# BEA AquaLogic Service Bus Upgrade Guide



## Contents

# Upgrade Overview

You can upgrade ALSB configurations from versions 2.1, 2.5, 2.6, or 2.6RP1 to version 3.0. Version 2.0 cannot be upgraded.

The upgrade of *in-place* domains is not supported; only a *migration* upgrade method is supported. That is, you must create a new ALSB 3.0 domain, then import a configuration that was exported from a 2.x domain into the newly created 3.0 domain. In effect, you move the configuration from the old domain to the new 3.0 domain.

The upgrade to 3.0 domains is supported for both clustered and non-clustered domains.

💡 Before starting the upgrade process, please read the Upgrade Considerations section.

The following table identifies the version of WebLogic Server on which each version of ALSB runs.

| ALSB version | WLS version |
|---|---|
| 2.1 | 9.1 |
| 2.5 | 9.2 |
| 2.6 | 9.2 |
| 3.0 | 10.0 |

# Migration Upgrade

All migration steps are manual; no wizard is provided to facilitate the migration of an ALSB 2.x domain to an ALSB 3.0 domain. To upgrade an ALSB domain, complete the steps described in this section.

## Step 1. Export the ALSB 2.x Configuration

Use the ALSB Console to export the ALSB 2.x configuration that you want to upgrade. To do so, log on as Administrator, and select **Export Resources** from the **System Administration** panel in the Console. For

information about exporting ALSB configurations, see the appropriate *Using the AquaLogic Service Bus Console* document:

- "Exporting Configuration Data" in System Administration (2.1)
- "Exporting Configuration Data" in System Administration (2.5)
- "Exporting Configuration Data" in System Administration (2.6)

You can also export configurations using the ALSB DeploymentMBean (for 2.1) and the ALSBConfigurationMBean (for 2.5) and ALSBConfigurationMBean (for 2.6). For information, see the appropriate ALSB Deployment Guide:

- Using the ALSB Deployment API (for 2.1)
- Using the Deployment APIs (for 2.5)
- Using the Deployment APIs (for 2.6)

**Note:** In most cases, you cannot export WebLogic Server resources, such as the JMS resources, SNMP trap settings, or the Work Manager definitions. You must re-create these objects in the new ALSB domain, as described in Step 6.

## Step 2. Export the Security Configurations

Use the WebLogic Server Administration Console to export security data from the domain. In the WebLogic Server Administration Console, select **Domain Structure > Security Realms**, then choose the security realm. Select **Migration > Export** to export the data.
The following table summarizes the security data and the types of security providers in which it is stored.

| Security Data | Security Provider Type |
|---|---|
| User accounts | Authentication Provider |
| Group definitions | Authentication Provider |
| Role definitions | Role Mapping Provider |
| User names and passwords in service accounts | Username/Password Credential Mapping Provider |
| PKI credential map entries | PKI Credential Mapping Provider |
| SAML Relying Parties | SAML Credential Mapping Provider V2 |
| SAML Asserting Parties | SAML Identity Assertion Provider V2 |
| Trusted Certificates (for SSL and WSS) | Certification Path Provider (Certificate Registry) |

**Note** The set of providers to export is different depending on whether you are upgrading from 2.1 or 2.5 as described in the following sections:

- Exporting 2.1 Security Configurations
- Exporting 2.5 and 2.6 Security Configurations

### Exporting 2.1 Security Configurations

If you created service accounts and added user names and passwords to the service accounts, then your domain includes a username/password credential mapping provider. If the domain includes this provider or a PKI credential mapping provider, you must configure the export process to export credential mapping passwords in clear text (unencrypted). Your new domain cannot use passwords that were encrypted by a different domain.

**To Export Credential Mapping Data with Unencrypted Passwords**

1. Carefully restrict access to the directory and file into which you export credential maps so that unauthorized users cannot read the unencrypted passwords. When you import the credential maps into the new domain, the credential mapping provider encrypts the passwords. After the upgrade is complete, securely dispose of the file containing the unencrypted passwords.
2. Export data from each security provider individually.
   Although WebLogic Server allows you to export all of the security data in a single export operation, choose to export data from each security provider individually. The single export operation does not allow you to export passwords in clear text.
3. While exporting data from the credential mapping providers, export the passwords for the credentials in clear text, as follows: When the WebLogic Server Administration Console displays a page with the **Export Constraints** text box, enter passwords=cleartext.

For more information, see Migrating Security Data in *Securing WebLogic Server*.

### Exporting 2.5 and 2.6 Security Configurations

Starting with the ALSB 2.5 release, both PKI and username/password credentials are stored in both the WebLogic Server realm and in the ALSB configuration repository. Consequently, these credentials are exported as part of the configuration JAR that was generated and exported in Step 1 of this procedure. When the JAR is imported into the new domain, the realm data is populated based on the contents of the JAR file. This means that you do not need to export PKI Credentials or username/password credentials when you upgrade from ALSB 2.5 or 2.6.

