

Table of Contents

Overview

- Editions

Get Started

- Create BizTalk Services

How To

- Plan and design

 - Administration and developer task list

- Back up and restore

- Configure

 - Throttling

 - Service settings and monitoring

- Migrate

 - BizTalk Server EDI solutions to BizTalk Services

- Monitor

 - Tasks that depend on service states

- Secure

 - Access control

- Troubleshoot

- Hybrid connections

 - Overview

 - Create and manage

Resources

- Azure Roadmap

- MSDN Forum

- Pricing

- Release notes

- Service updates

- Stack Overflow

- Videos

BizTalk Services: Editions chart

6/27/2017 • 6 min to read • [Edit Online](#)

IMPORTANT

Microsoft Azure BizTalk Services (MABS) is being retired, and replaced with Azure Logic Apps. If you currently use MABS, then [Move from BizTalk Services to Logic Apps](#) provides some guidance on moving your integration solutions to Logic Apps.

If you're brand new to Logic Apps, then we suggest getting started here:

- [What are Logic Apps](#)
- [Create your first logic app](#), or quickly get started using a [pre-built template](#)
- [View all the available connectors](#) you can use in a logic app

Azure BizTalk Services offers several editions. Use this article to determine which edition is right for your scenario and business needs.

Compare the editions

Free (Preview)

Can create and manage Hybrid Connections. A Hybrid Connection is an easy way to connect an Azure website to an on-premises system, like SQL Server.

Developer

Includes Hybrid Connections, EAI and EDI message processing with an easy-to-use trading partner management portal, and support for common EDI schemas and rich EDI processing over X12 and AS2. Can create common EAI scenarios connecting services in the cloud with any HTTP/S, REST, FTP, WCF and SFTP protocols to read and write messages. Utilize connectivity to on-premises LOB systems with ready-to-use SAP, Oracle eBusiness, Oracle DB, Siebel, and SQL Server adapters. Use a developer centric environment with Visual Studio tools for easy development and deployment. Limited to development and test purposes only with no Service Level Agreement (SLA).

Basic

Includes most of the Developer capabilities with increases in Hybrid Connections, EAI bridges, EDI Agreements, and BizTalk Adapter Pack connections. Also offers high availability, and the option to scale with a Service Level Agreement (SLA).

Standard

Includes all the Basic capabilities with increases in Hybrid Connections, EAI bridges, EDI Agreements, and BizTalk Adapter Pack connections. Also offers high availability, and the option to scale with a Service Level Agreement (SLA).

Premium

Includes all the Standard capabilities with increases in Hybrid Connections, EAI bridges, EDI Agreements, and BizTalk Adapter Pack connections. Also includes archiving, high availability, and the option to scale with a Service Level Agreement (SLA).

Editions chart

The following table lists the differences.

	FREE (PREVIEW)	DEVELOPER	BASIC	STANDARD	PREMIUM
Starting price	Azure BizTalk Services Pricing Azure Pricing Calculator				
Default minimum configuration	1 Free Unit	1 Developer Unit	1 Basic Unit	1 Standard Unit	1 Premium Unit
Scale	No Scale	No Scale	Yes, in increments of 1 Basic unit	Yes, in increments of 1 Standard unit	Yes, in increments of 1 Premium unit
Maximum allowed scale out	No Scale	No Scale	Up to 8 Units	Up to 8 Units	Up to 8 Units
EAI Bridges per unit	Not included	25	25	125	500
EDI, AS2 Includes TPM agreements	Not included	Included. 10 agreements per unit.	Included. 50 agreements per unit.	Included. 250 agreements per unit.	Included. 1000 agreements per unit.
Hybrid Connections per unit	5	5	10	50	100
Hybrid Connection Data Transfer (GB) per Unit	5	5	50	250	500
BizTalk Adapter Service connections to on-premises LOB systems	Not included	1 connection	2 connections	5 connections	25 connections

Supported protocols/Systems: <ul style="list-style-type: none"> • HTTP • HTTPS • FTP • SFTP • WCF • Service Bus (SB) • Azure Blob • REST APIs 	Not included	Included	Included	Included	Included
High availability For Service Level Agreement (SLA), see Azure BizTalk Services Pricing .	Not included	Not included	Included	Included	Included
Backup and restore	Not included	Included	Included	Included	Included
Tracking	Not included	Included	Included	Included	Included
Archiving Includes Non-repudiation of Receipt (NRR) and downloading tracked messages	Not included	Included	Not Included	Not Included	Included
Use of custom code	Not included	Included	Included	Included	Included
Use of transforms, including custom XSLT	Not included	Included	Included	Included	Included

NOTE

For resiliency against hardware failures, High Availability implies having multiple VMs within a single BizTalk Unit.

FAQs

What is a BizTalk unit?

A "unit" is the atomic level of an Azure BizTalk Services deployment. Each edition comes with a unit that has different compute capacity and memory. For example, a Basic unit has more compute than Developer, Standard has more compute than Basic, and so on. When you scale a BizTalk Service, you scale in terms of units.

What is the difference between BizTalk Services and Azure BizTalk VM?

BizTalk Services provides a true Platform-as-a-Service (PaaS) architecture for building integration solutions in the cloud. With the PaaS model, you focus completely on the application logic and leave all of the infrastructure management to Microsoft, including:

- No need to manage or patch virtual machines.
- Microsoft ensures availability.
- You control scale on-demand by simply requesting more or less capacity through the Azure portal.

BizTalk Server on Azure Virtual Machines provides an Infrastructure-as-a-Service (IaaS) architecture. You create virtual machines and configure them exactly like your on-premises environment, making it easier to run existing applications in the cloud with no code changes. With IaaS, you are still responsible for configuring the virtual machines, managing the virtual machines (for example, installing software and OS patches), and architecting the application for high availability.

If you are looking at building new integration solutions that minimize your infrastructure management effort, use BizTalk Services. If you are looking to quickly migrate your existing BizTalk solutions or looking for an on-demand environment to develop and test BizTalk Server applications, use BizTalk Server on an Azure Virtual Machine.

What is the difference between BizTalk Adapter Service and Hybrid Connections?

The BizTalk Adapter Service is used by an Azure BizTalk Service. The BizTalk Adapter Service uses the BizTalk Adapter Pack to connect to an on-premises Line of Business (LOB) system. A Hybrid Connection provides an easy and convenient way to connect Azure applications, like the Web Apps feature in Azure App Service and Azure Mobile Services, to an on-premises resource.

What does "Hybrid Connection Data Transfer (GB) per unit" mean? Is this per minute/hour/day/week/month? What happens when the limit is reached?

The Hybrid Connection cost per unit depends on the BizTalk Services edition. Simply put, costs depend on how much data you transfer. For example, transferring 10 GB data daily costs less than transferring 100 GB daily. Use the [Pricing Calculator](#) for BizTalk Services to determine specific costs. Typically, the limits are enforced daily. If you exceed the limit, any overage is charged at the rate of \$1 per GB.

When I create an agreement in BizTalk Services, why does the number of bridges go up by two instead of just one?

Each agreement comprises of two different bridges: a send side communication bridge and a receive side communication bridge.

What happens when I hit the quota limit on the number of bridges or agreements?

You are not able to deploy any new bridges or create any new agreements. To deploy more, you need to scale up to more units of the BizTalk service, or upgrade to a higher edition.

How do I migrate from one tier of BizTalk Services to another?

The Free edition cannot be migrated or 'scaled up' to another tier, and cannot be backed up and restored to another tier. If you need another tier, create a new BizTalk Service using the new tier. Any artifacts created using the Free edition, including hybrid connections, need to be recreated in the new BizTalk Service.

For the remaining editions, use the backup and restore for migrating your artifacts from one tier to another. For example, backup your artifacts in the Standard tier, and then restore them to the Premium tier. [BizTalk Services: Backup and Restore](#) describes the supported migration paths and lists what artifacts are backed up. Note that Hybrid Connections are not backed up. After backing up and restoring to a new tier, you then recreate the hybrid connections.

Is the BizTalk Adapter Service included in the service? How do I receive the software?

Yes, the BizTalk Adapter Service with the BizTalk Adapter Pack are included with the Azure BizTalk Services SDK [download](#).

Next steps

To create Azure BizTalk Services in the Azure portal, go to [BizTalk Services: Provisioning using the Azure portal](#). To

start creating applications, go to [Azure BizTalk Services](#).

Additional resources

- [BizTalk Services: Provisioning using the Azure portal](#)
- [BizTalk Services: Provisioning Status Chart](#)
- [BizTalk Services: Dashboard, Monitor and Scale tabs](#)
- [BizTalk Services: Backup and restore](#)
- [BizTalk Services: Throttling](#)
- [BizTalk Services: Issuer Name and Issuer Key](#)
- [How do I Start Using the Azure BizTalk Services SDK](#)

Create BizTalk Services using the Azure portal

6/27/2017 • 8 min to read • [Edit Online](#)

IMPORTANT

Microsoft Azure BizTalk Services (MABS) is being retired, and replaced with Azure Logic Apps. If you currently use MABS, then [Move from BizTalk Services to Logic Apps](#) provides some guidance on moving your integration solutions to Logic Apps.

If you're brand new to Logic Apps, then we suggest getting started here:

- [What are Logic Apps](#)
- [Create your first logic app](#), or quickly get started using a [pre-built template](#)
- [View all the available connectors](#) you can use in a logic app

TIP

To sign in to the Azure portal, you need an Azure account and Azure subscription. If you don't have an account, you can create a free trial account within a few minutes. See [Azure Free Trial](#).

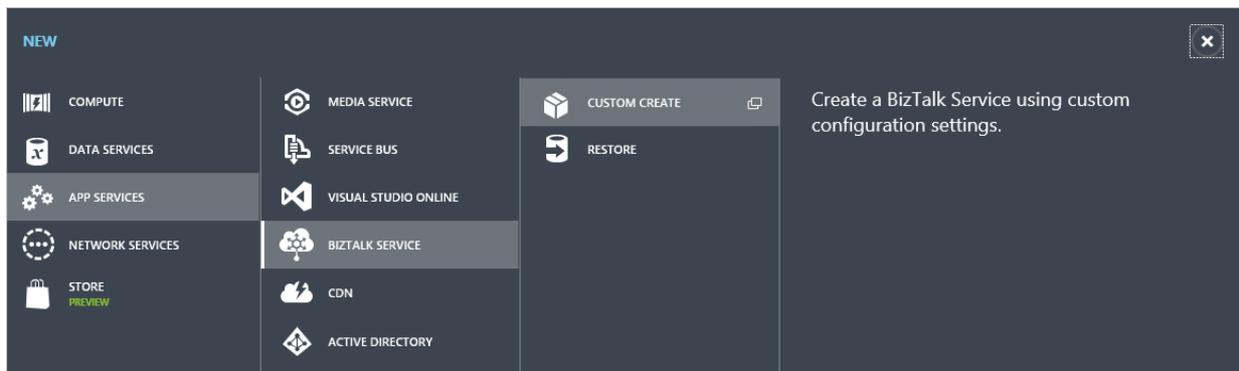
Create a BizTalk Service

Depending on the Edition you choose, not all BizTalk Service settings may be available.

