BEA AquaLogic Service Bus™

Interoperability Solution for Tuxedo

Version: 2.6
Document Revised: January 2007

Copyright

Restricted Rights Legend

Trademarks and Service Marks

Contents

Tuxedo Transport Buffer Transformation

Tuxedo Transport Transaction Processing

CHAPTER 1

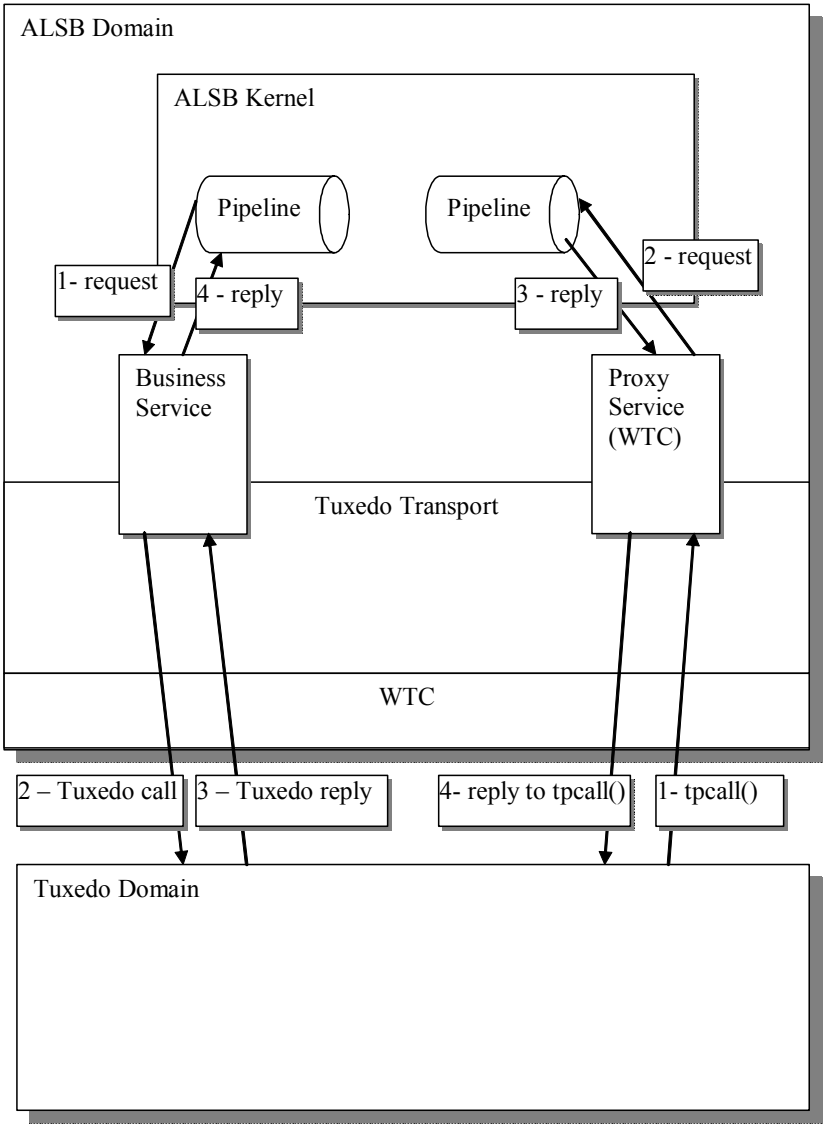
Introduction to Tuxedo Transport

BEA AquaLogic Service Bus and BEA Tuxedo can interoperate to use the services that each product offers. The Tuxedo Transport allows secure, guaranteed, high performance, bi-directional access to Tuxedo domain from BEA AquaLogic Service Bus. The Tuxedo Transport allows Tuxedo domains to call services as well as have services called in a Tuxedo domain.

- When AquaLogic Service Bus uses services offered by BEA Tuxedo, the Tuxedo Transport facilitates access to those Tuxedo services. The term *outbound* refers to Business Services scenario.
- When BEA Tuxedo uses services offered by AquaLogic Service Bus, BEA Tuxedo services can call AquaLogic Service Bus services as though it were another BEA Tuxedo application. The term *inbound* refers to this Proxy Service scenario.

You can configure the Tuxedo Transport using the AquaLogic Service Bus Console. Specific parameters provide definitions for both Proxy and Business Services. A basic WTC configuration with one local access point and one remote access point is required to enable configuration of the Tuxedo Transport. Support for transactional and security contexts are available as well.

The following diagram summarizes this message handling process.


ALSB Domain
ALSB Kernel
Pipeline
Pipeline
1- request
4 - reply
3 - reply
2 - request
Business Service
Proxy Service (WTC)
Tuxedo Transport
WTC
2 – Tuxedo call
3 – Tuxedo reply
4- reply to tpcall()
1- tpcall()
Tuxedo Domain

Capabilities of the Tuxedo Transport

The following capabilities are available in the native Tuxedo Transport in BEA AquaLogic Service Bus.

- First class tier transport

  The native Tuxedo Transport is fully integrated into the BEA AquaLogic Service Bus configuration, management, and monitoring console. You configure the Tuxedo Transport using the BEA AquaLogic Service Bus console. You can configure, manage, and monitor both Tuxedo Proxy Services and Tuxedo Businesses Services.

- Bi-directional access

  AquaLogic Service Bus is an intermediary between a SOAP, JMS, or other services and Tuxedo. The Tuxedo Transport enables access to Tuxedo ATMI services as Business Services to AquaLogic Service Bus and allows AquaLogic Service Bus Business Services to transparently be seen by Tuxedo as another ATMI service.

- Buffer transformation

  Transformation facilities exist to transform XML messages to Tuxedo buffer types and Tuxedo buffers types to XML. All standard Tuxedo buffer types are supported and transformation is automatic and transparent. For more information about buffer transformation, refer to "Tuxedo Transport Buffer Transformation" on page 5-1.

- Transactional integrity

  Tuxedo Transport provides transactional integrity for inbound and outbound messages. You can call Tuxedo services in the context of a global transaction allowing Exactly Once quality of service (QoS). A Tuxedo application can start a transaction, call an AquaLogic Service Bus-based service and the XA transaction context is carried through to AquaLogic Service Bus, through the pipeline, and finally to the destination transport. For more information about transactions, refer to "Tuxedo Transport Transaction Processing" on page 6-1.