## Step 3. Install ALSB 3.0

Install the ALSB 3.0 software as described in the ALSB Installation Guide

## Step 4. Create a New ALSB 3.0 Domain

Create a new ALSB 3.0 domain using the Domain Configuration Wizard or using the offline configuration tools, as described in:

- Creating a New ALSB Domain in Creating WebLogic Domains Using the Configuration Wizard
  or
- "Creating and Extending Domains" in Using Offline Configuration Tools.

## Step 5. Configure WebLogic Server Security

In the new domain, configure the WebLogic security framework with SSL and the security providers needed to support your proxy and business services. See Configuring the WebLogic Security Framework: MainSteps in the *ALSB Security Guide*.

Note the following security-related information about importing a 2.x configuration:

- In ALSB 2.5, the WebLogic Default Authorization provider and Default Role Mapping provider was deprecated. Instead of configuring these providers in your new domain, BEA recommends that you use the WebLogic XACML Authorization provider and XACML Role Mapping provider. Later in the upgrade process, Step 7, you can import 2.1 or 2.5 policies and role maps into the XACML providers. See Deprecated Security Features in the *ALSB 2.5 Release Notes*.
- If your new domain uses a PKI credential mapping provider, copy the keystores to the new domain and configure the PKI credential mapping provider to use the keystore.
- If your 2.1, 2.5, or 2.6 domain has modified the Web Service security configurations named *_SERVICE_BUS_INBOUND_WEB_SERVICE_SECURITY_MBEAN_* or *_SERVICE_BUS_OUTBOUND_WEB_SERVICE_SECURITY_MBEAN_*, make the same modifications in the 3.0 domain.
  For example, if in your 2.1 domain, you added the **UseX509ForIdentity** property to the*_SERVICE_BUS_INBOUND_WEB_SERVICE_SECURITY_MBEAN_* configuration (which is required to support inbound authentication with an X.509 token), add the property in the new domain. See Use X.509 certificates to establish identity in *The WebLogic Server Administration Console Online Help*.
- Session-Aware Access Control Management of Proxy Services. Using a pre-import process, ALSB 3.0 will perform in-place upgrade of a 2.x JAR to 3.0. See Session-Aware Access Control Management of Proxy Services.

## Step 6. Recreate Other WebLogic Server Objects

In the new ALSB 3.0 domain, recreate the WebLogic Server objects that could not be exported in Step 1, including:

- JMS resources, such as connection factories, queues, topics, and so on.
- Work Manager definitions
- SNMP agent and trap destination settings

For more information about configuring WebLogic Server domain resources, see Overview of WebLogic Server System Administration in *Introduction to BEA WebLogic Server and BEA WebLogic Express*.

**Tip:** You should configure the domain-scoped SNMP agent in the WebLogic Server Console. WebLogic Server 10.x has enhanced SNMP features. For more information about SNMP, see the WebLogic SNMP Management Guide.

- Add the Tuxedo domain ID as a WebLogic Server user (this is a requirement to invoke a successful request to a Tuxedo service)
- Configure WTC Local Access Point and Remote Access Point resources when your configuration includes Tuxedo transport-based services

For information, see Configuring WebLogic Tuxedo Connector for Tuxedo Transport in *Interoperability Solution for Tuxedo*.

## Step 7. Import Security Data

Use the WebLogic Server Administration Console to import the security data that you exported in Step 2 into the new ALSB domain. See Import data into a security provider in the *WebLogic Server Administration Console Online Help*.
Note the following:

- Import the security information for each security provider separately.
- See Only One Credential Mapping Provider Allowed.
- BEA recommends that you import access control policies into the WebLogic XACML Authorization Provider. If you exported data from the WebLogic Default Authorization Provider in your 2.1 or 2.5 domain, when you import into the XACML Authorization Provider, make sure that you select DefaultAtz from the **Import Format** list.
- BEA recommends that you import security role maps into the WebLogic XACML Role Mapping Provider. If you exported data from the WebLogic Default Role Mapper Provider in your 2.1 or 2.5 domain, when you import into the XACML provider, make sure you select DefaultRoles in the **Import Format** list.

## Step 8. Import the ALSB Configuration Data

Import the 2.x configuration data that you exported in Step 1 into the new 3.0 domain.

**Note**: You can either import the configuration data into a newly created domain, or you can import a configuration JAR that contains only projects, folders, or services unrelated to the artifacts you are importing into an existing domain. Importing artifacts from a 2.x domain into a 3.0 domain of the same type and name as in the 3.0 domain is undefined.

To import configuration data, log on to the ALSB Console as Administrator, and select **Import Resources** from the **System Administration** panel. For information about importing ALSB configurations, see "Importing Resources" in Using the AquaLogic Service Bus Plug-in for WorkSpace Studio.

### 2.1 Service Accounts

In 2.1 service accounts, the import process attempts to re-bind each 2.1 service account to the user names and passwords that are in the username/password credential mapping provider. For example, if your 2.1 domain included a service account with the user name of "pat" and password of "patspassword", the import process looks in the username/password credential mapping provider in the 3.0 domain for "pat" and "patspassword." If the import process does not find the credentials for a service account in the username/password credential mapping provider, you must add credentials to the service account before you can activate the session. You cannot import empty service accounts into ALSB.