1. Sign in to the [Azure portal](#).
2. In the bottom navigation pane, select **NEW**:



3. Select **APP SERVICES** > **BIZTALK SERVICE** > **CUSTOM CREATE**:



4. Enter the BizTalk Service settings:

BizTalk service name	<p>You can enter any name but be specific. Some examples include:</p> <p><i>mycompany.biztalk.windows.net</i> <i>mycompanymyapplication.biztalk.windows.net</i> <i>myapplication.biztalk.windows.net</i></p> <p>".biztalk.windows.net" is automatically added to the name you enter. This creates a URL that is used to access your BizTalk Service, like https://myapplication.biztalk.windows.net</p>
Edition	<p>If you are in the testing/development phase, choose Developer. If you are in the production phase, use the BizTalk Services: Editions Chart to determine if Premium, Standard, or Basic is the correct choice for your business scenario.</p>
Region	<p>Select the geographic region to host your BizTalk Service.</p>
Domain URL	<p>Optional. By default, the domain URL is <i>YourBizTalkServiceName.biztalk.windows.net</i>. A custom domain can also be entered. For example, if your domain is <i>contoso</i>, you can enter:</p> <p><i>MyCompany.contoso.com</i> <i>MyCompanyMyApplication.contoso.com</i> <i>MyApplication.contoso.com</i> <i>YourBizTalkServiceName.contoso.com</i></p>

Select the NEXT arrow.

5. Enter the Storage and Database Settings:

Monitoring/Archiving storage account	<p>Select an existing storage account or create a new storage account.</p> <p>If you create a new Storage account, enter the Storage Account Name.</p>
Tracking database	<p>If you use an existing Azure SQL Database, it cannot be used by another BizTalk Service. You need the login name and password entered when that Azure SQL Database Server was created.</p> <p>TIP Create the Tracking database and Monitoring/Archiving storage account in the same region as the BizTalk Service.</p>

Select the NEXT arrow.

6. Enter the Database settings:

Name	<p>Available when Create a new SQL Database instance is selected in the previous screen.</p> <p>Enter a SQL Database name to be used by your BizTalk Service.</p>
-------------	--

Server	Available when Create a new SQL Database instance is selected in the previous screen. Select an existing SQL Database Server or create a new SQL Database server.
Server login name	Enter the login user name.
Server login password	Enter the login password.
Region	Available when Create a new SQL Database instance is selected. Select the geographic region to host your SQL Database.

Select the check mark to complete the wizard. The progress icon appears:



When complete, the Azure BizTalk Service is created and ready for your applications. The default settings are sufficient. If you want to change the default settings, select **BIZTALK SERVICES** in the left navigation pane, and then select your BizTalk Service. Additional settings are displayed in the [Dashboard, Monitor, and Scale tabs](#) at the top.

Depending on the state of the BizTalk Service, there are some operations that cannot be completed. For a list of these operations, go to [BizTalk Services State Chart](#).

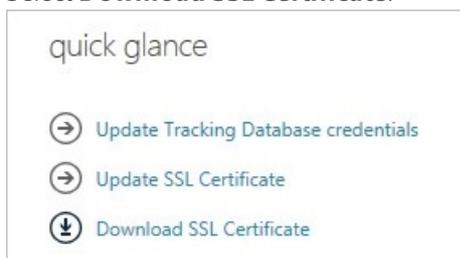
Post-provisioning steps

- [Install the certificate on a local computer](#)
- [Add a production-ready certificate](#)
- [Get the Access Control namespace](#)

Install the certificate on a local computer

As part of BizTalk Service provisioning, a self-signed certificate is created and associated with your BizTalk Service subscription. You must download this certificate and install it on computers from where you either deploy BizTalk Service applications or send messages to a BizTalk Service endpoint.

1. Sign in to the [Azure portal](#).
2. Select **BIZTALK SERVICES** in the left navigation pane, and then select your BizTalk Service subscription.
3. Select the **Dashboard** tab.
4. Select **Download SSL Certificate**:



5. Double-click the certificate and run through the wizard to install the certificate. Make sure you install the certificate under the **Trusted Root Certificate Authorities** store.

Add a production-ready certificate

The self-signed certificate that is automatically created when creating BizTalk Services is intended for use in development environments only. For production scenarios, replace it with a production-ready certificate.

1. On the **Dashboard** tab, select **Update SSL Certificate**.
2. Browse to your private SSL certificate (*CertificateName.pfx*) that includes your BizTalk Service name, enter the password, and then click the check mark.

Get the Access Control namespace

1. Sign in to the [Azure portal](#).
2. Select **BIZTALK SERVICES** in the left navigation pane, and then select your BizTalk Service.
3. In the task bar, select **Connection Information**:

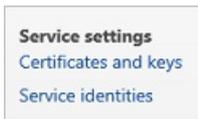


4. Copy the Access Control values.

When you deploy a BizTalk Service project from Visual Studio, you enter this Access Control namespace. The Access Control namespace is automatically created for your BizTalk Service.

The Access Control values can be used with any application. When Azure BizTalk Services is created, this Access Control namespace controls the authentication with your BizTalk Service deployment. If you want to change the subscription or manage the namespace, select **ACTIVE DIRECTORY** in the left navigation pane and then select your namespace. The task bar lists your options.

Clicking **Manage** opens the Access Control Management Portal. In the Access Control Management Portal, the BizTalk Service uses **Service identities**:



The Access Control service identity is a set of credentials that allow applications or clients to authenticate directly with Access Control and receive a token.

IMPORTANT

The BizTalk Service uses **Owner** for the default service identity and the **Password** value. If you use the Symmetric Key value instead of the Password value, the following error may occur.

Could not connect to the Access Control Management Service account with the specified credentials

[Managing Your ACS Namespace](#) lists some guidelines and recommendations.

Requirements explained

These requirements do not apply to the Free Edition.

What you need	Why you need it
---------------	-----------------

<p>Azure subscription</p>	<p>The subscription determines who can sign in to the Azure portal. The Account holder creates the subscription at Azure Subscriptions.</p> <p>The Azure account can have multiple subscriptions and can be managed by anyone who is permitted. For example, your Azure account holder creates a subscription named <i>BizTalkServiceSubscription</i> and gives the BizTalk Administrators within your company (for example, ContosoBTSAdmins@live.com) access to this subscription. In this scenario, the BizTalk Administrators sign in to the Azure portal and have full Administrator rights to all the hosted services in the subscription, including Azure BizTalk Services. The BizTalk Administrators are not the Azure account holders and therefore don't have access to any billing information.</p> <p>Manage Subscriptions and Storage Accounts in the Azure portal provides more information.</p>
<p>Azure SQL Database</p>	<p>Stores the tables, views, and stored procedures used by the BizTalk Service, including the Tracking data.</p> <p>When you create a BizTalk Service, you can use an existing Azure SQL Server, Azure SQL Database, or automatically create a new Server or Database.</p> <p>The SQL Database scale is automatically configured. Typically, the default scale is sufficient for a BizTalk Service. Changing the scale impacts pricing. See Accounts and Billing in Azure SQL Database</p> <p>Notes</p> <ul style="list-style-type: none"> • When you create a new Azure SQL Server and Database, Azure Services is automatically enabled. The BizTalk Service requires Azure Services be enabled. • If you create a new Azure SQL Database on an existing Azure SQL Server, the firewall rules of the Server are not changed. As a result, it's possible other Azure Services are not allowed access to the Server's databases.
<p>Azure Access Control namespace</p>	<p>Authenticates with Azure BizTalk Services. When you deploy a BizTalk Service project from Visual Studio, you enter this Access Control namespace. When you create a BizTalk Service, the Access Control namespace is automatically created.</p>

<p>Azure Storage account</p>	<p>Gives access to tables, blobs, and queues used by your BizTalk Service to save the following:</p> <ul style="list-style-type: none"> • Log files that monitor the BizTalk Service. The monitoring output is also displayed in the Monitoring tab in the Azure portal. • When creating an X12 or AS2 agreement between partners, you can enable the Archiving feature to store message properties. This data is saved in the Storage account. <p>When you create a BizTalk Service, you can use an existing Storage account or automatically create a new Storage account.</p> <p>The default Storage settings are sufficient for a BizTalk Service.</p> <p>When you create a Storage account, a Primary Key and Secondary Key are automatically created. These Keys control access to your Storage account. The BizTalk Service automatically uses the Primary Key.</p> <p>See Storage for more information.</p>
<p>SSL private certificate</p>	<p>When an Azure BizTalk Service is created, an HTTPS URL that includes your BizTalk Service name is also created. This URL is automatically configured to use a self-signed development-only certificate. For production, you need a private SSL certificate.</p> <p>Important SSL Certificate Information</p> <ul style="list-style-type: none"> • The certificate expiration date must be less than 5 years. • All private certificates require a password. Know this password and as a best practice, share this password with your administrators. • Self-signed certificates are used in a test/development environment. When using self-signed certificates, import the certificate to your Personal certificate store and the Trusted Root Certification Authorities certificate store. <p>When sending the production certificate request to your certification authority, give the following certificate properties:</p> <ul style="list-style-type: none"> • Enhanced Key Usage: At a minimum, Azure BizTalk Services requires Server Authentication. • Common Name: Enter the fully qualified domain name (FQDN) of your Azure BizTalk Service URL. See Create a BizTalk Service in this article. <p>A new or different certificate can be added after the BizTalk Service is created.</p>

Hybrid Connections

When you create an Azure BizTalk Service, the **Hybrid Connections** tab is available:



Hybrid Connections are used to connect an Azure website or Azure mobile service to any on-premises resource that uses a static TCP port, such as SQL Server, MySQL, HTTP Web APIs, Mobile Services, and most custom Web Services. Hybrid Connections and the BizTalk Adapter Service are different. The BizTalk Adapter Service is used to connect Azure BizTalk Services to an on-premises Line of Business (LOB) system.

See [Hybrid Connections](#) to learn more, including creating and managing Hybrid Connections.

Next steps

Now that a BizTalk Service is created, familiarize yourself with the different [BizTalk Services: Dashboard, Monitor and Scale tabs](#). Your BizTalk Service is ready for your applications. To start creating applications, go to [Azure BizTalk Services](#).

See also

- [BizTalk Services: Editions Chart](#)
- [BizTalk Services: State Chart](#)
- [BizTalk Services: Backup and Restore](#)
- [BizTalk Services: Throttling](#)
- [BizTalk Services: Issuer Name and Issuer Key](#)
- [How do I Start Using the Azure BizTalk Services SDK](#)
- [Hybrid Connections](#)

Administration and Development Task List in BizTalk Services

6/27/2017 • 2 min to read • [Edit Online](#)

IMPORTANT

Microsoft Azure BizTalk Services (MABS) is being retired, and replaced with Azure Logic Apps. If you currently use MABS, then [Move from BizTalk Services to Logic Apps](#) provides some guidance on moving your integration solutions to Logic Apps.