- Security propagation

  The security context established at the beginning of the message flow, from either a Tuxedo client or an AquaLogic Service Bus client is propagated to the other system. This means that an incoming SOAP over HTTP request to AquaLogic Service Bus that requires authentication is authenticated by AquaLogic Service Bus. As with transactions, this support is fully bi-directional, so a client authenticated to Tuxedo can make requests to AquaLogic Service Bus services without requiring authentication a second time.

- Encrypted network links

  You can encrypt the connections between AquaLogic Service Bus and Tuxedo through WTC configuration to ensure the security and privacy of communications between the two systems.

- Load balancing

  A single network connection is the only requirement to connect AquaLogic Service Bus to a Tuxedo domain. However, it may be necessary to support multiple connections in case of a machine or network failure. You can make multiple connections to a single domain or multiple domains for purposes of load balancing.

Refer to the following sections for detailed information on configurations for the following scenarios:

- Configuring WebLogic Tuxedo Connector for Tuxedo Transport
- Using BEA Tuxedo Services from AquaLogic Service Bus
- Using AquaLogic Service Bus from BEA Tuxedo

You can use the following related documentation to learn more about this environment:

- For information on setting up your AquaLogic Service Bus environment, see the *Using the AquaLogic Service Bus Console*.
- For more information on setting up your BEA Tuxedo environment, see *Setting Up a Tuxedo Application* in the BEA Tuxedo documentation.
- For information about the WebLogic Tuxedo Connector, see WebLogic Tuxedo Connector in the WebLogic Server documentation.

CHAPTER 2

Configuring WebLogic Tuxedo Connector for Tuxedo Transport

AquaLogic Service Bus Tuxedo Transport enables access to BEA Tuxedo services using WebLogic Tuxedo Connector (WTC). To use the Tuxedo Transport, you must configure a basic WTC server including one local access point and one remote access point.

The following sections describe how to configure WTC:

Before You Begin

Gather the following information about the Tuxedo application that AquaLogic Service Bus will use:

- ID of the Tuxedo local access point.
- Network address of the Tuxedo local access point.
- Name of the exported Tuxedo service.
- Whether the service needs XML-to-FML and FML-to-XML conversion or View-to-XML or XML-to-View conversion.

  The example described in the following sections assumes the use of FML/FML32 buffer types.

- ID of the access point that the Tuxedo domain gateway will use to refer to this WebLogic Tuxedo Connector instance.
- Network address that the Tuxedo domain gateway has defined for this WebLogic Tuxedo Connector Local Access Point.

Configuring WebLogic Tuxedo Connector

You configure WebLogic Tuxedo Connector using the WebLogic Server Administration Console. For additional information about the WebLogic Tuxedo Connector, see WebLogic Tuxedo Connector in the WebLogic Server documentation.

Log in to the WebLogic Server Administration Console. Perform the configuration steps in the order presented, using the instructions in the following sections.

- Create a New WTC Server
- Create a Local Access Point
- Create a Remote Access Point
- Activate Changes
- Create a User ID

Create a New WTC Server

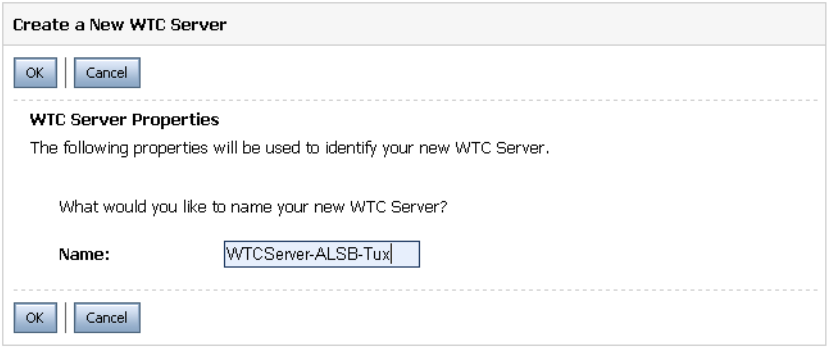
Follow these steps:

1. Click **WTC Servers** under the Interoperability tab.
2. Click **Lock & Edit;** this allows you to make changes. A display similar to Figure 2-1 appears:

Figure 2-1 WTC Server Display

3. Click **New** to add the new WTC server. A display similar to Figure 2-2 appears:

Figure 2-2 New WTC Server Data Entry Display



4. Enter a name for the WTC server and click **OK**.

   A message at the top of the page indicates that the server was added correctly.

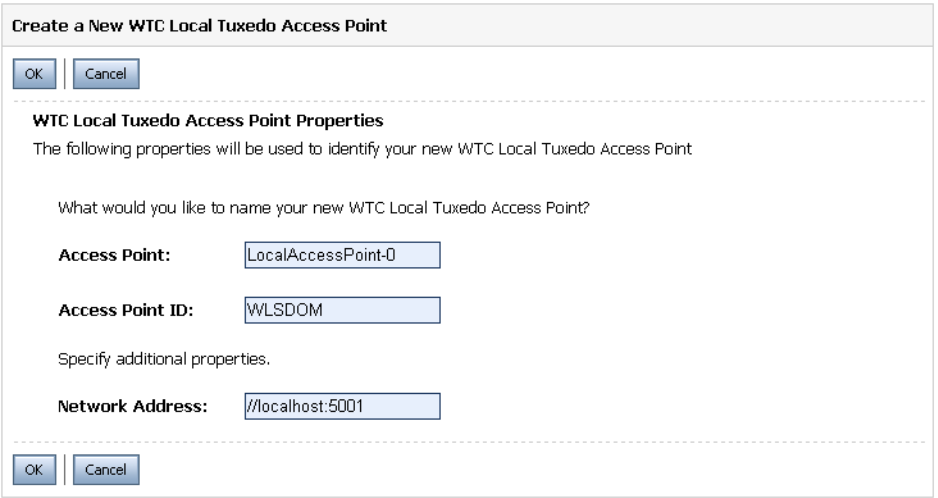
5. Click the newly created WTC server to display its settings.