For each 2.1 proxy service provider, the import process does the following:

- Searches the PKI credential mapping provider for alias-to-key-pair bindings that match those in the imported proxy service provider. If it finds a match, the import process enables the proxy service provider to use those key-pair bindings. If it does not find a match, the import process imports the proxy service provider without any key-pair bindings.

While it is valid to create a proxy service provider that does not contain any key-pair bindings, to use the provider to provide credentials, you must use the ALSB Console to add key-pair bindings to the proxy service provider.
- Prompts you to remove X.509 certificates that were used only for Web Service Security (WSS) authentication.

In ALSB 3.0, you cannot create a proxy service provider that supplies an X.509 credential only for WSS authentication. However, you can create a proxy service provider that supplies X.509 credentials for digital signatures, digital encryption, or SSL client authentication. The proxy service provider uses the X.509 digital-signature credential for those web services that require the certificate for both WSS authentication and digital signature.

If a 2.1 proxy service provider contained a digital-signature credential and an X.509 authentication credential, and if both credentials refer to the same key-pair, the import process does not import the X.509 token authentication credential. You do not need to remove the credential. To confirm that the X.509 token authentication credential will not be imported into the 3.0? domain, the import process outputs the following message:
*Service Provider has been upgraded. The Web Service Security X.509 Token key has been removed. This credential has been deprecated in ALSB 2.5. The Digital Signature key will be used instead*.

For information about the security changes in ALSB versions after 2.1, see Transport-Level Access Control Changes After ALSB 2.1. For additional information about the security changes between ALSB 2.1 and 2.5, see Security Updates Expand Configuration Options in "What's New in ALSB" in the *ALSB Release Notes*.

### Step 9. Complete Any Remaining Manual Upgrade Procedures

Some ALSB domain configuration changes are not automated and must be implemented manually. See Upgrade Considerations.

This completes the migration of the 2.x configuration into your new ALSB 3.0 domain.

# Upgrade Considerations

This section describes considerations for upgrading various ALSB configuration artifacts. It describes how ALSB 2.1, 2.5, and 2.6 differ in behavior from ALSB 3.0 in areas that may impact the configurations you are upgrading.

- 2.x to 3.0 Upgrade Considerations
- 2.1 or 2.5 to 3.0 Upgrade Considerations
- 2.1 to 3.0 Upgrade Considerations

### 2.x to 3.0 Upgrade Considerations

**Note:** You cannot upgrade ALSB 2.0 to 3.0.

Please read the following upgrade considerations when you are upgrading 2.1, 2.5, or 2.6 configurations to 3.0:



## ALSB 3.0 IDE

In ALSB 3.0, many of the design time features that were in the ALSB Console are now available in the ALSB IDE.

If you want to use the IDE instead of the Console, you can import a 2.6 configuration JAR directly into the 3.0 IDE. For information about importing a JAR file into the 3.0 IDE, see "Importing Resources" in Import/Export in *Using the AquaLogic Service Bus IDE*.

To import a 2.1 or 2.5 configuration JAR into the 3.0 IDE, you must first import the JAR into the 3.0 ALSB Console. From the console, export the configuration JAR file and then import it into the 3.0 IDE.

**Note:** When you import any configuration JAR into the 3.0 IDE, the operational and administrative settings are removed. To retain these settings, import the same configuration JAR into the corresponding ALSB 3.0 Console. Operational and administrative settings are only preserved in the console.

For information about exporting JAR files, see Step 1. Export the ALSB 2.x Configuration.

## Alert Rules

The SLA alert rules functionality in ALSB 3.0 differ slightly from previous releases. These changes do not affect the run-time evaluation or how alerts are issued. However, you may notice the following changes:

- In ALSB 2.1, 2.5, and 2.6, Alert Rule resources were created as separate resources and individually maintained. In ALSB 3.0 alert rules are part of the service definition. Because Alert Rules are part of the service definition and are no longer resources themselves, this affects their display in the *References* and *Referenced By* pages in the ALSB Console. For information about viewing references, see *View References* in Working with Projects, Folders, and Resources in *Using the AquaLogic Service Bus Console*.
- You may notice minor changes in the content of the alerts as issued to various destinations. For information about alert destinations, see Alert Destinations in *Using the AquaLogic Service Bus Console*.
- In ALSB 3.0, if an alert rule is renamed, then for the alerts issued in the past and under the old name, the console will no longer provide access to the rule definition on the "Alert Summary" and "Extended SLA Alert History" pages.

## Displaying References from Alerts to Alert Destinations

Unlike previous releases where you could see distinct entries of alert rules in the ALSB Console, references to alert destinations through alert rules from proxy and business services are maintained and displayed as a single reference. For example, in a proxy service, if multiple alert rules and multiple pipeline alert actions use the same alert destination, only one entry for the alert destination is displayed in the *Referenced By* page for that alert destination. For more information, see Alert Destinations in *Using the AquaLogic Service Bus Console*.