If you're brand new to Logic Apps, then we suggest getting started here:

- [What are Logic Apps](#)
- [Create your first logic app](#), or quickly get started using a [pre-built template](#)
- [View all the available connectors](#) you can use in a logic app

Getting Started

When working with Microsoft Azure BizTalk Services, there are several on-premises and cloud-based components to consider. To get started, consider the following process flow:

STEP	WHO'S RESPONSIBLE	TASK	RELATED LINKS
1.	Administrator	Create the Microsoft Azure Subscription using a Microsoft account or an Organizational account	Azure classic portal
2.	Administrator	Create or provision a BizTalk Service.	Create a BizTalk Service using Azure classic portal
3.	Administrator	Register you or your company's BizTalk Services deployment	Registering and Updating a BizTalk Service Deployment on the BizTalk Services Portal
4.	Administrator	Applies if the application uses BizTalk Adapter Service to connect to an on-premises Line-of-Business (LOB) system or uses a Queue or Topic Destination. Create the Azure Service Bus Namespace. Give this namespace, Service Bus Issuer Name, and Service Bus Issuer Key values to the developer.	How to: Create or Modify a Service Bus Service Namespace and Get Issuer Name and Issuer Key values

STEP	WHO'S RESPONSIBLE	TASK	RELATED LINKS
5.	Developer	Install the SDK and create the BizTalk Service project in Visual Studio.	Install Azure BizTalk Services SDK and Create Rich Messaging Endpoints on Azure
6.	Developer	Deploy your BizTalk Service project to your BizTalk Service hosted on Azure.	Deploying and Refreshing the BizTalk Services Project
7.	Administrator	Applies if you are using EDI. You can add Partners and create Agreements on the Microsoft Azure BizTalk Services Portal. When you create an Agreement, you can add the bridge and/or Transforms created by the developer to the Agreement settings.	Configuring EDI, AS2, and EDIFACT on BizTalk Services Portal
8.	Administrator	Using the Azure classic Portal, monitor the health of your BizTalk Service, including performance metrics.	BizTalk Services: Dashboard, Monitor and Scale tabs
9.	Administrator	Using the Microsoft Azure BizTalk Services Portal, manage the artifacts used by BizTalk Services, and track messages as they are processed by the bridge files.	Using the BizTalk Services Portal
10.	Administrator	Create a backup plan to back up the BizTalk Service.	Business Continuity and Disaster Recovery in BizTalk Services

Next Steps

[Tutorials and Samples](#)

[Create the project in Visual Studio](#)

[Install Azure BizTalk Services SDK](#)

Concepts

[Create the project in Visual Studio](#)

[EDI, AS2, and EDIFACT Messaging \(Business to Business\)](#)

Other Resources

[Add Source, Destination, and Bridge Messaging Endpoints](#)

[Learn and create Message Maps and Transforms](#)

[Using the BizTalk Adapter Service \(BAS\)](#)

[Azure BizTalk Services](#)

BizTalk Services: Backup and Restore

6/27/2017 • 7 min to read • [Edit Online](#)

IMPORTANT

Microsoft Azure BizTalk Services (MABS) is being retired, and replaced with Azure Logic Apps. If you currently use MABS, then [Move from BizTalk Services to Logic Apps](#) provides some guidance on moving your integration solutions to Logic Apps.

If you're brand new to Logic Apps, then we suggest getting started here:

- [What are Logic Apps](#)
- [Create your first logic app](#), or quickly get started using a [pre-built template](#)
- [View all the available connectors](#) you can use in a logic app

Azure BizTalk Services includes Backup and Restore capabilities. This topic describes how to backup and restore BizTalk Services using the Azure classic portal.

You can also back up BizTalk Services using the [BizTalk Services REST API](#).

NOTE

Hybrid Connections are NOT backed up, regardless of the Edition. You must recreate your hybrid connections.

Before you Begin

- Backup and Restore may not be available for all editions. See [BizTalk Services: Editions Chart](#).
- Using the Azure classic portal, you can create an On Demand backup or create a scheduled backup.
- Backup content can be restored to the same BizTalk Service or to a new BizTalk Service. To restore the BizTalk Service using the same name, the existing BizTalk Service must be deleted and the name must be available. After you delete a BizTalk Service, it can take longer time than wanted for the same name to be available. If you cannot wait for the same name to be available, then restore to a new BizTalk Service.
- BizTalk Services can be restored to the same edition or a higher edition. Restoring BizTalk Services to a lower edition, from when the backup was taken, is not supported.

For example, a backup using the Basic Edition can be restored to the Premium Edition. A backup using the Premium Edition cannot be restored to the Standard Edition.

- The EDI Control numbers are backed up to maintain continuity of the control numbers. If messages are processed after the last backup, restoring this backup content can cause duplicate control numbers.
- If a batch has active messages, process the batch **before** running a backup. When creating a backup (as needed or scheduled), messages in batches are never stored.

If a backup is taken with active messages in a batch, these messages are not backed up and are therefore lost.

- Optional: In the BizTalk Services Portal, stop any management operations.

Create a backup

A backup can be taken at any time and is completely controlled by you. This section lists the steps to create backups using the Azure classic portal, including:

On Demand backup

Schedule a backup

On Demand backup

1. In the Azure classic portal, select **BizTalk Services**, and then select the BizTalk Service you want to backup.
2. In the **Dashboard** tab, select **Back up** at the bottom of the page.
3. Enter a backup name. For example, enter *myBizTalkServiceBUDate*.
4. Choose a blob storage account and select the checkmark to start the backup.

Once the backup completes, a container with the backup name you enter is created in the storage account. This container contains your BizTalk Service backup configuration.

Schedule a backup

1. In the Azure classic portal, select **BizTalk Services**, select the BizTalk Service name you want to schedule the backup, and then select the **Configure** tab.
2. Set the **Backup Status** to **Automatic**.
3. Select the **Storage Account** to store the backup, enter the **Frequency** to create the backups, and how long to keep the backups (**Retention Days**):

The screenshot shows the 'automated backup' configuration interface in the Azure classic portal. At the top, there are navigation tabs: DASHBOARD, MONITOR, SCALE, and CONFIGURE. The main heading is 'automated backup'. Below this, there are four sections for configuration:

- BACKUP STATUS:** Two radio buttons are present: 'NONE' and 'AUTOMATIC'. The 'AUTOMATIC' option is selected and highlighted with a purple border.
- STORAGE ACCOUNT:** A dropdown menu with the text 'Choose a storage account' and a downward arrow.
- FREQUENCY:** This section includes 'Every' followed by a text input field containing '7' and the word 'days'. Below this is 'Start Date' followed by a text input field containing '2014-04-27', a time dropdown menu set to '2:30 PM', and the text 'Local Time (UTC-04:00)'.
- RETENTION DAYS:** A text input field containing '20' followed by the word 'days'. Below this is a checked checkbox with the text 'Always keep at least one backup'.

Notes

- In **Retention Days**, the retention period must be greater than the backup frequency.
- Select **Always keep at least one backup**, even if it is past the retention period.

4. Select **Save**.

When a scheduled backup job runs, it creates a container (to store backup data) in the storage account you entered. The name of the container is named *BizTalk Service Name-date-time*.

If the BizTalk Service dashboard shows a **Failed** status:

LAST SCHEDULED BACKUP STATUS
Failed

The link opens the Management Services Operation Logs to help troubleshoot. See [BizTalk Services: Troubleshoot using operation logs](#).

Restore

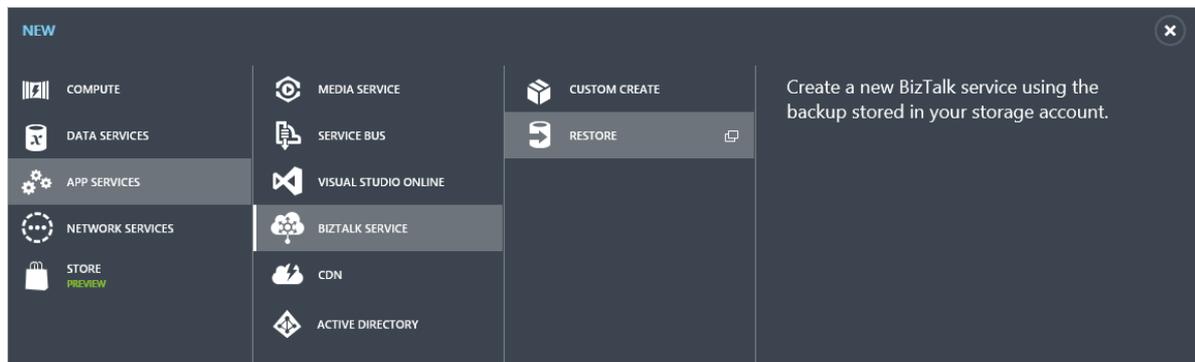
You can restore backups from the Azure classic portal or from the [Restore BizTalk Service REST API](#). This section lists the steps to restore using the classic portal.

Before restoring a backup

- New tracking, archiving, and monitoring stores can be entered while restoring a BizTalk Service.
- The same EDI Runtime data is restored. The EDI Runtime backup stores the control numbers. The restored control numbers are in sequence from the time of the backup. If messages are processed after the last backup, restoring this backup content can cause duplicate control numbers.

Restore a backup

1. In the Azure classic portal, select **New > App Services > BizTalk Service > Restore**:



2. In **Backup URL**, select the folder icon and expand the Azure storage account that stores the BizTalk Service configuration backup. Expand the container and in the right pane, select the corresponding back up .txt file.

Select **Open**.

3. On the **Restore BizTalk Service** page, enter a **BizTalk Service Name** and verify the **Domain URL**, **Edition**, and **Region** for the restored BizTalk Service. **Create a new SQL database instance** for the tracking database:

NEW BIZTALK SERVICE - RESTORE

Restore BizTalk Service

BACKUP URL Required

!

BIZTALK SERVICE NAME

.biztalk.windows.net ?

DOMAIN URL

?

EDITION

Basic v ?

REGION

East Asia v

TRACKING DATABASE

Create a new SQL Database instance v ?

Select the next arrow.

4. Verify the name of the SQL database, enter the physical server where the SQL database will be created, and a username/password for that server.

If you want to configure the SQL database edition, size, and other properties, select **Configure Advanced Database Settings**.

Select the next arrow.

5. Create a new storage account or enter an existing storage account for the BizTalk Service.
6. Select the checkmark to start the restore.

Once the restoration successfully completes, a new BizTalk Service is listed in a suspended state on the BizTalk Services page in the Azure classic portal.

After restoring a backup

The BizTalk Service is always restored in a **Suspended** state. In this state, you can make any configuration changes before the new environment is functional, including:

- If you created BizTalk Service applications using the Azure BizTalk Services SDK, you may need to update the Access Control (ACS) credentials in those applications to work with the restored environment.
- You restore a BizTalk Service to replicate an existing BizTalk Service environment. In this situation, if there are agreements configured in the original BizTalk Services portal that use a source FTP folder, you may need to update the agreements in the newly restored environment to use a different source FTP folder. Otherwise, there may be two different agreements trying to pull the same message.
- If you restored to have multiple BizTalk Service environments, make sure you target the correct environment in the Visual Studio applications, PowerShell cmdlets, REST APIs, or Trading Partner Management OM APIs.
- It's a good practice to configure automated backups on the newly restored BizTalk Service environment.