Create a Local Access Point

Follow these steps:

1. Click **Local APs** on the Configuration tab.
2. Click **New** to create a new WTC Local Access Point. A display similar to Figure 2-3 appears:

Figure 2-3 New Local Access Point Data Entry Display



3. Enter the following values:

   Access Point – A name for this access point.

   Access Point ID – The unique name WebLogic Server will use to refer to the access point. This value must match the Remote Access Point ID that the Tuxedo domain gateway has been configured to use for this WTC instance.

   Network Address – This value must match the remote network address that the Tuxedo domain gateway has been configured to use for this WTC instance.

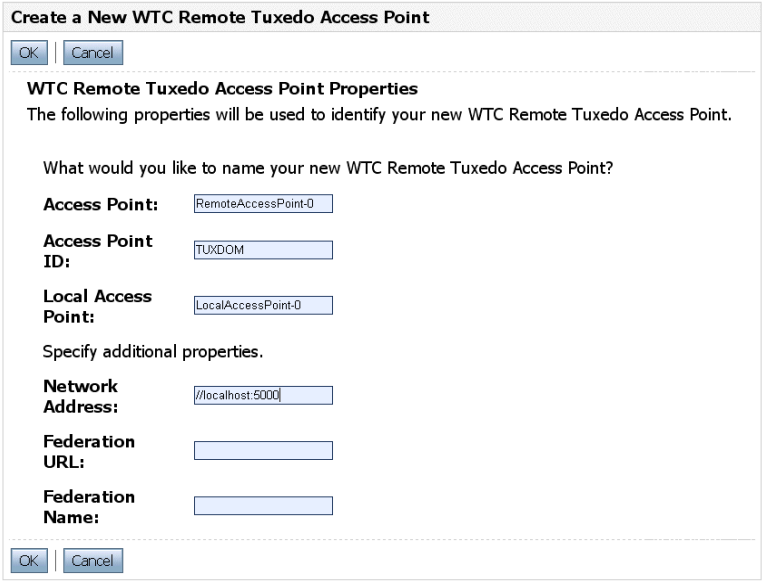
4. Click **OK**.

Create a Remote Access Point

Follow these steps:

1. Click **Remote APs** on the Configuration tab.
2. Click **New** to create a new WTC Remote Access Point. A display similar to Figure 2-4 appears:

Figure 2-4 New Remote Access Point Data Entry Display



3. Enter the following values:

   Access Point – A name for this access point.

   Access Point ID – The name WebLogic Server will use to refer to the access point. This value must match the Local Access Point ID that the Tuxedo domain gateway has been configured to use for this WTC instance.

   Local Access Point – The name of WTC Local Access Point.

   Network Address – This value must match the local network address that the Tuxedo domain gateway has been configured to use for this WTC instance.

4. Click **OK**.

Activate Changes

To activate the changes you made, click **Activate Changes** on the WebLogic Server Administration Console.

Create a User ID

In order to allow a remote Tuxedo domain to perform inbound calls, the connection should be identified.

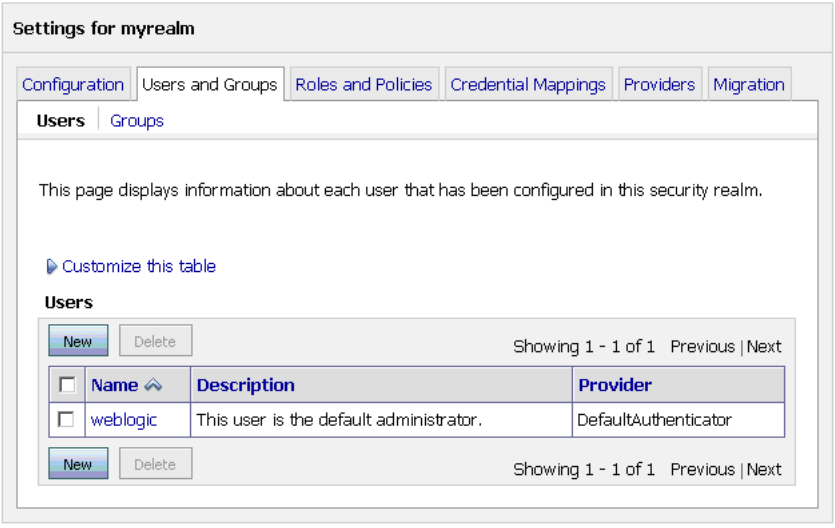
If a remote Tuxedo domain calls a service advertised by AquaLogic Service Bus, running the client as is will result in an error “8” (TPEPERM: permission denied) being returned. To allow inbound calls, you should create a user in the default security realm with the same name as the remote Tuxedo domain Access Point ID.

Note: You do not need to be in session to create a user.

Perform the following steps to add the user TUXDOM:

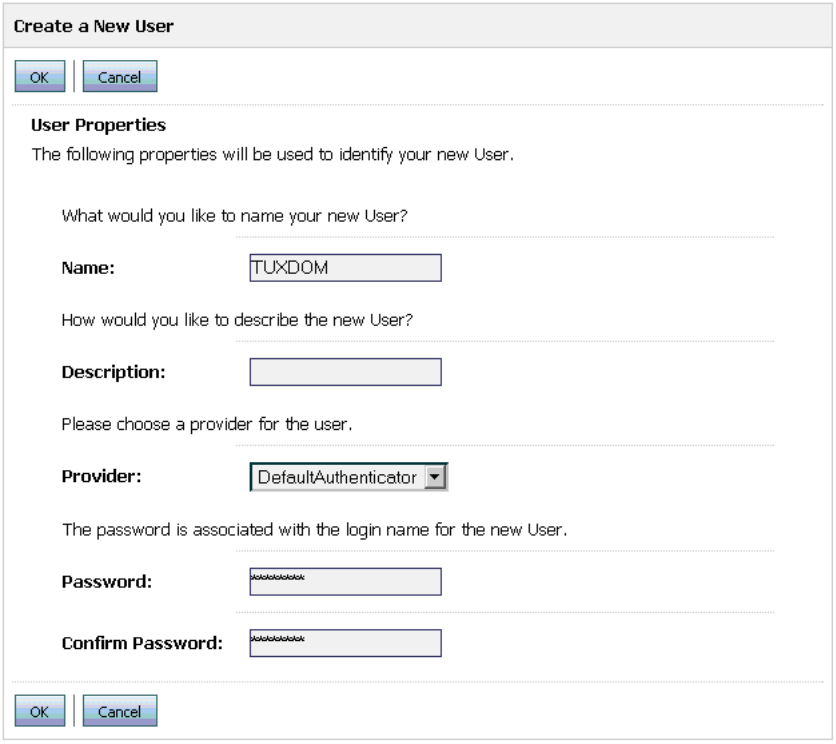
1. Click **Security Realms** under the your application’s **Security Settings** tab.
2. Click **myrealm** (or your default realm if your security settings have been customized).
3. Select the **Users and Groups** tab.