Since Alerts are no longer separate resources in ALSB 3.0, the way references from alerts to alert destinations are displayed is different from previous releases. Two pages in the Console are affected. First, the "Referenced By" field in the Alert Destination page no longer displays the Alert Rule that is referencing the destination. Instead, the Reference BY field displays the service that contains the alert. A side effect of this is that if a service has multiple alerts (SLA alerts or pipeline alerts) that reference the same Alert destination, the associated service is listed only once in the Referenced By field. Second, the Alert Rule page no longer contains the Reference information. Instead, the Service Summary page for the associated service contains the Alert Destinations referenced in the "References" field. For more information, see Alert Destinations in *Using the AquaLogic Service Bus Console*.

## Details Sent to Alert Destinations

When an SLA alert is issued to configured destinations, the alert details include a Rule ID as in previous releases. However, in 3.0, the value of the Rule ID is set to a combination of the parent service's global name and the alert rule name. For the SNMP trap, the RuleID is truncated to 64 characters, as in previous releases.

## Import - Export Alert Rule Changes

Unlike ALSB 2.6, which during import merged alerts with any existing alerts, ALSB provides users with preserve-overwrite semantics. This allows you to either keep all existing alerts or overwrite them, regardless of whether the alerts have the same name or not.

## Session-Aware Access Control Management of Proxy Services

The exported JARs from ALSB 2.1, 2.5, or 2.6 do not contain any access control policies. Before importing a configuration JAR from an earlier release, ALSB 3.0 uses a *pre-import* access control policy to do an *in-place* upgrade of the JAR to 3.0. To perform the in-place upgrade, the pre-processor queries all manageable authorization providers for each proxy service and retrieves a list of applicable access control policies. It then inserts those policies into the service definition of the proxy service. This is done on best-effort basis.

A manageable authorization provider is a authorization provider that implements the PolicyEditorMBean interface. Such providers expose read-write APIs that allow the WebLogic Server and the ALSB console to add, modify, and delete policies stored in them.

For transport level and default message-level policies, the system queries only those providers that expose the PolicyEditorMBean and retrieves any applicable policies and inserts these policies into the service definition.

For operational message-level policies, the system can only query providers that have implemented the

PolicyListerMBean. For providers that have not implemented the PolicyListerMBean interface, the operation-level policies are not retrieved.

After the in-place upgrade finishes, the import process proceeds as if the configuration JAR is of type 3.0, including the access control policies retrieved from the authorization providers. The following table explains various combinations of the applicable parameters and the outcome of the import process. Note the outcome is the result of the import process and does not represent anything done after the configuration is imported.

**Table 1: Applicable Parameters and Import Outcomes**

| JAR 1 Version | Proxy | ACLs exist in core repository | Preserve Policies | ACLs in Session Service Definition | Explanation |
|---|---|---|---|---|---|
| 2.6 | New | NA (No) | NA (No) | From APs 3 | Upgrades the JAR to 3.0, which involves pulling all applicable ACLs from all manageable authorization providers. The service definition in session will have ACLs from the configuration JAR directly from the Authorization Providers. |
| 2.6 | Exists | No | No | From APs 3 | Upgrades the JAR to 3.0. The service definition in session has ACLs from the configuration JAR directly from the Authorization Providers. |
| 2.6 | Exists | No | Yes | None | Upgrades the JAR to 3.0. The service definition in session has **no** ACLs. |

| | | | | | |
|---|---|---|---|---|---|
| 2.6 | Exists | Yes | No | From APs 3 | Upgrades the JAR to 3.0. The service definition in session has ACLs from the configuration JAR directly from the Authorization Providers. |
| 2.6 | Exists | Yes | Yes | From Core 2 | Upgrades the JAR to 3.0. The service definition in session **retains** ACLs from the core repository. |

1 JAR — The configuration JAR file being imported.

2 CORE — The core repository in the configuration framework.

3 APs — Manageable authorization providers.

If an authorization provider does not exist in the target ALSB 3.0 system, the import ignores the imported ACLS for the authorization provider and displays a warning. In this case, you can discard the session, or undo the import task, and then add the authorization providers to the server and re-import. Alternately, you can do a dummy update operation of security parameters in the ALSB Console and the system will auto-correct any conflicts that it can on best-effort basis. Note that unlike the changes described in About Access Control Policies during Upgrade, these changes are atomic and reversible if you discard the session.

For more information about the updating security parameters, see the Message Level Security Configuration page in *Using the AquaLogic Service Bus Console*.

## Transport SDK and Transport Provider Changes

The following changes may impact the ALSB configuration:

- Message Retry Count for Business Service Configuration
- Duplicate URIs for a Business Service are Removed
- Application Errors Retries
- HTTPS transport changes
- Transport Configuration in the Design Environment

### Message Retry Count for Business Service Configuration

In ALSB 2.6, the message retry count applies to the list of URIs for a business service. In 3.0, the retry count applies to the individual URL endpoints. The upgrade process maintains the 2.6 behavior as follows:

new_retries = N-1 + old_retries*N

where N is the total number of URIs and old_retries is the 2.6 retry count.