To start the BizTalk Service in the Azure classic portal, select the restored BizTalk Service and select **Resume** in the task bar.

What gets backed up

When a backup is created, the following items are backed up:

	ITEMS BACKED UP
Azure BizTalk Services Portal	
Configuration and Runtime	<ul style="list-style-type: none">• Partner and profile details• Partner Agreements• Custom assemblies deployed• Bridges deployed• Certificates• Transforms deployed• Pipelines• Templates created and saved in the BizTalk Services Portal• X12 ST01 and GS01 mappings• Control numbers (EDI)• AS2 Message MIC values
Azure BizTalk Service	
SSL Certificate	<ul style="list-style-type: none">• SSL Certificate Data• SSL Certificate Password
BizTalk Service Settings	<ul style="list-style-type: none">• Scale unit count• Edition• Product Version• Region/Datacenter• Access Control Service (ACS) namespace and key• Tracking database connection string• Archiving Storage account connection string• Monitoring storage account connection string
Additional Items	
Tracking Database	<p>When the BizTalk Service is created, the Tracking Database details are entered, including the Azure SQL Database Server and the Tracking Database name. The Tracking Database is not automatically backed up.</p> <p>Important If the Tracking Database is deleted and the database needs recovered, a previous backup must exist. If a backup does not exist, the Tracking Database and its data are not recoverable. In this situation, create a new Tracking Database with the same database name. Geo-Replication is recommended.</p>

Next

To create Azure BizTalk Services in the Azure classic portal, go to [BizTalk Services: Provisioning Using Azure classic portal](#). To start creating applications, go to [Azure BizTalk Services](#).

See Also

- [Backup BizTalk Service](#)
- [Restore BizTalk Service from Backup](#)
- [BizTalk Services: Developer, Basic, Standard and Premium Editions Chart](#)
- [BizTalk Services: Provisioning Using Azure classic portal](#)
- [BizTalk Services: Provisioning Status Chart](#)
- [BizTalk Services: Dashboard, Monitor and Scale tabs](#)
- [BizTalk Services: Throttling](#)
- [BizTalk Services: Issuer Name and Issuer Key](#)
- [How do I Start Using the Azure BizTalk Services SDK](#)

BizTalk Services: Throttling

6/27/2017 • 2 min to read • [Edit Online](#)

IMPORTANT

Microsoft Azure BizTalk Services (MABS) is being retired, and replaced with Azure Logic Apps. If you currently use MABS, then [Move from BizTalk Services to Logic Apps](#) provides some guidance on moving your integration solutions to Logic Apps.

If you're brand new to Logic Apps, then we suggest getting started here:

- [What are Logic Apps](#)
- [Create your first logic app](#), or quickly get started using a [pre-built template](#)
- [View all the available connectors](#) you can use in a logic app

Azure BizTalk Services implements service throttling based on two conditions: memory usage and the number of simultaneous messages processing. This topic lists the throttling thresholds and describes the Runtime behavior when a throttling condition occurs.

Throttling Thresholds

The following table lists the throttling source and thresholds:

	DESCRIPTION	LOW THRESHOLD	HIGH THRESHOLD
Memory	% of total system memory available/PageFileBytes. Total available PageFileBytes is approximately 2 times the RAM of the system.	60%	70%
Message Processing	Number of messages processing simultaneously	40 * number of cores	100 * number of cores

When a high threshold is reached, Azure BizTalk Services starts to throttle. Throttling stops when the low threshold is reached. For example, your service is using 65% system memory. In this situation, the service does not throttle. Your service starts using 70% system memory. In this situation, the service throttles and continues to throttle until the service uses 60% (low threshold) system memory.

Azure BizTalk Services tracks the throttling status (normal state vs. throttled state) and the throttling duration.

Runtime Behavior

When Azure BizTalk Services enters a throttling state, the following occurs:

- Throttling is per role instance. For example:
RoleInstanceA is throttling. RoleInstanceB is not throttling. In this situation, messages in RoleInstanceB are processed as expected. Messages in RoleInstanceA are discarded and fail with the following error:

Server is busy. Please try again.

- Any pull sources do not poll or download a message. For example:
A pipeline pulls messages from an external FTP source. The role instance doing the pull gets into a throttling state. In this situation, the pipeline stops downloading additional messages until the role instance stops throttling.
- A response is sent to the client so the client can resubmit the message.
- You must wait until the throttling is resolved. Specifically, you must wait until the low threshold is reached.

Important notes

- Throttling cannot be disabled.
- Throttling thresholds cannot be modified.
- Throttling is implemented system-wide.
- The Azure SQL Database Server also has built-in throttling.

Additional Azure BizTalk Services topics

- [Installing the Azure BizTalk Services SDK](#)
- [Tutorials: Azure BizTalk Services](#)
- [How do I Start Using the Azure BizTalk Services SDK](#)
- [Azure BizTalk Services](#)

See Also

- [BizTalk Services: Developer, Basic, Standard and Premium Editions Chart](#)
- [BizTalk Services: Provisioning Using Azure classic portal](#)
- [BizTalk Services: Provisioning Status Chart](#)
- [BizTalk Services: Dashboard, Monitor and Scale tabs](#)
- [BizTalk Services: Backup and Restore](#)
- [BizTalk Services: Issuer Name and Issuer Key](#)

Review the Dashboard, Monitor, Scale, Configure, and Hybrid Connection tabs

6/27/2017 • 12 min to read • [Edit Online](#)

IMPORTANT

Microsoft Azure BizTalk Services (MABS) is being retired, and replaced with Azure Logic Apps. If you currently use MABS, then [Move from BizTalk Services to Logic Apps](#) provides some guidance on moving your integration solutions to Logic Apps.

If you're brand new to Logic Apps, then we suggest getting started here:

- [What are Logic Apps](#)
- [Create your first logic app](#), or quickly get started using a [pre-built template](#)
- [View all the available connectors](#) you can use in a logic app

After you create your BizTalk Service and deploy your application, you can change some of the BizTalk Service settings and monitor the application performance.

When you open the Azure classic portal, you are automatically placed at the **ALL ITEMS** tab. To view your BizTalk Service, select your BizTalk Service in the **ALL ITEMS** tab or select the **BIZTALK SERVICES** tab; and then select your BizTalk Service name.

This opens a new window with the following tabs. This topic describes these tabs.

Quickstart

Depending on the BizTalk Services Edition, all options listed may not be available.

Get the tools	<p>Download the BizTalk Services SDK to install the Visual Studio project templates on your on-premises development computer. These templates create the BizTalk Services (bridge) and the BizTalk Service Artifacts (Transform) Visual Studio projects that are deployed to your BizTalk Service.</p> <p>How do I Start Using the Azure BizTalk Services SDK and Installing the Azure BizTalk Services SDK lists the steps to get started.</p>
Create partner agreements	<p>Opens the Azure BizTalk Services Portal hosted on Azure where you add partners and create X12, AS2, and EDIFACT EDI agreements.</p> <p>Configuring Components for EDI Messaging on BizTalk Services Portal lists the steps to get started.</p>
Learn more about BizTalk Services	<p>Go to the learning center to learn more about Azure BizTalk Services.</p>

In the task bar at the bottom, you can:

<p>Manage your application deployment</p>	<p>Opens the Azure BizTalk Services portal. The BizTalk Services Portal is the entrance to EDI configuration, including adding partners and creating X12, AS2, and EDIFACT agreements.</p> <p>This is the same as Create partner agreements on the Quick Start tab.</p> <p>Configuring Components for EDI Messaging on BizTalk Services Portal provides more information on the BizTalk Services Portal.</p>
<p>Connection Information of the Access Control Namespace</p>	<p>When you select Connection Information, then the Access Control Namespace, Default Issuer, and Default Key are displayed. You can copy these values.</p> <p>You can also open the Access Control Portal. Create an Access control Namespace provides more information on the Access Control Portal.</p>
<p>Sync Keys in the Storage Account</p>	<p>When you create a Storage account, a Primary Key and Secondary Key are automatically created. These encryption Keys control access to your Storage Account. Your BizTalk Service automatically uses the Primary Key. Sync Keys enable users to switch between the Primary Key and the Secondary Key without disrupting the BizTalk Service.</p> <p>For example, you want the BizTalk Service to use a new Primary Key for the Storage Account. To do this:</p> <ol style="list-style-type: none"> 1. Select your BizTalk Service and select Sync Keys. Select the Secondary Key. When you do this, the BizTalk Service starts using the Secondary Key. 2. In the Azure classic portal, select your Storage account and Regenerate the Primary Key. Remember, your BizTalk Service is using the Secondary Key. 3. Select your BizTalk Service and select Sync Keys. Now, select the Primary Key. This is the new Primary Key you regenerated. 4. In the Azure classic portal, select your Storage account and Regenerate the Secondary Key. <p>This process is called "rollover keys". The purpose is to enable users to switch between the Primary Key and the Secondary Key without disrupting the BizTalk Service.</p>
<p>Delete your application</p>	<p>When you select Delete, your BizTalk Service and all items deployed to it are removed.</p>

Dashboard

Depending on the BizTalk Services Edition, all options listed may not be available.

When you select your BizTalk Service name, the Dashboard tab is displayed. In Dashboard, you can:

Usage Overview: Shows the number of used Hybrid Connections

Also displays the data usage in GB.

Metric Graph: Shows a fixed list of performance metrics

These metrics provide real-time values regarding the health of the BizTalk Service. You can also choose the **Relative** or **Absolute** values and the time range **Interval** of the metrics that are displayed in the graph.

For a description of these performance metrics, go to [Available Metrics](#) in this topic.

Quick Glance: Lists your BizTalk Service properties

Update Tracking Database credentials	Changes the user name and password used to log into the Tracking Database.
Update SSL Certificate	Can update the BizTalk Service to use a different SSL certificate. A self-signed SSL certificate is automatically created when you create the BizTalk Service .
Download Certificate	You can download the SSL certificate used by your BizTalk Service to a local machine.
Status	Displays the current status of your BizTalk Service. See BizTalk Services: Service state chart .
Service URL	The URL for your BizTalk Service. This is the same as the Domain URL entered when your BizTalk Service is created.
Public Virtual IP (VIP) Address	The IP address assigned to your BizTalk Service. It is used for all input endpoints and is the source address for outbound traffic. This IP address belongs to your BizTalk Service as long as it is created. If you delete the BizTalk Service, the IP address is assigned to another BizTalk Service.
ACS Namespace	Authenticates with the BizTalk Service.
Edition	Lists the Edition entered when the BizTalk Service is created.
Location	Displays the geographic region that hosts your BizTalk Service.
Created	Displays the date and time the BizTalk Service was created.
Tracking Database	The Azure SQL Database name that stores the tracking tables used by your BizTalk Service. Requirements Explained provides details on the Tracking Database.
Monitoring/Archiving Storage	The Azure Storage account name that stores the monitoring output of your BizTalk Service. Requirements Explained provides details on the Storage account.
Subscription Name	Lists the subscription that hosts your BizTalk Service. The subscription governs access to the Azure classic portal.
Subscription ID	When a subscription is created, a subscription ID is automatically generated. When using REST APIs, you may need to enter the Subscription ID.