Figure 2-5 Defining a User



4. Click **New**. A page similar to the following figure displays:

Figure 2-6 Create a New User Page



5. Enter `TUXDOM` in the **Name** field.
6. Enter a password and confirm it. It is not necessary to remember the password value.
7. Click **OK**. The user has been created and incoming requests can be processed.

CHAPTER 3

Using BEA Tuxedo Services from AquaLogic Service Bus

The following sections describe how to use BEA Tuxedo services from AquaLogic Service Bus, which is referred to as outbound:

- Before You Begin
- Configuring a New Business Service
- Load Balancing and Failover
- Testing Your Configuration

Before You Begin

You must have a WTC server configured with one local access point and one remote access point. This provides access from AquaLogic Service Bus to Tuxedo services. For information on configuring WTC, refer to "Configuring WebLogic Tuxedo Connector for Tuxedo Transport" on page 2-1.

Configuring a New Business Service

To utilize the Tuxedo service from AquaLogic Service Bus, you must configure a new Business Service in the AquaLogic Service Bus Console. For more information about Business Services, see Business Services in the *Using the AquaLogic Service Bus Console*.

Log in to the AquaLogic Service Bus Console. Perform the configuration steps in the order presented, using the instructions in the following sections.

Add a New Project

Follow these steps:

1. Click **Create** to start a new console session.

   You must be in a session to edit resources.

2. Click **Project Explorer**.

3. Enter a name for the new project and click **Add Project**.

   A message at the top of the page indicates that the server was added correctly.

Add a Business Service

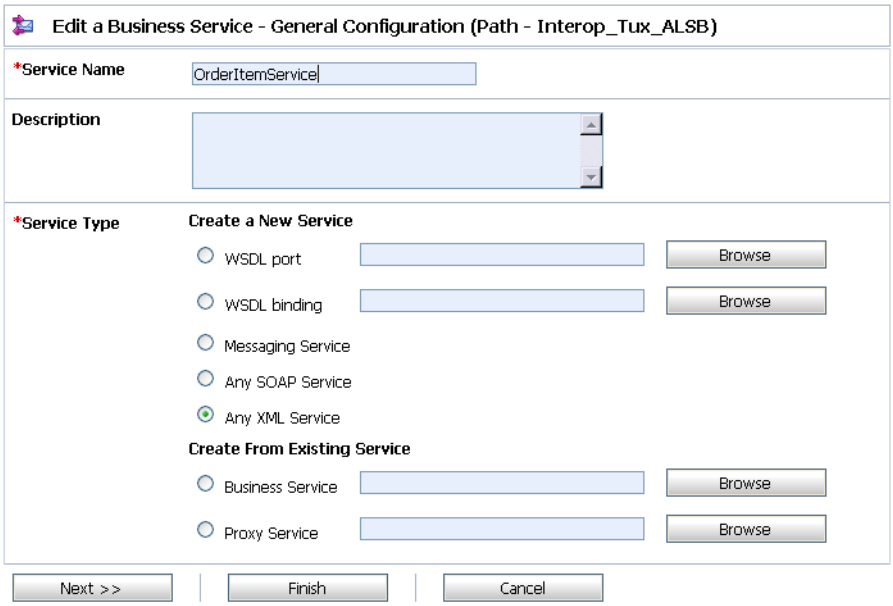
Follow these steps:

1. Click the newly created project.

2. In the **Resources** area **Create Resource** drop down menu, select **Business Service**.

   The **Edit a Business Service – General Configuration** page displays, as shown in Figure 3-1.

Figure 3-1 New Business Service Page 1



3. Enter the following values:

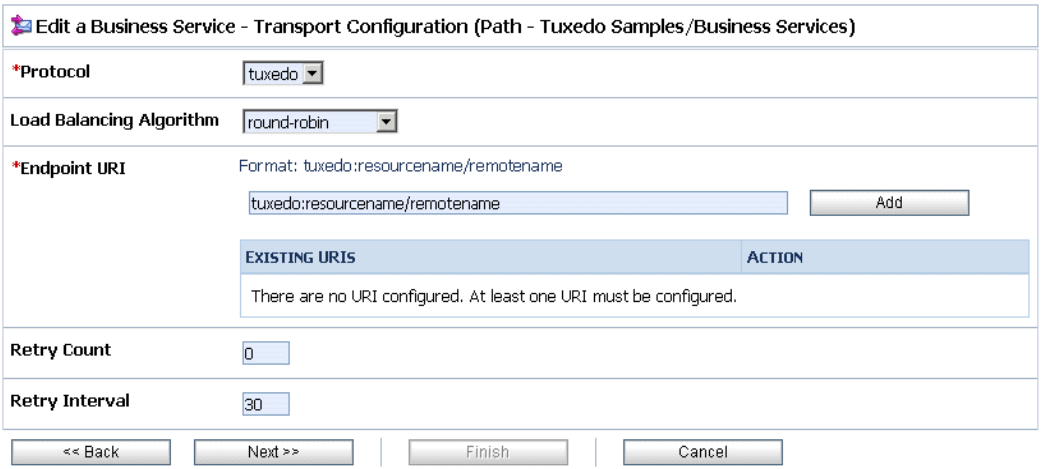
   Service Name – The name of the service

   Service Type – Select **Any XML Service** (the default).