For example, suppose that in ALSB 2.6 you have three URLs configured for the business service and a retry count of one. With the 2.6 retry mechanism all three URLs are tried. Then after the retry delay, all three URLs are retried again. To obtain the same behavior in 3.0, the retry count is changed to five, which is obtained by applying the formula: (3 - 1) + (1*3) = 5. The net effect is exactly the same: all three URLs are tried once (using two of the five retries), then after the retry delay, the three URLs are tried once more (using the last three of the five retries).

If only a single URL is configured, the old behavior and the new behavior are the same; the retry count does not change during the upgrade.

For more information, see Business Services: Creating and Managing in *Using the ALSB Console*. Also see New Errors Codes in ALSB 3.0.

### Duplicate URIs for a Business Service are Removed

When importing business services into ALSB 3.0, the import process removes the duplicate URIs in the 2.6 configurations. If the URIs use randomly-weighted load balancing algorithms and the weights are set, the weights are adjusted accordingly. For example, if the business service is configured with the following URIs and weights:

- URI_A 1
- URI_B 3
- URI_A 1

When the business service is imported into 3.0, the URI set is modified as follows:

- URI_A 2
- URI_B 3

For Business services configured with other algorithms, the upgrade removes the duplicate URIs and no other changes are made.

For more information about setting the parameters for the load balancing algorithm, see Business Services: Creating and Managing in Using the *AquaLogic Service Bus Console*.

### Application Errors Retries

In case of delivery failure when sending outbound requests, ALSB allows you to specify whether to retry endpoint URIs for application errors, such as a SOAP fault. This does not affect retries for communication errors. This new option is available on the Transport Configuration page for business services. For more

information, see "Transport Configuration page" in Business Services: Creating and Managing in *Using the AquaLogic Service Bus Console*. After the number of retries is exceeded, an error is issued; see New Errors Codes in ALSB 3.0.

To maintain the 2.6 behavior, the default value for upgrading is set to Yes on the Transport Configuration page.

To use this option, you must configure the transport provider to convey to the Transport SDK whether the exception is communication error or application error. In ALSB 3.0, the Transport SDK is enhanced for this purpose. For more information, see Developing a Transport Provider in the *Transport SDK User Guide*.

### HTTPS transport changes

**Note:** This functionality is only applicable to HTTP proxy services.

To simplify switching between HTTP and HTTPS, in the ALSB 3.0 Console, the HTTPS transport configuration has been removed and its functionality has been added to the 3.0 inbound HTTP transport provider. The 3.0 HTTP Transport Configuration page contains a check box to enable to HTTPS.

A new element, `use-https`, is added to the schema of the HTTP Transport inbound properties.

```
<xs:element name="use-https" type="xs:boolean" minOccurs="0"/>
```

Any existing HTTPS Transport configurations are upgraded to HTTP transport with this flag set to `true`.

### Transport Configuration in the Design Environment

In prior releases, ALSB transports were configured only through the ALSB Console. However, starting in ALSB 3.0, the transports can be designed on Eclipse. For more information, see Developing ALSB Transports for WorkSpace Studio in the *Transport SDK User Guide*.

## 2.1 or 2.5 to 3.0 Upgrade Considerations

Please read the following considerations when you upgrade either ALSB 2.1 or 2.5 configurations to ALSB 3.0:

- "Use SSL" Attribute Controls Whether SSL is Used to Access JMS Queues
- SOAP Services Imported from 2.1 or 2.5 JARs use SOAP 1.1 by Default
- UDDI Configuration
- Operational Customization

### "Use SSL" Attribute Controls Whether SSL is Used to Access JMS Queues

ALSB JMS services (proxy and business) have a "Use SSL" attribute that controls whether SSL should be

used to access the JMS queues. However, in ALSB 2.1 and 2.5, JMS business services did not use SSL when reading the outbound responses even if "Use SSL" was specified. This was corrected in ALSB 2.6. This means that when such a business service (JMS request/response business service with "Use SSL" selected) from ALSB 2.1 or 2.5 is imported into a newer domain, there may be a problem if the outbound response URL corresponds to a non-SSL port. Attempts to use SSL to talk to this non-SSL port results in an error.

**Workaround**: When you import a 2.1 or 2.5 configuration JAR that contains a request/response outbound JMS business service with "Use SSL" specified, a warning is issued in the ALSB Console. To fix change the outbound response queue URL to use the SSL port.

## SOAP Services Imported from 2.1 or 2.5 JARs use SOAP 1.1 by Default

All of the SOAP services imported from 2.1 or 2.5 JARs use SOAP 1.1 by default. ALSB 2.6 and 3.0 releases support SOAP 1.2.

## UDDI Configuration

The UDDI auto import feature is enhanced in 2.6 and later releases to allow the auto synchronization with UDDI registries. When a change occurs in the registry for a service to which ALSB is subscribed, notifications are sent from the UDDI registry to ALSB.

For a single node, the notification is sent to the managed server HTTP address. In clustered configurations, the notification is sent to the Admin server HTTP address.

An "Auto Import" flag is added to the registry configuration to indicate whether auto synchronization is enabled for the registry. While importing an ALSB 2.1 or 2.5 JAR file, ALSB disables this flag to retain the original behavior.