[BizTalk Services: Provisioning Using Azure classic portal](#) lists the steps to create a BizTalk Service.

Manage, Connection Information, Sync Keys, and Delete in the task bar:

<p>Manage your application deployment</p>	<p>Opens the Azure BizTalk Services Portal. The BizTalk Services Portal is the entrance to EDI configuration, including adding partners and creating X12, AS2, and EDIFACT agreements.</p> <p>This is the same as Create partner agreements on the Quick Start tab.</p> <p>Configuring Components for EDI Messaging on BizTalk Services Portal provides more information on the BizTalk Services Portal.</p>
<p>Connection Information of the Access Control Namespace</p>	<p>Displays the Access Control Namespace, Default Issuer, and Default Key values; which can be copied.</p> <p>You can also open the Access Control Portal. This Access Control Portal is the same as using the Active Directory option in the left navigation pane.</p> <p>Managing Your ACS Namespace provides more information on the Access Control Portal.</p>
<p>Sync Keys in the Storage Account</p>	<p>When you create a Storage account, a Primary Key and Secondary Key are automatically created. These encryption Keys control access to your Storage Account. Your BizTalk Service automatically uses the Primary Key. Sync Keys enable users to switch between the Primary Key and the Secondary Key without disrupting the BizTalk Service.</p> <p>For example, you want the BizTalk Service to use a new Primary Key for the Storage Account. To do this:</p> <ol style="list-style-type: none"> 1. Select your BizTalk Service and select Sync Keys. Select the Secondary Key. When you do this, the BizTalk Service starts using the Secondary Key. 2. In the Azure classic portal, select your Storage account and Regenerate the Primary Key. Remember, your BizTalk Service is using the Secondary Key. 3. Select your BizTalk Service and select Sync Keys. Now, select the Primary Key. This is the new Primary Key you regenerated. 4. In the Azure classic portal, select your Storage account and Regenerate the Secondary Key. <p>This process is called "rollover keys". The purpose is to enable users to switch between the Primary Key and the Secondary Key without disrupting the BizTalk Service.</p>
<p>Delete your application</p>	<p>Your BizTalk Service and all items deployed to it are removed.</p>

Monitor

Does not apply to the Free Edition.

When you select your BizTalk Service name, the Monitor tab is available and displays the following:

Metric Graph: Displays the selected performance metrics

These metrics provide real-time values regarding the health of the BizTalk Service. You choose which performance metrics are displayed. A maximum of six performance metrics can be displayed simultaneously.

You can also choose the **Relative** or **Absolute** values and the time range **Interval** of the metrics that are

displayed.

To remove or display metrics in the graph:

1. Select the **Monitor** tab.
2. Select **Add Metrics** in the task bar:



3. Check the performance metrics you want to display.
4. Select the checkmark to return to the **Monitor** tab.
5. Select the circle next to the metric to display that metric's value in the graph.

For example, the **CPU Usage** metric is grayed out; its output is not displayed in the graph:

<input type="radio"/> CPU Usage	1.77 %	3.73 %	2.14 %	---
---------------------------------	--------	--------	--------	-----

Select the grayed out circle to enable the **CPU Usage** metric to display its output in the graph:

<input checked="" type="radio"/> CPU Usage	1.77 %	3.73 %	2.14 %	---
--	--------	--------	--------	-----

6. To remove a metric from the display graph and the list, select **Delete Metric** in the task bar. To add the metric back to the list, select **Add Metrics** in the task bar, check the metric, and select the checkmark to return to the **Monitor** tab. Select the grayed out circle to enable the metric.

Available Metrics

The following performance counters/metrics are available:

RoundTrip Latency	<p>Displays the average time taken in milliseconds (ms) to process a message from the time the message is received until the message is fully processed by the BizTalk Service across all bridges. Only messages successfully processed are counted.</p> <p>When the following events occur, a timestamp is created:</p> <ul style="list-style-type: none">• Message enters the gateway• Message is routed to the destination• Destination response is received• Destination acknowledgement response sent to the gateway <p>This metric shows the result of the following calculation:</p> <p>[Destination acknowledgement response sent to the gateway] - [Message enters the gateway]</p>
Failures At Source	<p>Displays the total number of messages that failed by the BizTalk Service when pulling messages from the source endpoints.</p>
CPU Usage	<p>Lists the average %Processor Time of all role instances.</p>

<p>Processing Latency</p>	<p>Displays the average time taken In milliseconds (ms) to process a message by the BizTalk Service across all bridges, excluding the time spent in destinations. Only messages successfully processed are counted.</p> <p>When each of the following events occur, a timestamp is created:</p> <ul style="list-style-type: none"> • Message enters the gateway • Message is routed to the destination • Destination response is received • Destination acknowledgement response sent to the gateway <p>This metric shows the result of the following calculation:</p> <p>[Destination acknowledgement response sent to the gateway] - [Message enters the gateway] - [Destination response is received] + [Message is routed to the destination]</p>
<p>Failures In Process</p>	<p>Displays the total number of messages that failed during processing by the BizTalk Service across all the bridges within a time interval.</p>
<p>Messages Sent</p>	<p>Displays the total number of messages sent by the BizTalk Service across all bridges within a time interval. This metric is incremented when a message sent from a pipeline reaches the route destination. This metric does not indicate that a message is successfully processed.</p> <p>In a Request-Reply scenario, the metric is incremented when the route destination sends a receipt acknowledgement back to the pipeline.</p>
<p>Messages Received</p>	<p>Displays the total number of messages received by the BizTalk Service across all bridges within a time interval. This metric is incremented when a new message is received by the pipeline.</p>
<p>Messages In Process</p>	<p>Displays the total number of messages currently being processed by the BizTalk Service within a time interval.</p>
<p>Messages Processed</p>	<p>Displays the total number of messages successfully processed by the BizTalk Service across all bridges within a time interval. This metric is incremented when a message is successfully received by the pipeline and successfully routed to the destination.</p>

Scale

In the Scale tab, you can add or subtract the number of units used by your BizTalk Service. By default, there is one Unit configured. Additional Units can be added to scale your BizTalk Service. When you increase the scale, you are increasing throughput. The amount of resources also increases, including deployed bridges, agreements, LOB connections, and processing power. For example, you increase the scale from 1 Unit to 2 Units. In this situation, you can deploy double the number of bridges, double the agreements, double the LOB connections, and double the processing power.

Some BizTalk editions do not offer a scale option. In this situation, one Unit is permitted. To determine how many units your edition can be scaled, refer to [BizTalk Services: Editions Chart](#).

Increasing the number of units may impact pricing. If you increase the Units, selecting **Save** displays a message that tells you if billing is impacted. You then choose to continue. When you increase the number of Units, the BizTalk Service status changes from Active to Updating. In the Updating state, your BizTalk Service continues to run.

[BizTalk Services: Editions Chart](#) defines a "Unit".

Configure

Does not apply to Hybrid Connections.

Sets the Backup Status to None or Automatic. When set to None, no backups are automatically created. When set to Automatic, you configure the backup location, the frequency of the backup, and how long to keep the backup files.

[BizTalk Services: Backup and Restore](#) provides the details.

Hybrid Connections

Hybrid Connections connect an Azure application, like Web Apps or Mobile Apps in Azure App Service, to an on-premises resource that uses a static TCP port, such as SQL Server, MySQL, HTTP Web APIs, and most custom Web Services. Hybrid Connections are managed in BizTalk Services in the Azure classic portal.

To create Hybrid Connections in Azure App Service, see [Access on-premises resources using hybrid connections in Azure App Service](#).

To create or manage Hybrid Connections in Azure BizTalk Services, see [Hybrid Connections](#).

Next

Now that you're familiar with the different tabs, you can learn more about the Azure BizTalk Services features:

- [BizTalk Services: Throttling](#)
- [BizTalk Services: Issuer Name and Issuer Key](#)
- [BizTalk Services: Backup and Restore](#)

See Also

- [Hybrid Connections](#)
- [BizTalk Services: Developer, Basic, Standard and Premium Editions Chart](#)
- [BizTalk Services: Provisioning Using Azure classic portal](#)
- [BizTalk Services: BizTalk Service State Chart](#)
- [How do I Start Using the Azure BizTalk Services SDK](#)

Migrating BizTalk Server EDI Solutions to BizTalk Services: Technical Guide

6/27/2017 • 10 min to read • [Edit Online](#)

IMPORTANT

Microsoft Azure BizTalk Services (MABS) is being retired, and replaced with Azure Logic Apps. If you currently use MABS, then [Move from BizTalk Services to Logic Apps](#) provides some guidance on moving your integration solutions to Logic Apps.

If you're brand new to Logic Apps, then we suggest getting started here:

- [What are Logic Apps](#)
- [Create your first logic app](#), or quickly get started using a [pre-built template](#)
- [View all the available connectors](#) you can use in a logic app

Author: Tim Wieman and Nitin Mehrotra

Reviewers: Karthik Bharthy

Written using: Microsoft Azure BizTalk Services – February 2014 release.

Introduction

Electronic Data Interchange (EDI) is one of the most prevalent means by which businesses exchange data electronically, also termed as Business-to-Business or B2B transactions. BizTalk Server has had EDI support for over a decade, since the initial BizTalk Server release. With BizTalk Services, Microsoft continues the support for EDI solutions on the Microsoft Azure platform. B2B transactions are mostly external to an organization, and hence it's easier to implement if it was implemented on a cloud platform. Microsoft Azure provides this capability through BizTalk Services.

While some customers look at BizTalk Services as a "greenfield" platform for new EDI solutions, many customers have current BizTalk Server EDI solutions that they may want to migrate to Azure. Because BizTalk Services EDI is architected based on the same key entities as BizTalk Server EDI architecture (trading partners, entities, agreements), it is possible to migrate BizTalk Server EDI artifacts to BizTalk Services.

This document discusses some of the differences involved with migrating BizTalk Server EDI artifacts to BizTalk Services. This document assumes a working knowledge of BizTalk Server EDI processing and Trading Partner Agreements. For more information on BizTalk Server EDI, see [Trading Partner Management Using BizTalk Server](#).

Which version of BizTalk Server EDI artifacts can be migrated to BizTalk Services?