   Note: Tuxedo Transport only supports **Any XML Service** and **Messaging Service** as Service Types.

   Click **Next** to display the **Edit a Business Service – Transport Configuration** page as shown in Figure 3-2.

Figure 3-2 New Business Service Page 2



4. Enter the following values:

   Protocol – Select **tuxedo.**

   Load Balancing Algorithm – Leave the default as is, or select another algorithm.

   Endpoint URI - Specify one or more endpoint URIs that correspond to the endpoint URI on the server where the service was deployed. The URI format is `tuxedo:resourcename[/remotename]`. The URI `resourcename` corresponds to a WTC Import name and the `remotename` corresponds to the service name exported by the remote Tuxedo domain.The URI `resourcename` is required, and the `remotename` is optional.

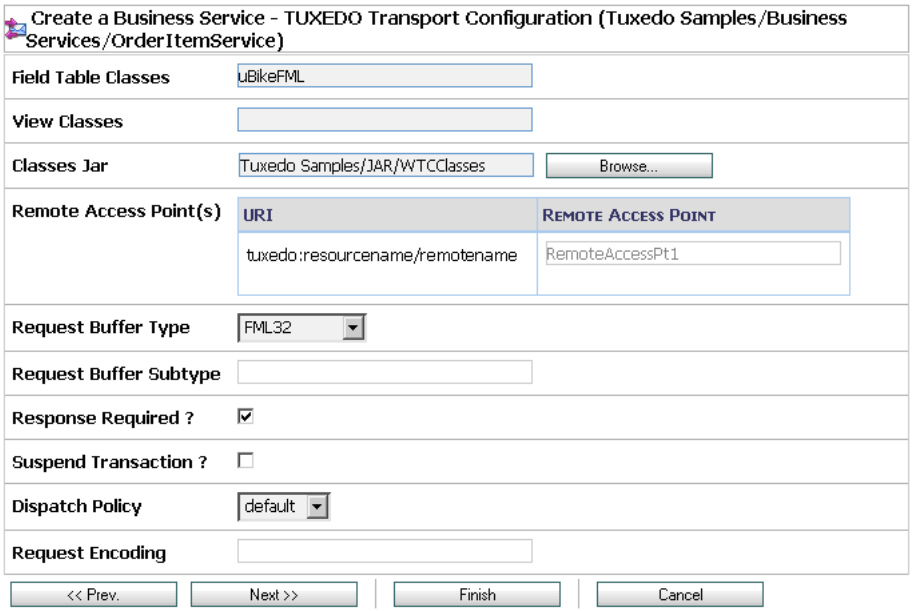
   If more than one URI is specified, you must have unique resource names for the endpoints. If no remote name is specified, its value is the value of the resource name. If no remote name is entered or if remote and resource name are the same, only one URI is allowed. In this case resource name and remote name will have the same value. This allows already defined WTC Imports to make use of WTC load-balancing and failover.

   Note: If you configure two identical URIs, an error displays notifying you that the service name already exists.

   The Tuxedo Transport uses the resource name and remote name from the URI to dynamically create a WTC Import.

5. Click **Next** to continue.

Figure 3-3 New Business Service - Tuxedo Transport Configuration



6. Enter the following values:

   Field Table Classes – Optional field. Enter the name of the class or classes describing the FML/FML32 buffer received. These are used for the FML/FML32-to-XML conversion routines to map field names to element names. This is a space separated list of fully qualified class names.

   View Classes – Optional field. Enter the name of the class or classes describing the VIEW/VIEW32 buffer received or sent. These are used for the VIEW-to-XML or VIEW32-to-XML conversion routines to map field names to element names. This is a space separated list of fully qualified class names.

   Classes Jar – Select a Generic Resource that contains a JAR file with the FML/FML32 or VIEW/VIEW32 classes necessary for this endpoint operation.

   Remote Access Point – Select a remote access point from the drop down list that is associated with the Import. The drop down list contain remote access points configured in WTC. A Business Service cannot be created if there is no associated remote access point.

   If more than one URI has been specified, there will be one remote access point field per URI and the URI displays for informative purposes. If more than one URI exists, each

requires a different remote access point. If the URI specified already corresponds to an existing WTC resource, the corresponding remote access point displays, but cannot be modified.

Request Buffer Type – Select from the drop down list the type of buffer that the remote Tuxedo service will receive.

Request Buffer Subtype – This field is enabled if the previous Request Buffer Type value is VIEW or VIEW32. Enter the buffer subtype with which to associate the request buffer.

Response Required? – Select the checkbox to indicate a bidirectional call. If not checked, the underlying `tpcall` is invoked with `TPNOREPLY` flag, and a null response is posted asynchronously.

Suspend Transaction – Select the checkbox to suspend the transaction, if it exists. This is useful when the remote service does not support transactions.

Dispatch Policy – Select a WLS work manager from the drop down list, if available. The default work manager is presented and used if no other WLS work manager exists. This work manager is used to asynchronously post a null reply in the case of a one-way invocation.

Request Encoding – This field is enabled if the **Request Buffer Type** is MBSTRING. Select the checkbox for **Request Encoding** to override the encoding of an MBSTRING buffer when sent to the Tuxedo service.

7. Click **Finish**.
8. At the **Summary** page, click **Save**.

Load Balancing and Failover

When specifying a Business Service and defining the endpoint URIs, entering a remote name that is different from the resource name allows you to use the AquaLogic Service Bus load balancing and failover capabilities. In this case, you can define multiple service names and associate them to a service that is replicated across multiple remote domains. The resource name must be unique, but remote names do not have the same restriction.

Testing Your Configuration

Now that you have configured AquaLogic Service Bus to work with BEA Tuxedo, you can test the application. One way to test the configuration is to click on the Test icon from the AquaLogic Service Bus Console.

The following list of tasks summarizes the process of testing outbound usage of BEA Tuxedo by AquaLogic Service Bus.

1. Build and start the Tuxedo servers.
2. Set up a Tuxedo service to call the AquaLogic Service Bus proxy.
3. In **Project Explorer**, select the **Business Service** you want to test.
4. Click the **Test Console** icon to launch the Test Console.
5. Enter a payload in the Test Console.
6. Click **Execute** to begin testing the Proxy Service.