## Operational Customization

Operational parameters were enhanced in ALSB 2.6 and again in 3.0. Consequently, values that were set for some parameters in 2.1 or 2.5 configurations are set differently in the 3.0 domain. The following table describes the value that ALSB 3.0 uses for Service Operational Parameters.

| Parameter | Version | Value set in 3.0 Configuration |
|---|---|---|
| Service State | Existing | As specified in imported JAR |
| 'Monitoring' enable/disable | Existing | As specified in imported JAR |
| 'Monitoring Aggregation Interval' | Existing | As specified in imported JAR<br>If none specified, then default is set to 10 min. |
| 'Reporting' enable/disable | Introduced in 2.6 a | Enable |
| 'Tracing' enable/disable | Existing | As specified in imported JAR |
| 'Logging' enable/disable<br>'Logging' severity level | Introduced in 2.6 a<br>Introduced in 2.6 a | Enable<br>Debug |

| | | |
|---|---|---|
| 'SLA Alerting' enable/disable<br>'SLA Alerting' severity level | Introduced in 2.6 a<br>Introduced in 2.6 a | Enable<br>Normal |
| 'Pipeline Alerting' enable/disable<br>'Pipeline Alerting' severity level | Introduced in 2.6 a<br>Introduced in 2.6 a | If Monitoring is Enabled: Enable and Normal<br>If Monitoring is Disabled: Disabled and Normal |
| 'Offline Endpoint URIs' enable/disable<br>Associated 'Retry Interval' field | New in 3.0<br>New in 3.0 | Disabled<br>0 hours 0 mins 0 secs |
| 'Throttling State' enable/disable<br>Associated 'Maximum Concurrency' field<br>Associated 'Throttling Queue' field<br>Associated 'Message Expiration' field | New in 3.0<br>New in 3.0<br>New in 3.0<br>New in 3.0 | Disabled<br>0<br>0 messages<br>0 msecs |

a Parameters set in a 2.6 configuration will retain the value specified in the imported JAR when imported into a 3.0 domain.

## 2.1 to 3.0 Upgrade Considerations

When upgrading ALSB 2.1 configurations to ALSB 3.0, consider the following:

- Some Error Codes Are Not Generated in 2.5 (or later) Versions
- New Error Codes Require Update
- Users in the IntegrationOperator Role Do Not Have Export Privileges
- Only One Credential Mapping Provider Allowed
- 2.1 SLA Alert Logs are Unavailable
- New Alert Summary Field
- 3.0 Alert Destination Resources are Created from 2.1 Alert Rules
- Run Time Does Not Generate Missing Headers when Sending Multipart Messages
- Transport-Level Access Control Changes After ALSB 2.1

**Note:** These issues do not apply when upgrading 2.5 or 2.6 configurations to 3.0.

### Some Error Codes Are Not Generated in 2.5 (or later) Versions

In ALSB 2.5 and later versions, error codes BEA-382101, BEA-382102, and BEA-382151 are not generated while preparing an inbound response or outbound request.

In ALSB 2.1, these errors were generated for the conditions as described in the following:

- BEA-382101 — invalid content assigned to $inbound/transport/response
- BEA-382102 — invalid content assigned to $outbound/transport/request
- BEA-382151 — invalid service name assigned to $outbound@name

In ALSB 2.1, these errors are caught in the binding layer at run time. In ALSB 2.5, 2.6, and 3.0, these errors are caught at design time in the Replace action and result in BEA-382040 error code, which indicates that an Assign action failed. Additionally, BEA-382040 may also be returned, which indicates that the replace action failed updating the body variable.

## New Error Codes Require Update

If you use WSS or relied on specific ALSB 2.1 error codes, either on proxy service error-handlers or client-side code, note the following change in ALSB 2.6 and later.

Whenever WebLogic Server WSS returns a SOAP fault to ALSB, the ALSB message-context has a fault with:

- BEA-386201 — A web service security fault occurred [<root-wss-error>][<root-wss-fault-string>] where:
  - root-wss-error is the error-code from the WebLogic Server WSS SOAP fault.
  - root-wss-fault-string is the fault-string from the WebLogic Server WSS SOAP fault.
- An instance of XML element WebServiceSecurityFault. This XML element also contains the root-fault error-code, fault-string, and fault-details.

The ALSB default error handler returns the root SOAP fault to the client.

**Workaround**:

- Update your error-handlers and/or client-side code to deal with the new error codes.
- You can also write an error-handler that maps the new error-codes back to the ALSB 2.1 error code. However, this is not a BEA-recommended approach.

## Users in the IntegrationOperator Role Do Not Have Export Privileges

In ALSB 2.1, users in the IntegrationOperator role were allowed to export ALSB configurations. However, in subsequent releases of ALSB, they are not; the workaround is to reassign such users to a different role.

## Only One Credential Mapping Provider Allowed

Only one PKI and one username/password credential mapping provider is allowed.