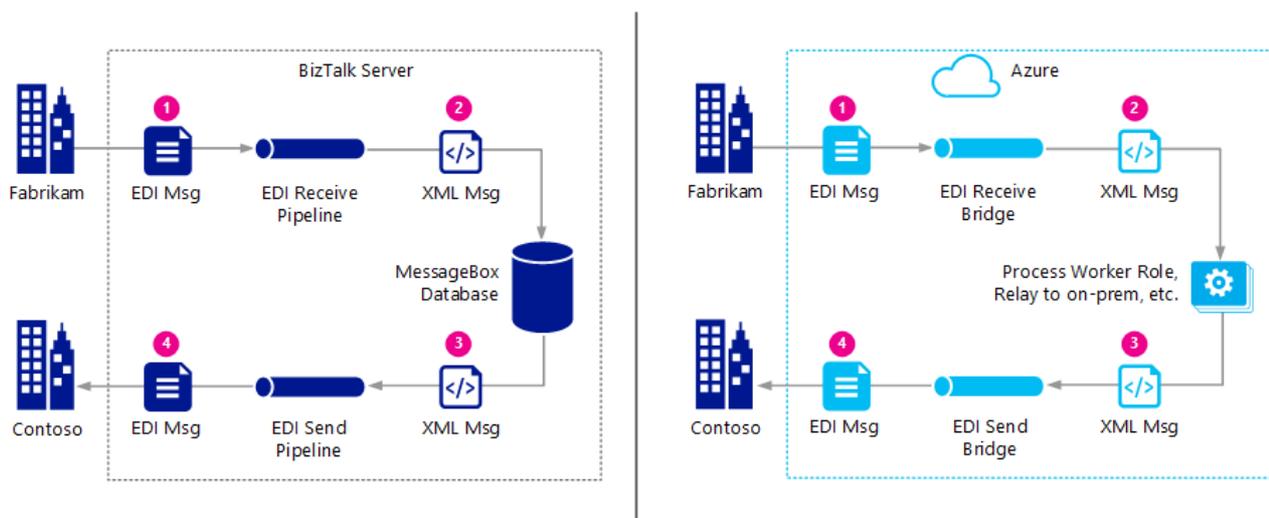
The BizTalk Server EDI module was significantly enhanced for BizTalk Server 2010, when it was remodeled to include partners, profiles, and agreements. BizTalk Services uses the same model to organize the trading partners and the business divisions within those trading partners. As a result, migrating EDI artifacts from BizTalk Server 2010 and later versions to BizTalk Services, is a much more straight forward process. To migrate EDI artifacts associated with versions prior to BizTalk Server 2010, you must first upgrade to BizTalk Server 2010 and then migrate your EDI artifacts to BizTalk Services.

Scenarios/Message Flow

As with BizTalk Server, EDI processing in BizTalk Services is built around a Trading Partner Management (TPM) solution. The TPM solution has the following key components:

- Trading partners, which represent organization in a B2B transaction.
- Profiles, which represent divisions within a trading partner.
- Trading partner agreements (or agreements), which represent the business agreement between two partners/profiles.

The following illustration depicts the similarities, as well as differences, between a BizTalk Server EDI solution and BizTalk Services EDI solution:



The key differences, and similarities, between an EDI solution flow in BizTalk Server and BizTalk Services are:

- Just like BizTalk Server uses an EDIReceive pipeline to receive an EDI message and an EDISend pipeline to send an EDI message, BizTalk Services uses an EDI Receive bridge to receive and EDI Send bridge to send out EDI messages. In BizTalk Server, the pipelines are associated with an agreement by using send or receive ports. In BizTalk Services, the agreement itself denotes the send or receive bridge.
- In BizTalk Server, after the EDIReceive pipeline processes the EDI message, the message is dumped to a SQL Server database. The EDISend pipeline then picks up the message from the SQL Server database, processes it, and then sends it out to the trading partner.

In BizTalk Services, after the EDI receive bridge processes the EDI message, it routes the message to an external process. The external process could be running on Microsoft Azure or on-premises. The external process should route the message to the EDI send bridge; the send bridge does not inherently pull the message. After processing the message, the EDI send bridge routes the message to the trading partner.

BizTalk Services provides an easy-to-use configuration experience to quickly create and deploy a B2B agreement between trading partners without configuring any Microsoft Azure Compute instances (Web or Worker roles), any Microsoft Azure SQL Databases, or any Microsoft Azure storage accounts. More complex scenarios will require tying in workflows or other service processing "around the edges" of a Trading Partner agreement, that is, before or after Trading Partner Agreement EDI bridge processing. In detail, the following sequences of events occur during an EDI message processing in BizTalk Services.

1. An EDI message is received from trading partner, Fabrikam. For receiving EDI messages from trading partners, BizTalk Services supports transport protocols such as FTP, SFTP, AS2, and HTTP/S.
2. The trading partner agreement receive-side processing disassembles the EDI message to XML format. You can route the disassembled EDI message (in XML format) to Service Bus endpoints such as a Service Bus Relay endpoint, Service Bus Topic, Service Bus Queue, or a BizTalk Services bridge.
3. The disassembled XML messages could then be received from the endpoint for further custom processing.

These endpoints could be processed by an on-premises component or a Microsoft Azure Compute instance to further process the message in a Windows Workflow (WF) or Windows Communication Foundation (WCF) service, for example.

4. The "send-side processing" of the trading partner agreement then assembles the XML message into EDI format and sends it to trading partner, Contoso. For sending EDI messages to trading partners, BizTalk Services supports the same protocols as those used for receiving EDI messages.

This document further provides conceptual guidance on migrating some of the different BizTalk Server EDI artifacts to BizTalk Services.

Send/Receive Ports to Trading Partners

In BizTalk Server you set up Receive Locations and Receive Ports to receive EDI/XML messages from trading partners, and you set up Send Ports to send EDI/XML messages to trading partner. You then tie up these ports to a trading partner agreement using the BizTalk Server Administration console. In BizTalk Services, the locations where you receive messages from trading partners and where you send messages to trading partners are configured as part of the trading partner agreement itself (as part of Transport Settings) in the BizTalk Services Portal. So you do not really have the concept of "send ports" and "receive locations", per se, in BizTalk Services. For more information, see [Creating Agreements](#).

Pipelines (Bridges)

In BizTalk Server EDI, pipelines are message processing entities that can also include custom logic for specific processing capabilities, as required by the application. For BizTalk Services, the equivalent would be an EDI bridge. However in BizTalk Services, for now, the EDI bridges are "closed". That is, you cannot add your own custom activities to an EDI bridge. Any custom processing must be done outside the EDI bridge in your application, either before or after the message enters the bridge configured as part of the Trading Partner agreement. EAI bridges have the option to do custom processing. If you want custom processing, you can use EAI bridges before or after the message is processed by the EDI bridge. For more information, see [How to Include Custom Code in Bridges](#).

You can insert a publish/subscribe flow with custom code and/or using Service Bus messaging Queues and Topics before the trading partner agreement receives the message, or after the agreement processes the message and routes it to a Service Bus endpoint.

See **Scenarios/Message Flow** in this topic for the message flow pattern.

Agreements

If you are familiar with the BizTalk Server 2010 Trading Partner Agreements used for EDI processing, then BizTalk Services trading partner agreements look very familiar. Most of the agreement settings are the same and use the same terminology. In some cases, the agreement settings are much simpler compared to the same settings in BizTalk Server. Microsoft Azure BizTalk Services supports X12, EDIFACT, and AS2 transport.

Microsoft Azure BizTalk Services also provides a **TPM Data Migration** tool to migrate trading partners and agreements from BizTalk Server Trading Partner module to BizTalk Services Portal. The TPM Data Migration tool is available as part of a Tools package, which can be downloaded from the [MABS SDK](#). The package also includes a readme that provides instructions on how to use the tool, and basic troubleshooting information for the tool.

Schemas

BizTalk Services provides EDI schemas which can be used in BizTalk Services solutions. In addition, BizTalk Server EDI schemas can also be used with BizTalk Services because the root node of the EDI schema is the same across BizTalk Server as well as BizTalk Services. Thus, you will be able to directly take your BizTalk Server EDI schemas and use them in the EDI solutions that you develop using BizTalk Services. You can also download the schemas from the

Maps (Transforms)

Maps in BizTalk Server are called Transforms in BizTalk Services. Migrating maps from BizTalk Server to BizTalk Services could be one of the more complex tasks to achieve (depending on map complexity). The mapping tool used for BizTalk Services is different from the BizTalk mapper. Even though the mapper looks mostly the same, the underlying map format is different. The functoids (called **Map Operations** in BizTalk Services) available to the users are different as well. In effect, you cannot directly use a BizTalk map in BizTalk Services. Also, not all the functoids available in BizTalk Server are available as map operations in BizTalk Services.

New Transform Operations

While the list of Transform map operations available may seem quite different from the BizTalk Server mapper, BizTalk Services Transforms have new ways of accomplishing the same tasks. For example, BizTalk Services Transforms have **List Operations** available. This was not available in the BizTalk mapper. The **List Operations** enable you to create and operate on a "List", where a list is a set of items (also known as "rows") and where each item can have multiple members (also known as "columns"). You can sort the list, select items based on a condition, etc.

Another example of new functionality in BizTalk Services Transforms are the **Loop Operations**. It is difficult to create nested loops in the BizTalk Server mapper. Thus, the Loop map operations are added for the BizTalk Services Transforms.

Yet another example is the **If-Then-Else** Expression map operation. Doing an if-then-else operation was possible in the BizTalk mapper, but it required multiple functoids to accomplish a seemingly simple task.

Migrating BizTalk Server Maps

Microsoft Azure BizTalk Services provides a tool to migrate BizTalk Server maps to BizTalk Services transforms. The **BTMMigrationTool** is available as part of the **Tools** package provided with the [BizTalk Services SDK download](#). For more information about the tool, see [Convert a BizTalk map to a BizTalk Services Transform](#).

You can also look at a sample by Sandro Pereira, BizTalk MVP, on how to [migrate BizTalk Server maps to BizTalk Services transforms](#).

Orchestrations

If you need to migrate BizTalk Server orchestration processing to Microsoft Azure, the orchestrations would need to be rewritten because Microsoft Azure does not support running BizTalk Server orchestrations. You could rewrite the orchestration functionality in a Windows Workflow Foundation 4.0 (WF4) service. This would be a complete rewrite as there is currently no migration from BizTalk Server orchestrations to WF4. Here are some resources for Windows Workflow:

- [How to integrate a WCF Workflow Service with Service Bus Queues and Topics](#) by Paolo Salvatori.
- [Building apps with Windows Workflow Foundation and Azure session](#) from the Build 2011 conference.
- [Windows Workflow Foundation Developer Center](#) on MSDN.
- [Windows Workflow Foundation 4 \(WF4\) documentation](#) on MSDN.

Other Considerations

Following are a few considerations that you must make while using BizTalk Services.

Fallback Agreements

BizTalk Server EDI processing has the concept of "Fallback Agreements". BizTalk Services does **not** have a Fallback Agreement concept so far. See BizTalk documentation topics [The Role of Agreements in EDI Processing](#) and [Configuring Global or Fallback Agreement Properties](#) for information on how Fallback Agreements are used in

BizTalk Server.

Routing to multiple destinations

BizTalk Services bridges, in its current state does not support routing messages to multiple destinations using a publish-subscribe model. Instead you could route messages from a BizTalk Services bridge to a Service Bus topic, which can then have multiple subscriptions to receive the message at more than one endpoint.

Conclusion

Microsoft Azure BizTalk Services is updated at regular milestones to add more features and capabilities. With each update, we look forward to supporting increased functionality to facilitate creating end-to-end solutions using BizTalk Services and other Azure technologies.

See Also

[Developing Enterprise Applications with Azure](#)

What you can and can't do using the BizTalk Service state

6/27/2017 • 1 min to read • [Edit Online](#)

IMPORTANT

Microsoft Azure BizTalk Services (MABS) is being retired, and replaced with Azure Logic Apps. If you currently use MABS, then [Move from BizTalk Services to Logic Apps](#) provides some guidance on moving your integration solutions to Logic Apps.