   A response page displays the results of the service request.

CHAPTER 4

Using AquaLogic Service Bus from BEA Tuxedo

The following sections describe how to use AquaLogic Service Bus services from BEA Tuxedo, which is an inbound example:

- Before You Begin
- Adding and Configuring a Proxy Service
- Testing Your Configuration

Before You Begin

You must have a WTC server configured with one local access point and one remote access point. This provides access from AquaLogic Service Bus to Tuxedo services. For information on configuring WTC, refer to "Configuring WebLogic Tuxedo Connector for Tuxedo Transport" on page 2-1.

Adding and Configuring a Proxy Service

To utilize theAquaLogic Service Bus service from Tuxedo, you must configure a new proxy service using the AquaLogic Service Bus Console. For more information about proxy services, see Proxy Services in the *Using the AquaLogic Service Bus Console*.

Log in to the AquaLogic Service Bus Console and perform these steps in the order presented.

To complete this configuration, you will perform the tasks described in the following sections:

- Add a New Project

- Add a Proxy Service
- Configure the Proxy Service

Add a New Project

Follow these steps:

1. Click **Create** to start a new console session.

   You must be in a session to edit resources.

2. Click **Project Explorer**.

3. Enter a name for the new project and click **Add Project**.

   A message at the top of the page indicates that the server was added correctly.

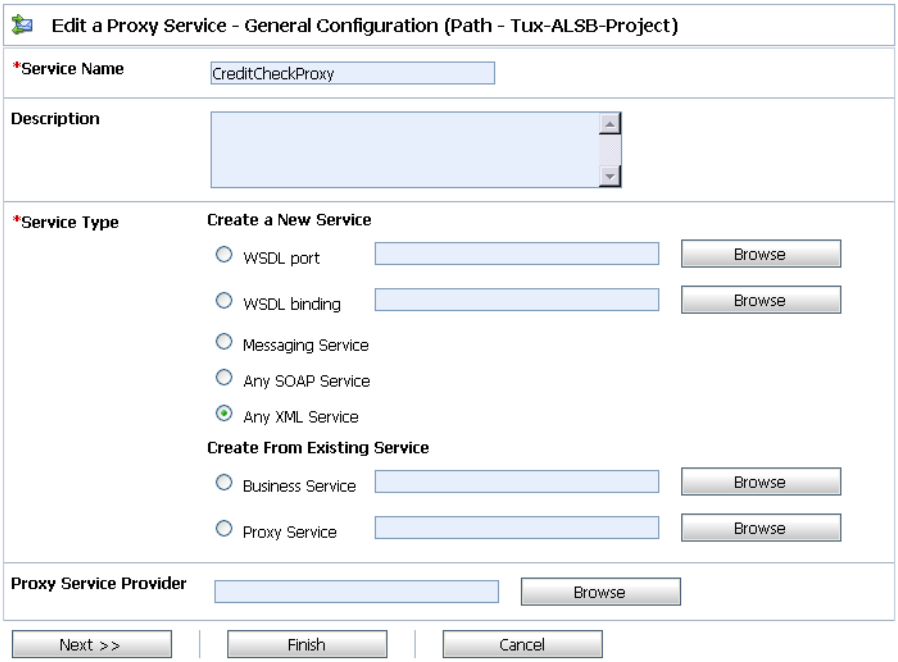
Add a Proxy Service

Follow these steps:

1. In the **Resources** area **Create Resource** drop down menu, select **Proxy Service**.

   The **Edit a Proxy Service – General Configuration** page displays, as shown in Figure 4-1.

Figure 4-1 New Proxy Service Data Entry Page 1 - Inbound



2. Enter the following values:

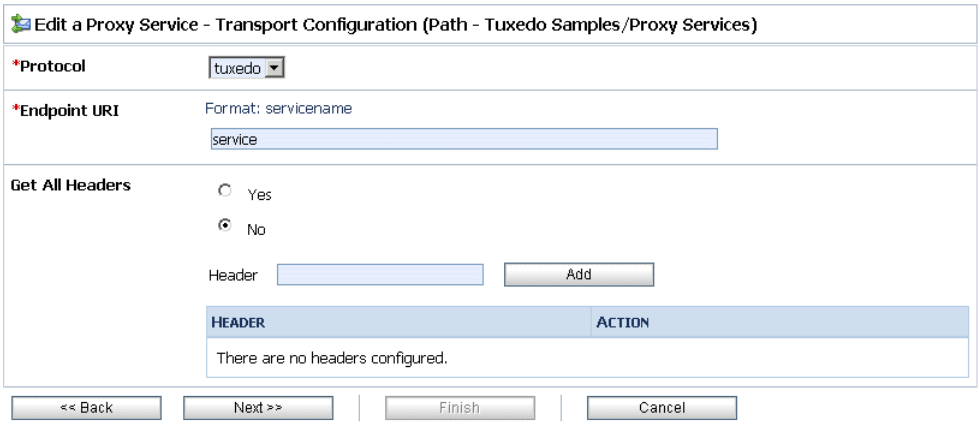
   Service Name – The name of the service

   Service Type – Select **Any XML Service** (the default).

   Note: Tuxedo Transport only supports **Any XML Service** and **Messaging Service** as Service Types.

   Click **Next** to display the **Edit a Proxy Service – Transport Configuration** page, as shown in Figure 4-2.

Figure 4-2 New Proxy Service Data Entry Page 2 - Inbound



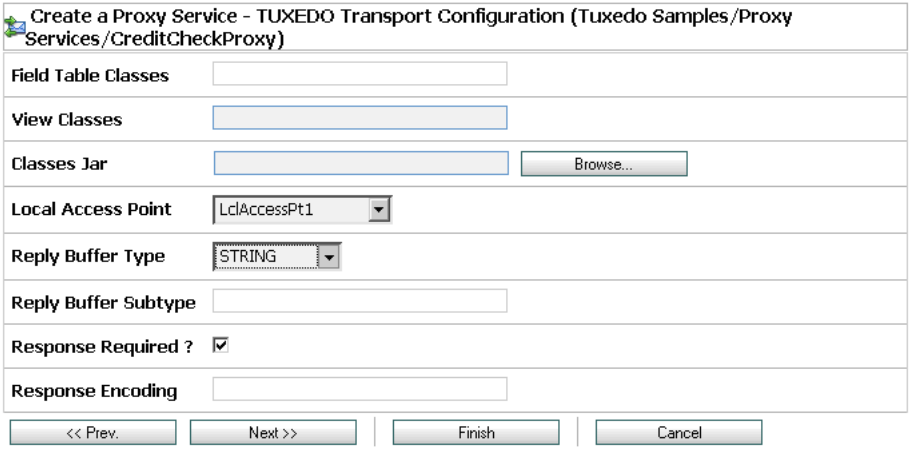
3. Enter the following required values:

   Protocol – Select **tuxedo**.