In AquaLogic 2.1, you can have multiple PKI credential mapping providers and multiple username/password credential mapping providers. In AquaLogic Service 2.5 and later, you can configure only one PKI credential mapping provider and only one username/password credential mapping provider. Consequently, if you are upgrading from AquaLogic 2.1 and you have created multiple PKI or username/password credential mapping providers, you must import all PKI mapping data into a single PKI credential mapping provider and import all username/password mapping data into a single username/password credential mapping provider.

## 2.1 SLA Alert Logs are Unavailable

In 2.1, SLA alerts were captured in the WebLogic Diagnostics Framework (WLDF) log. Migration to a 3.0 domain removes the contents of the 2.1 log. Consequently, the alerts and their details are not displayed in the new domain's ALSB Console and
are removed from the reporting log when you perform an upgrade.

If the 2.1 alerts are issued in E-mail, they continue to be available in the user E-mail accounts after the upgrade. However, if any of those alerts were configured with a JMS action only (that is, with either a JMS queue or JMS topic defined as the JMS destination), you must set up new queues and topics in the new (upgrade) domain. Consequently, the old JMS actions are lost.

## New Alert Summary Field

In ALSB 2.1 you could not customize the content of the alert summary field when you defined an E-mail action for an SLA alert. All alert summaries (the contents of which populate the E-mail's Subject line) contained the text "ALSB Alert". In ALSB 2.5 and later releases, a new customizable alert-summary field is provided when you configure SLA alerts and pipeline alert actions.

For those SLA alerts that are migrated from 2.1, ALSB 3.0 populates the alert summary field with the 2.1 text "ALSB Alert". After you complete the upgrade, you can change the message in the alert summary field to something more descriptive. For more information about configuring alert actions, see "Alert" under Proxy Service: Actions in *Using the AquaLogic Service Bus Console*. See also "Creating and Alert Rule" in Monitoring in *Using the AquaLogic Service Bus Console*.

## 3.0 Alert Destination Resources are Created from 2.1 Alert Rules

A resource called an Alert Destination was introduced in ALSB 2.5. It is used to capture a list of recipients that can receive alert notifications from ALSB. When an SLA alert rule is upgraded from 2.1, the alert actions configured in the 2.1 SLA Alert Rule are extracted and used to create an Alert Destination resource. The SLA Alert Rule is then updated to reference this resource.

The Alert Destination resides in the same project and folder as the service with which the alert rule is associated. The name of the Alert Destination is specified as Alert Destination - xxxxxx, where xxxxxx is a unique number.

The upgrade process creates one Alert Destination for each unique combination of recipients. In other words, if ten SLA Alert Rules with the same set of recipients are upgraded from 2.1, only one Alert Destination resource is created in the same project and folder as the service that is associated with the first SLA Alert Rule.

For information about Alert Destinations, see Alert Destinations in *Using the AquaLogic Service Bus Console*.

## Run Time Does Not Generate Missing Headers when Sending Multipart Messages

If an ALSB proxy service receives a multipart message (that is, a message with attachments) where the

root part does not have an associated Content-ID MIME header, subsequent multipart messages sent by that proxy service will not have a Content-ID MIME header for the root part. In 2.1, ALSB corrected for the missing headers in client messages by generating the header when sending multipart messages. In 2.6 and later releases, the missing header is not automatically generated. Therefore, you must ensure that clients sending multipart messages to ALSB include a Content-ID header and "start" parameter.

The presence of this Content-ID header directly affects the presence of the "start" parameter in the "multipart/related" Content-Type of the multipart message.

While the Content-ID header and "start" parameter are considered optional by MIME standards, some Web Service stacks may require them and may return an error response back to the proxy service if they are absent.

For more information, see Message Context in the *AquaLogic Service Bus User Guide*.

## Transport-Level Access Control Changes After ALSB 2.1

In ALSB 2.1, transport-level access control was limited to HTTP and HTTPS proxy services. Access control was enforced by the web-container. The authorization check was done against a weblogic.security.service.URLResource. See http://e-docs.bea.com/wls/docs91/javadocs/weblogic/security/service/URLResource.html

In 2.5 and later versions, ALSB has a transport-level access control check on entry to all proxy services, regardless of transport. The call to the authorization service is now done in ALSB code; the web-container no longer performs an authorization check. As a side-effect, the check is now done against a com.bea.wli.sb.security.ALSBProxyServiceResource. The default policy on ALSBProxyServiceResource grants access to all requests. You can configure a transport-level access control policy on a proxy service in the ALSB console, as described in http://e-docs.bea.com/alsb/docs30/consolehelp/securityconfiguration.html.