If you're brand new to Logic Apps, then we suggest getting started here:

- [What are Logic Apps](#)
- [Create your first logic app](#), or quickly get started using a [pre-built template](#)
- [View all the available connectors](#) you can use in a logic app

Depending on the current state of the BizTalk service, there are operations that you can or cannot perform on the BizTalk service.

For example, you provision a new BizTalk service in the Azure classic portal. When it completes successfully, the BizTalk service is in `active` state. In the active state, you can stop, suspend, and delete the BizTalk service. If you stop the BizTalk service, and stop fails, then the BizTalk service goes to a `StopFailed` state. In the `StopFailed` state, you can restart the BizTalk service. If you try an operation that is not allowed, like resume, the following error occurs:

```
Operation not allowed
```

View the possible states

The following tables list the operations or actions that can be done when the BizTalk Service is in a specific state. A ✓ means the operation is allowed while in that state. A blank entry means the operation cannot be performed while in that state.

SERVICE STATE	START	STOP	RESTART	SUSPEND	RESUME	DELETE	SCALE	UPDATE CONFIGURATION	BACKUP
Active		✓	✓	✓		✓	✓	✓	✓
Disabled						✓			
Suspended					✓	✓			✓
Stopped	✓		✓			✓			✓
Service Update Failed						✓			

SERVICE STATE	START	STOP	RESTART	SUSPENDED	RESUME	DELETE	SCALE	UPDATE CONFIGURATION	BACKUP
DisableFailed						✓			
EnableFailed						✓			
StartFailed StopFailed RestartFailed	✓	✓	✓			✓		✓	
SuspendedFailed ResumeFailed				✓	✓	✓			
CreatedFailed RestoreFailed						✓			
ConfigUpdateFailed			✓			✓		✓	
ScaleFailed						✓	✓		

See Also

- [Create a BizTalk Service using the Azure classic portal](#)
- [What you can do in the dashboard, monitor and scale tabs in BizTalk Services](#)
- [What you get with the Developer, Basic, Standard, and Premium editions in BizTalk Services](#)
- [How to back up and restore a BizTalk Service](#)
- [Throttling explained in BizTalk Services](#)
- [Retrieve the Service Bus and Access Control issuer name and issuer key values for your BizTalk Service](#)
- [How do I Start Using the Azure BizTalk Services SDK](#)

BizTalk Services: Issuer Name and Issuer Key

6/27/2017 • 2 min to read • [Edit Online](#)

IMPORTANT

Microsoft Azure BizTalk Services (MABS) is being retired, and replaced with Azure Logic Apps. If you currently use MABS, then [Move from BizTalk Services to Logic Apps](#) provides some guidance on moving your integration solutions to Logic Apps.

If you're brand new to Logic Apps, then we suggest getting started here:

- [What are Logic Apps](#)
- [Create your first logic app](#), or quickly get started using a [pre-built template](#)
- [View all the available connectors](#) you can use in a logic app

Azure BizTalk Services uses the Service Bus Issuer Name and Issuer Key, and the Access Control Issuer Name and Issuer Key. Specifically:

TASK	WHICH ISSUER NAME AND ISSUER KEY
Deploying your application from Visual Studio	Access Control Issuer Name and Issuer Key
Configuring the Azure BizTalk Services Portal	Access Control Issuer Name and Issuer Key
Creating LOB Relays with the BizTalk Adapter Services in Visual Studio	Service Bus Issuer Name and Issuer Key

This topic lists the steps to retrieve the Issuer Name and Issuer Key.

Access Control Issuer Name and Issuer Key

The Access Control Issuer Name and Issuer Key are used by the following:

- Your Azure BizTalk Service application created in Visual Studio: To successfully deploy your BizTalk Service application in Visual Studio to Azure, you enter the Access Control Issuer Name and Issuer Key.
- The Azure BizTalk Services Portal: When you create a BizTalk Service and open the BizTalk Services Portal, your Access Control Issuer Name and Issuer Key are automatically registered for your deployments with the same Access Control values.

Get the Access Control Issuer Name and Issuer Key

To use ACS for authentication, and get the Issuer Name and Issuer Key values, the overall steps include:

1. Install the [Azure Powershell cmdlets](#).
2. Add your Azure account: `Add-AzureAccount`
3. Return your subscription name: `get-azuresubscription`
4. Select your subscription: `select-azuresubscription <name of your subscription>`

5. Create a new namespace:

```
new-azuresbnamespace <name for the service bus> "Location" -CreateACSNamespace $true -NamespaceType Messaging
```

Example:

```
new-azuresbnamespace biztalksbnamespace "South Central US" -CreateACSNamespace $true -NamespaceType Messaging
```

6. When the new ACS namespace is created (which can take several minutes), the Issuer Name and Issuer Key values are listed in the connection string:

```
Name : biztalksbnamespace
Region : South Central US
DefaultKey : abcdefghijklmnopqrstuvwxyz
Status : Active
CreatedAt : 10/18/2016 9:36:30 PM
AcsManagementEndpoint : https://biztalksbnamespace-sb.accesscontrol.windows.net/
ServiceBusEndpoint : https://biztalksbnamespace.servicebus.windows.net/
ConnectionString :
Endpoint=sb://biztalksbnamespace.servicebus.windows.net/;SharedSecretIssuer=owner;SharedSecretValue=abc
defghijklmnopqrstuvwxyz
NamespaceType : Messaging
```

To summarize:

Issuer Name = SharedSecretIssuer

Issuer Key = SharedSecretKey

More on the [New-AzureSBNamespace](#) cmdlet.

Service Bus Issuer Name and Issuer Key

Service Bus Issuer Name and Issuer Key are used by BizTalk Adapter Services. In your BizTalk Services project in Visual Studio, you use the BizTalk Adapter Services to connect to an on-premises Line-of-Business (LOB) system. To connect, you create the LOB Relay and enter your LOB system details. When doing this, you also enter the Service Bus Issuer Name and Issuer Key.

To retrieve the Service Bus Issuer Name and Issuer Key

1. Sign in to the [Azure classic portal](#).
2. In the left navigation pane, select **Service Bus**.
3. Select your namespace. In the task bar, select **Connection Information**. This displays the **Default Issuer** (Issuer Name) and **Default Key** (Issuer Key). Their values can be copied.

To summarize:

Issuer Name = Default Issuer

Issuer Key = Default Key

Next

Additional Azure BizTalk Services topics:

- [Installing the Azure BizTalk Services SDK](#)
- [Tutorials: Azure BizTalk Services](#)
- [How do I Start Using the Azure BizTalk Services SDK](#)
- [Azure BizTalk Services](#)

See Also

- [How to: Use ACS Management Service to Configure Service Identities](#)
- [BizTalk Services: Developer, Basic, Standard and Premium Editions Chart](#)
- [BizTalk Services: Provisioning Using Azure classic portal](#)

- [BizTalk Services: Provisioning Status Chart](#)
- [BizTalk Services: Dashboard, Monitor and Scale tabs](#)
- [BizTalk Services: Backup and Restore](#)
- [BizTalk Services: Throttling](#)

BizTalk Services: Troubleshoot using operation logs

6/27/2017 • 2 min to read • [Edit Online](#)

IMPORTANT

Microsoft Azure BizTalk Services (MABS) is being retired, and replaced with Azure Logic Apps. If you currently use MABS, then [Move from BizTalk Services to Logic Apps](#) provides some guidance on moving your integration solutions to Logic Apps.

If you're brand new to Logic Apps, then we suggest getting started here:

- [What are Logic Apps](#)
- [Create your first logic app](#), or quickly get started using a [pre-built template](#)
- [View all the available connectors](#) you can use in a logic app

What are the Operation Logs

Operation Logs is a Management Services feature available in the Azure classic portal that allows you to view historical logs of operations performed on your Azure services, including BizTalk Services. This enables you to view historical data related to management operations on your BizTalk Service subscription as far back as 180 days.

NOTE

This feature only captures logs for management operations on BizTalk Services, such as when the service was started, backed up, and so on. Such operations are tracked irrespective of whether they are performed from the Azure classic portal or by using the [BizTalk Service REST APIs](#). For a complete list of operations that are tracked using Management Services, see [Operations Tracked Using Azure Management Services](#).

This does not capture the logs for activities related to BizTalk Service runtime (such as message processed by bridges, and so on.). To view these logs, use the Tracking view from the BizTalk Services portal. For more information, see [Tracking Messages](#).

View BizTalk Services Operation Logs

1. In the Azure classic portal, select **Management Services**, and then select the **Operation Logs** tab.
2. You can filter the logs based on different parameters like subscription, date range, service type (e.g. BizTalk Services), service name, or status of the operation (Succeeded, Failed).
3. Select the checkmark to view the filtered list. The following image shows activities related to testbiztalkservice:

SUBSCRIPTION FROM 2014-2-04 5:00 AM TO 2014-2-04 1:00 PM

STATUS TYPE SERVICE NAME

Click the checkmark button to execute the query.

TIME STAMP	OPERATION	STATUS	SERVICE NAME	TYPE	CALLER	OPERATIO...
2/4/2014 10:59:54 AM	CreateBizTalkService	Started	testbiztalkservice	BizTalk Services	Not Available	9cc735a8-b3e2-c...
2/4/2014 11:15:51 AM	CreateBizTalkService	Succeeded	testbiztalkservice	BizTalk Services	Not Available	9cc735a8-b3e2-c...
2/4/2014 11:24:48 AM	ConfigUpdateBizTalkService	Started	testbiztalkservice	BizTalk Services	Not Available	14ff8e83-1f3e-c2...
2/4/2014 11:25:14 AM	ConfigUpdateBizTalkService	Succeeded	testbiztalkservice	BizTalk Services	Not Available	14ff8e83-1f3e-c2...
2/4/2014 12:00:01 PM	BackupBizTalkService	Started	testbiztalkservice	BizTalk Services	Automatic	4df130e3-9e70-4...
2/4/2014 12:00:01 PM	PurgeBackupBizTalkService	Started	testbiztalkservice	BizTalk Services	Automatic	1f94832f-3784-4...
2/4/2014 12:20:59 PM	BackupBizTalkService	Succeeded	testbiztalkservice	BizTalk Services	Automatic	4df130e3-9e70-4...