   Endpoint URI - Enter a service name that corresponds to the endpoint URI on the server where the service was deployed.

4. Click **Next** to continue.

Figure 4-3 New Proxy Service - Tuxedo Transport Configuration



5. Enter the following values:

   Field Table Classes – Optional field. Enter the name of the class or classes describing the FML/FML32 buffer received. These are used for the FML/FML32-to-XML conversion routines to map field names to element names. This is a space separated list of fully qualified class names.

   View Classes – Optional field. Enter the name of the class or classes describing the VIEW/VIEW32 buffer received or sent. These are used for the VIEW-to-XML or VIEW32-to-XML conversion routines to map field names to element names. This is a space separated list of fully qualified class names.

   Note: `X_C_TYPE` and `X_COMMON` Tuxedo buffer types are handles in the same manner as VIEW/VIEW32 buffers.

   Note: If an incoming request contains a VIEW, then the corresponding VIEW class should be specified in the AquaLogic Service Bus `CLASSPATH`.

   Classes Jar – Select a JAR Resource that contains a JAR file with the FML/FML32 or VIEW/VIEW32 classes necessary for this endpoint operation

   Local Access Point – Select a local access point from the drop down list that is associated with the Export. The drop down list contain local access points configured in WTC. A Proxy Service cannot be created if there is not an associated local access point.

   Reply Buffer Type – Select from the drop down list the type of buffer that the remote Tuxedo client will receive. This field is enabled if the **Response Required** field is checked.

   Reply Buffer Subtype – This field is enabled if the previous Request Buffer Type value is VIEW or VIEW32. Enter the buffer subtype with which to associate the reply buffer. This field is enabled if the **Response Required** field is checked.

   Response Required? – Select the checkbox if this service is expected to send a response. The default status is checked, and unchecked if the service type is **Messaging Service** and the response message type is **None**. In this case, the field is not enabled.

   Response Encoding – This field is enabled if the **Reply Buffer Type** is MBSTRING. Select the checkbox for **Response Encoding** to override the encoding of an MBSTRING buffer when sent to the Tuxedo client.

6. Click **Finish**.

7. At the **Summary** page, click **Save**.

Configure the Proxy Service

AquaLogic Service Bus Message Flows define the implementation of proxy services. Message flows can include zero or more pipeline pairs: request and response pipelines for the proxy service (or for the operations on the service); and error handler pipelines that can be defined for stages, pipelines, and proxy services. Pipelines can include one or more stages, which in turn include actions.

For more information about configuring AquaLogic Service Bus Proxy Services, see *Using the AquaLogic Service Bus Console* and Modeling Message Flow in AquaLogic Service Bus in the *AquaLogic Service Bus User Guide*.

The following sections provide one example of how to change the routing behavior of the proxy service and edit this message flow to:

- Add a route node
- Configure an action to route the Proxy Service to the Business Service resource that you created previously

Follow these steps:

1. In the AquaLogic Service Bus Console navigation panel, select Resource Browser from the list of available choices, if it is not already selected.

   The Resource Browser pane is opened in the navigation panel and the Summary of Proxy Services project page is displayed in the console.

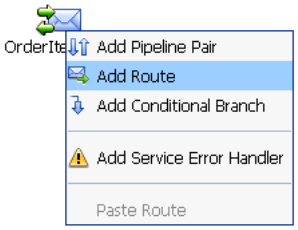
2. In Options, click the **Message Flow** icon . A display similar to Figure 4-4 appears:

Figure 4-4 Message Flow Default Display

   The Edit Message Flow page for the proxy service you created previously is displayed. This page displays the default message flow configuration. The default configuration consists of a start node. This is the minimum configuration of a message flow. The behavior of the message flow is sequential.

3. Click the Start Node, for example CreditCheckProxy Node. From the popup menu select the **Add Route** link, as shown in Figure 4-5.

Figure 4-5 Convert to Route Node Display

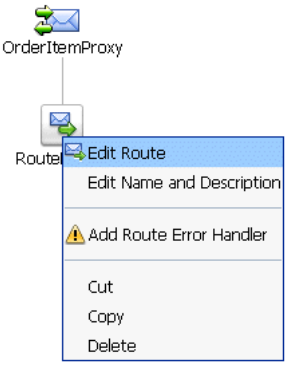


4. In the configuration dialog, name the route node as desired and click **Save**.

   In the message flow, the name of the node changes to display the route node name.

5. Click the route node and from the pop up menu select **Edit Route**, as shown in Figure 4-6:

Figure 4-6 Edit Route Node Display

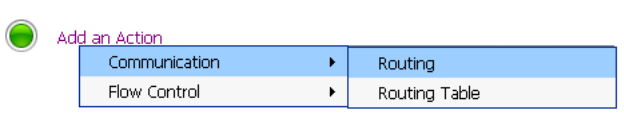


   The Edit Stage Configuration page is displayed. The page contains a single link, Add an Action.

   A stage is an element of a pipeline and it is a container for actions defined in a pipeline. Actions are the elements of a pipeline stage that define the handling of messages as they flow through a proxy service.

