAP External Adapter Properties— Connection Retry Settings

Name	Description	Required Value
Maximum Retries	Maximum number of retries to establish a connection upon failure to acquire one.	There is no required value. The default value is **5**.
Retry Interval	The number of Milliseconds to wait between connection retries.	Any valid number. The default value is **10000**.

Configuring the Connection Pool Settings

The LDAP Adapter Connection Pool Settings properties include the following parameters:

TABLE 73   LDAP External Adapter Properties— Connection Pool Settings

Name	Description	Required Value
Steady Pool Size	The minimum number of connections that must be maintained in the pool.	The default value is **1**.
Maximum Pool Size	The maximum number of connections allowed in the pool. 0 (zero) indicates that there is no maximum.	The default value is **10**.
Maximum Idle Timeout	The maximum time in Seconds that a connection can remain idle in the pool. Zero indicates that there is no limit.	The default value is **300**.

Note – The current Connection Pool behavior of LDAP Adapter is as described in the example. When the

1. Steady Pool Size is **3**.

2. Outbound LDAP connections used in the CMlink is **5**.
    At runtime, **3*5=15** connections are established.

Configuring the SNA Adapter Environment Properties

This task describes how to set the environment properties of the SNA Adapter.

The Adapter Environment Configuration properties contain parameters that define how the adapter connects to and interacts with other Sun Enterprise Service Bus components within the Environment. When you create a new SNA External System, you may configure the type of External System required.

Available External System properties include:

- "SNALU62 Inbound Adapter Properties" on page 88.
- "SNALU62 Outbound Adapter Properties" on page 90.

Property Categories Configured in the Application Server Environment

SNALU62 Inbound Adapter Properties

Before deploying your adapter, you will need to set the Environment properties. The Inbound SNA Adapter includes the following configuration sections:

- SNA Settings
- General Settings
- MDB Pool Settings

SNA Settings

Details for the SNALU62 Inbound Adapter SNA Settings are listed in the table.

TABLE 74 SNALU62 Inbound Adapter—SNA Settings

Name	Description	Required Value
Host Name	Specifies the host name where the LU62 Server runs. **Note** – This parameter is only required for the Brixton LU62 server and is ignored on other platforms.	Any valid string. The default is **localhost**.
Symbolic Dest Name	Specifies the symbolic destination name associated with a side information entry loaded from the configuration file. Refer to your *SNA documentation* for more information.	Any valid string. **Note** – This parameter is case-sensitive.
Local LU Name	Specifies the local LU name defined to the SunLink LU62 server. Refer to your *SNA documentation* for more information. **Note** – This parameter is required for SunLink P2P LU6.2 9.1 and is ignored on other platforms.	Any valid string. **Note** – This parameter is case-sensitive.
Local TP Name	Specifies the local Transaction Program (TP) name that is running on the local LU. Refer to your *SNA documentation* for more information.	Any valid string. **Note** – This parameter is case-sensitive.

General Settings

Details for the SNALU62 Inbound Adapter General Settings are listed in the table.

TABLE 75 SNALU62 Inbound Adapter—General Settings

Name	Description	Required Value
Persistent Storage Location	Specifies the Persistent Location (a local folder path and name) that contains the file used to store the persistent data. The base file name will be generated according to the project, deployment, and Collaboration information.	The absolute path and name of the directory. The default is **/temp/snalu62inbound/persist**.

MDB Pool Settings

Details for the SNALU62 Inbound Adapter MDB Pool Settings are listed in the table.

TABLE 76 SNALU62 Inbound Adapter—MDB Pool Settings

Name	Description	Required Value
Steady Pool Size	Specifies the minimum number of physical connections the pool should keep available at all times. 0 (zero) indicates that there should be no physical connections in the pool and the new connections should be created as needed. If the pool size is too small, you may experience a longer connection time due to the existing number of physical connections. A connection that stays in the pool allows transactions to use it through a logical connection which is faster.	A valid numeric value. The default is **10**.
Max Pool Size	Specifies the maximum number of physical connections the pool should keep available at all times. 0 (zero) indicates that there is no maximum. The pool size you set depends on the transaction volume and response time of the application. If the pool size is too big, you may end up with too many connections to the SNA destination.	A valid numeric value. The default is **60**.
Pool Idle Timeout in Seconds	Specifies the maximum number of seconds that a physical connection may remain unused before it is closed. 0 (zero) indicates that there is no limit.	A valid numeric value. The default is **600**.

SNALU62 Outbound Adapter Properties

Before deploying your adapter, you will need to set the Environment properties. The Outbound SNA Adapter includes the following configuration sections:

- SNA Settings
- General Settings
- Connection Pool Settings

SNA Settings

Details for the SNALU62 Outbound Adapter SNA Settings are listed in the table.

TABLE 77   SNALU62 Outbound Adapter—SNA Settings

Name	Description	Required Value
Host Name	Specifies the host name where the LU62 Server runs. **Note** – This parameter is only required for the Brixton LU62 server and is ignored on other platforms.	Any valid string. The default is localhost.
Symbolic Dest Name	Specifies the symbolic destination name associated with a side information entry loaded from the configuration file. Refer to your *SNA documentation* for more information.	Any valid string. **Note** – This parameter is case-sensitive.
Local LU Name	Specifies the local LU name defined to the SunLink LU62 server. Refer to your SNA documentation for more information. **Note** – This parameter is required for SunLink P2P LU6.2 9.1 and is ignored on other platforms.	Any valid string. **Note** – This parameter is case-sensitive.
Local TP Name	Specifies the local Transaction Program (TP) name that is running on the local LU. Refer to your *SNA documentation* for more information.	Any valid string. **Note** – This parameter is case-sensitive.

General Settings

Details for the SNALU62 Outbound Adapter General Settings are listed in the table.

TABLE 78 SNALU62 Outbound Adapter—General Settings

Name	Description	Required Value
Persistent Storage Location	Specifies the Persistent Location (a local folder path and name) that contains the file used to store the persistent data. The base file name will be generated according to the project, deployment, and Collaboration information.	The absolute path and name of the directory. The default is **/temp/snalu62outbound/persist**.

Connection Pool Settings

Details for the SNALU62 Outbound Adapter Connection Pool Settings are listed in the table.

TABLE 79 SNALU62 Outbound Adapter—Connection Pool Settings

Name	Description	Required Value
Steady Pool Size	Specifies the minimum number of physical connections the pool should keep available at all times. 0 (zero) indicates that there should be no physical connections in the pool and the new connections should be created as needed. If the pool size is too small, you may experience a longer connection time due to the existing number of physical connections. A connection that stays in the pool allows transactions to use it through a logical connection which is faster.	A valid numeric value. The default is **1**.
Max Pool Size	Specifies the maximum number of physical connections the pool should keep available at all times. 0 (zero) indicates that there is no maximum. The pool size you set depends on the transaction volume and response time of the application. If the pool size is too big, you may end up with too many connections to the SNA destination.	A valid numeric value. The default is **32**.
Pool Idle Timeout in Seconds	Specifies the maximum number of seconds that a physical connection may remain unused before it is closed. 0 (zero) indicates that there is no limit.	A valid numeric value. The default is **300**.