This change has the following implications:

- A policy is composed of one or more policy conditions, grouped together by boolean operators. Most policy conditions can be used to secure any resources, such as is-user-in-role. However, some policy conditions can only secure resources of a specific type. The policy conditions that can be applied only to URLResource are as follows:
  weblogic.entitlement.rules.HttpRequestAttrIsSet
  weblogic.entitlement.rules.HttpRequestAttrNumberPredicate
  weblogic.entitlement.rules.HttpRequestAttrNumberEquals
  weblogic.entitlement.rules.HttpRequestAttrNumberGreater
  weblogic.entitlement.rules.HttpRequestAttrNumberLess
  weblogic.entitlement.rules.HttpRequestAttrStringEquals
  In ALSB 2.1 you could use these policy conditions to secure HTTP/S proxy services, but these policy conditions are not allowed in ALSB 2.5 and later versions.
- The context properties passed to the security framework are also different.
  **Note**: Context properties are passed in a weblogic.security.service.ContextHandler instance. For more information, see
  Interface ContextHandler.
  Prior to ALSB 2.5, the web-container passed some URLResource-specific context properties to the security providers. ALSB 2.5 and later pass a different set of context properties. Some of these are

described in "Adding Policy Conditions" in Security Configuration in *Using the AquaLogic Service Bus Console*.

- As previously mentioned, for ALSB 2.5 and later, all proxy service invocations go through an access control check, including colocated optimized calls, calls to local-transport proxies, and calls from the ALSB test console.
- Because URLResource is not used in 2.5 and later, a proxy service's transport-level access control policy is no longer tied to the endpoint's URI. You can change the proxy service URI and the existing policy will still take effect. However, the policy is now dependent on the proxy service location (project and/or folders) and name. Operations such as resource/project/folder rename, move, clone, delete, and undo do not have any effect on an authorization provider's policy database. Be careful when performing these operations to avoid loosing an access control policy or leaving orphan access control policies in the system. See Known Issue for CR222177 in the ALSB Release Notes at the following URL: http://e-docs.bea.com/alsb/docs26/relnotes/.
  **Note**: This issue has been fixed in 3.0.
- The default policy on ALSBProxyServiceResource is automatically created in new ALSB domains.

### About Access Control Policies during Upgrade

During upgrade of ALSB 2.1, ALSB checks to see if the default policy on ALSBProxyServiceResource exists for at least one authorization provider. If this policy does not exist, it is created. Read access to the policy database is optional — if an authorization provider does not support reads, ALSB displays the following alert message:

> *"ALSB could not determine if the default ALSBProxyServiceResource policy is present or not because some authorization providers do not implement PolicyReaderMBean. ALSB could not deploy the policy because neither EntitleNet provider nor XACML provider is present. If the policy is indeed missing, the administrator must create it."*

Similarly, write access to the policy database is optional. If an authorization provider does not provide write access, ALSB displays the following alert message:

> *"ALSB has determined the default ALSBProxyServiceResource policy is missing. ALSB could not deploy the policy because neither EntitleNet provider nor XACML provider is present. Access to all ALSB proxy services will be denied. The administrator must create the policy using the provider tools."*

**Note**: The EntitleNet provider was deprecated in ALSB 2.5 and is not supported in ALSB 2.6 or later. If you are using the EntitleNet provider, you should upgrade to the XACML authorization provider.

During a 2.1 upgrade, access control policies on HTTP or HTTPS proxy services are automatically migrated. If a policy exists on the URLResource that matches the service URI, the policy is copied over to the corresponding ALSBProxyServiceResource. The original policy (the one on URLResource) is deleted. The one exception to this is if the original policy used one of the URLResource-specific conditions. In this case the policy cannot be upgraded and ALSB creates a policy for this service, which denies all access to the service and writes the following alert to the log file:

> *"[POLICY MIGRATION] [proxy service: <service>] [authorization provider: <provider>] The 2.1*

*policy cannot be migrated because it makes use of policy predicates which are specific to URLResource. A deny-all policy will be bound to the proxy. You must re-configure this policy in the console."*

⚠ **Warning** These automatic changes to the policy database occur while staging an ALSB configuration JAR during import. These changes are not atomic. Consider the scenario where a user creates a 3.0 ALSB session and imports a 2.1 configuration JAR, which causes some automatic policy updates. If the user now decides to abandon the ALSB session (by undoing the changes without activating) the policy changes are permanent and not rolled back.

## New Errors Codes in ALSB 3.0

The following error codes are added in ALSB 3.0:

- BEA-380001 — This transport specific status code indicates an application error. Also see Application Errors Retries.
- BEA-380002 — This transport specific status code indicates a connection error. Also see Message Retry Count for Business Service Configuration.
- BEA-380100 — This transport specific status code indicates that the target throttled business service could not process any more messages and there is no throttling queue configured for receiving backlog messages.
- BEA-380101 — This transport specific status code indicates that the message expired while waiting in the throttling queue to be processed by throttled business service.
- BEA-380102 — This transport specific status code indicates that the target throttled business service could not process the message and could not be inserted in the throttling queue because the queue is full and the message priority is lower than any of the messages already in this queue.

For more information about transports, see the following:

- Endpoint URIs — Managing Endpoint URIs in ALSB Console in the *AquaLogic Service Bus Operations Guide*.
- Application Retries — Transport SDK User Guide
- Throttling:
  - Throttling in AquaLogic Service Bus in the *AquaLogic Service Bus Operations Guide*.
  - "Configuring Operational Settings for Business Services" under Monitoring in *Using the AquaLogic Service Bus Console*.

---



BEA AquaLogic ServiceBus Upgrade Guide
Version 3.0
Document Date: February 2008