6. Click the **Add an Action** link, then select **Communication>Routing** from the popup menu, as shown in Figure 4-7:

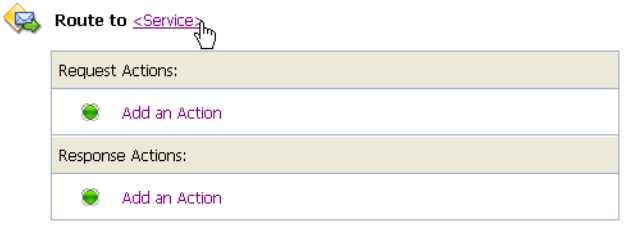
Figure 4-7 Message Flow Routing Display



The Edit Stage Configuration page changes to display the contents of the action. The contents of the action are defined by the type of node we created—a route node.

7. In Route to <Service>, click **<Service>**, as shown in Figure 4-8:

Figure 4-8 Route to Service Display



The Service Browser displays the names of the Proxy Service and Business Service that you created.

8. Select the Business Service that you want to expose to Tuxedo.
9. Click **Submit**.

   The display updates to show routing to the Business Service.

The configuration is completed and ready to test.

Testing Your Configuration

Now that you have configured BEA Tuxedo to work with AquaLogic Service Bus, you can perform a test to verify that it is working correctly. If you are using XML-to-FML32 and FML32-to-XML conversions, you can test this configuration using the "ud32" Tuxedo client program that is included with BEA Tuxedo. (If you are using FML conversions, you can use the "ud" client.) ud32 reads input consisting of text representation of FML buffers. For more information, see the information on the ud and ud32 commands in the *Tuxedo Command Reference*.

If you are not using XML-to-FML and FML-to-XML conversions, you must develop a test client program in Tuxedo to test this configuration. To find information on this task, refer to the BEA Tuxedo documentation.

CHAPTER 5

Tuxedo Transport Buffer Transformation

BEA AquaLogic Service Bus and BEA Tuxedo can interoperate to use the services that each product offers. Using the services can include buffer transformation. The AquaLogic Service Bus service type and the Tuxedo buffer type determine how transformation occurs.

Tuxedo Transport supports **Any XML Service** and **Messaging Service** service types in AquaLogic Service Bus.

- **Any XML Services (Non SOAP)**—The messages to XML-based services are XML, but can be of any Tuxedo buffer type allowed by the service configuration.
- **Messaging Services**—Messaging services are those that can receive messages of one data type and respond with messages of a different data type. The supported data types include XML, MFL, text, and untyped binary.

For more information about service types and buffer transformation in AquaLogic Service Bus, see Modeling Message Flow in AquaLogic Service Bus in the *AquaLogic Service Bus User Guide*.

The following sections explain how the Tuxedo Transport handles buffer transformation.

Any XML Service Type

The following table shows the behavior of the Tuxedo Transport depending on the Tuxedo buffer type when the service type is **Any XML Service**.

Note: For **Any XML Service**, the payload must be a well-formed XML document.

Table 5-1 Buffer Transformation for Any XML Service

Tuxedo Buffer Type	Tuxedo Transport Behavior...
STRING	Passes the buffer as is.
CARRAY X_OCTET	Passes the buffer as is.
FML/FML32	Converts the buffer to/from XML using the configured field classes.
XML	Passes the buffer as is. **Note:** The Tuxedo application *should not* send NULL byte when the Tuxedo buffer is XML.
VIEW/VIEW32 X_C_TYPE X_COMMON	Converts the buffer to/from XML using the configured view class.
MBSTRING	Passes the buffer as is. **Note:** Encoding is determined by the "`encoding`" element of the MetaData of the message sent or received.

Messaging Service Type

The following table shows the behavior of the Tuxedo Transport depending on the Tuxedo buffer type when the service type is **Messaging Service**.

Note: If **None** is specified in the subtype, the Tuxedo Transport should not receive any buffer.

Table 5-2 Buffer Transformation for Messaging Service

	Messaging Type			
Tuxedo Buffer Type	**Binary**	**Text**	**MFL**	**XML**[1]
STRING	Passed as is	Passed as is	Passed as is[2]	XML
CARRAY	Passed as is	Not supported	Passed as is [2]	XML
FML/FML32	Passed as is	Not supported	Not supported	XML
XML	Passed as is	Passed as is	Not supported	XML
VIEW/VIEW32	Passed as is	Not supported	Not supported	XML
MBSTRING	Passed as is	Passed as is	Passed as is [2]	XML

1. The Tuxedo Transport handles the buffer manipulation the same way as if the service was of type "Any XML service"
2. Assuming the buffer is in suitable format

For unsupported mappings, the Tuxedo Transport returns an error.

CHAPTER 6

Tuxedo Transport Transaction Processing

BEA AquaLogic Service Bus and BEA Tuxedo can interoperate to use the services that each product offers. Services often include transaction processing. Tuxedo Transport takes advantage of transactions or starts transactions in AquaLogic Service Bus.

Note: The exception to this transaction support is when the inbound transport is Tuxedo with a transactional message and the outbound is request/response XA-JMS. In this case, AquaLogic Service Bus detects this exception and results in a `TPESYSTEM` error.

The Tuxedo Transport transactional behavior is driven by the Quality of Service (QoS) setting available at the message context level. For information about Quality of Service setting, refer to the "Quality of Service" section in Modeling Message Flow in AquaLogic Service Bus in the *AquaLogic Service Bus User Guide*.

The following sections explain how the Tuxedo Transport handles transactions.

- Inbound Services
- Outbound Services

Inbound Services

When a transactional context is received, the message going into the pipeline sets the QoS to **Exactly Once,** otherwise QoS is set to **Best Effort**.

When a TransportException is caught before the reply is sent back to the client, the request aborts by throwing a TPESYSTEM exception and a transaction rollback results.

Outbound Services

When the thread calling the Business Service has a transactional context, the Tuxedo Transport behaves in the following manner.

- If QoS is set to **Exactly Once**, the Tuxedo Transport automatically forwards the transactional context to the remote domain unless the endpoint is configured to suspend the transaction.
- If QoS is set to **Best Effort**, the Tuxedo Transport suspends the transaction before making the call and resumes it after the call. This is equivalent to making an ATMI call with `TPNOTRAN` flag set.

When the thread calling the Business Service has no transaction associated, the Tuxedo call occurs nontransactionally, regardless of the QoS setting. In this case, it will correspond to a `tpcall()` or `tpacall()` with the `TPNOTRAN` flag set.