DETAILS

4. To view more about a specific operation, select the row, and click **Details** in the task bar at the bottom.

Operations Tracked Using Azure Management Services

The following table lists the operations that are tracked using the Azure Management Services:

OPERATION NAME	TASK
CreateBizTalkService	Operation to create a new BizTalk Service
DeleteBizTalkService	Operation to delete a BizTalk Service
RestartBizTalkService	Operation to restart a BizTalk Service
StartBizTalkService	Operation to start a BizTalk Service
StopBizTalkService	Operation to stop a BizTalk Service
DisableBizTalkService	Operation to disable a BizTalk Service
EnableBizTalkService	Operation to enable a BizTalk Service
BackupBizTalkService	Operation to back up a BizTalk Service
RestoreBizTalkService	Operation to restore a BizTalk Service from specified backup
SuspendBizTalkService	Operation to suspend a BizTalk Service
ResumeBizTalkService	Operation to resume a BizTalk Service
ScaleBizTalkService	Operation to scale a BizTalk Service up or down
ConfigUpdateBizTalkService	Operation to update the configuration of a BizTalk Service

OPERATION NAME	TASK
ServiceUpdateBizTalkService	Operation to upgrade or downgrade a BizTalk Service to a different version
PurgeBackupBizTalkService	Operation to purge backups of the BizTalk Service outside the retention period

See Also

- [Backup BizTalk Service](#)
- [Restore BizTalk Service from Backup](#)
- [BizTalk Services: Developer, Basic, Standard and Premium Editions Chart](#)
- [BizTalk Services: Provisioning Using Azure classic portal](#)
- [BizTalk Services: Provisioning Status Chart](#)
- [BizTalk Services: Dashboard, Monitor and Scale tabs](#)
- [BizTalk Services: Throttling](#)
- [BizTalk Services: Issuer Name and Issuer Key](#)
- [How do I Start Using the Azure BizTalk Services SDK](#)

Hybrid Connections overview

6/27/2017 • 4 min to read • [Edit Online](#)

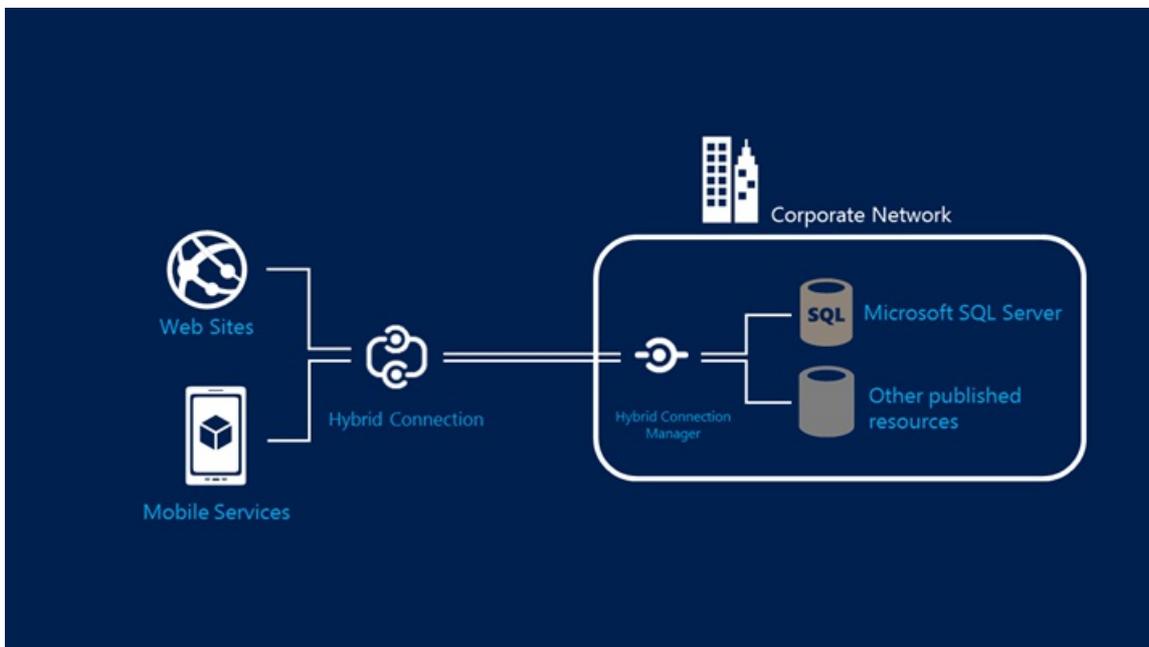
IMPORTANT

BizTalk Hybrid Connections is retired, and replaced by App Service Hybrid Connections. For more information, including how to manage your existing BizTalk Hybrid Connections, see [Azure App Service Hybrid Connections](#).

Introduction to Hybrid Connections, lists the supported configurations, and lists the required TCP ports.

What is a hybrid connection

Hybrid Connections are a feature of Azure BizTalk Services. Hybrid Connections provide an easy and convenient way to connect the Web Apps feature in Azure App Service (formerly Websites) and the Mobile Apps feature in Azure App Service (formerly Mobile Services) to on-premises resources behind your firewall.



Hybrid Connections benefits include:

- Web Apps and Mobile Apps can access existing on-premises data and services securely.
- Multiple Web Apps or Mobile Apps can share a Hybrid Connection to access an on-premises resource.
- Minimal TCP ports are required to access your network.
- Applications using Hybrid Connections access only the specific on-premises resource that is published through the Hybrid Connection.
- Can connect to any on-premises resource that uses a static TCP port, such as SQL Server, MySQL, HTTP Web APIs, and most custom Web Services.

NOTE

TCP-based services that use dynamic ports (such as FTP Passive Mode or Extended Passive Mode) are currently not supported. LDAP is also not supported. LDAP uses a static TCP port but it could also use UDP. As a result, it is not supported.

- Can be used with all frameworks supported by Web Apps (.NET, PHP, Java, Python, Node.js) and Mobile Apps (Node.js, .NET).
- Web Apps and Mobile Apps can access on-premises resources in exactly the same way as if the Web or Mobile App is located on your local network. For example, the same connection string used on-premises can also be used on Azure.

Hybrid Connections also provide enterprise administrators with control and visibility into the corporate resources accessed by hybrid applications, including:

- Using Group Policy settings, administrators can allow Hybrid Connections on the network and also designate resources that can be accessed by hybrid applications.
- Event and audit logs on the corporate network provide visibility into the resources accessed by Hybrid Connections.

Example scenarios

Hybrid Connections support the following framework and application combinations:

- .NET framework access to SQL Server
- .NET framework access to HTTP/HTTPS services with WebClient
- PHP access to SQL Server, MySQL
- Java access to SQL Server, MySQL and Oracle
- Java access to HTTP/HTTPS services

When using Hybrid Connections to access on-premises SQL Server, consider the following:

- SQL Express Named Instances must be configured to use static ports. By default, SQL Express named instances use dynamic ports.
- SQL Express Default Instances use a static port, but TCP must be enabled. By default, TCP is not enabled.
- When using Clustering or Availability Groups, the `MultiSubnetFailover=true` mode is currently not supported.
- The `ApplicationIntent=ReadOnly` is currently not supported.
- SQL Authentication may be required as the end-to-end authorization method supported by the Azure application and the on-premises SQL server.

Security and ports

Hybrid Connections use Shared Access Signature (SAS) authorization to secure the connections from the Azure applications and the on-premises Hybrid Connection Manager to the Hybrid Connection. Separate connection keys are created for the application and the on-premises Hybrid Connection Manager. These connection keys can be rolled over and revoked independently.

Hybrid Connections provide for seamless and secure distribution of the keys to the applications and the on-premises Hybrid Connection Manager.

See [Create and Manage Hybrid Connections](#).

Application authorization is separate from the Hybrid Connection. Any appropriate authorization method can be used. The authorization method depends on the end-to-end authorization methods supported across the Azure cloud and the on-premises components. For example, your Azure application accesses an on-premises SQL Server. In this scenario, SQL Authorization may be the authorization method that is supported end-to-end.

TCP ports

Hybrid Connections require only outbound TCP or HTTP connectivity from your private network. You do not need to open any firewall ports or change your network perimeter configuration to allow any inbound connectivity into your network.

The following TCP ports are used by Hybrid Connections:

PORT	WHY YOU NEED IT
9350 - 9354	<p>These ports are used for data transmission. The Service Bus relay manager probes port 9350 to determine if TCP connectivity is available. If it is available, then it assumes that port 9352 is also available. Data traffic goes over port 9352.</p> <p>Allow outbound connections to these ports.</p>
5671	<p>When port 9352 is used for data traffic, port 5671 is used as the control channel.</p> <p>Allow outbound connections to this port.</p>
80, 443	<p>These ports are used for some data requests to Azure. Also, if ports 9352 and 5671 are not usable, <i>then</i> ports 80 and 443 are the fallback ports used for data transmission and the control channel.</p> <p>Allow outbound connections to these ports.</p> <p>Note It is not recommended to use these as the fallback ports in place of the other TCP ports. The HTTP/WebSocket is used as the protocol instead of native TCP for data channels. It could result in lower performance.</p>

Next steps

[Create and manage Hybrid Connections](#)

[Connect Azure Web Apps to an On-Premises Resource](#)

[Connect to on-premises SQL Server from an Azure web app](#)

See Also

[REST API for Managing BizTalk Services on Microsoft Azure BizTalk Services: Editions Chart](#)

[Create a BizTalk Service using Azure portal](#)

[BizTalk Services: Dashboard, Monitor and Scale tabs](#)

Create and Manage Hybrid Connections

6/27/2017 • 5 min to read • [Edit Online](#)

IMPORTANT

BizTalk Hybrid Connections is retired, and replaced by App Service Hybrid Connections. For more information, including how to manage your existing BizTalk Hybrid Connections, see [Azure App Service Hybrid Connections](#).

Overview of the Steps

1. Create a Hybrid Connection by entering the **host name** or **FQDN** of the on-premises resource in your private network.
2. Link your Azure web apps or Azure mobile apps to the Hybrid Connection.
3. Install the Hybrid Connection Manager on your on-premises resource and connect to the specific Hybrid Connection. The Azure portal provides a single-click experience to install and connect.
4. Manage Hybrid Connections and their connection keys.

This topic lists these steps.

IMPORTANT

It is possible to set a Hybrid Connection endpoint to an IP address. If you use an IP address, you may or may not reach the on-premises resource, depending on your client. The Hybrid Connection depends on the client doing a DNS lookup. In most cases, the **client** is your application code. If the client does not perform a DNS lookup, (it does not try to resolve the IP address as if it were a domain name (x.x.x.x)), then traffic is not sent through the Hybrid Connection.

For example (pseudocode), you define **10.4.5.6** as your on-premises host:

The following scenario works:

```
Application code -> GetHostByName("10.4.5.6") -> Resolves to 127.0.0.3 -> Connect("127.0.0.3") -> Hybrid Connection -> on-prem host
```

The following scenario doesn't work:

```
Application code -> Connect("10.4.5.6") -> ?? -> No route to host
```

Create a Hybrid Connection

A Hybrid Connection can be created in the Azure portal using Web Apps **or** using BizTalk Services.

To create Hybrid Connections using Web Apps, see [Connect Azure Web Apps to an On-Premises Resource](#). You can also install the Hybrid Connection Manager (HCM) from your web app, which is the preferred method.

To create Hybrid Connections in BizTalk Services:

1. Sign in to the [Azure classic portal](#).
2. In the left navigation pane, select **BizTalk Services** and then select your BizTalk Service.

If you don't have an existing BizTalk Service, you can [Create a BizTalk Service](#).

3. Select the **Hybrid Connections** tab:



4. Select **Create a Hybrid Connection** or select the **ADD** button in the task bar. Enter the following:

PROPERTY	DESCRIPTION
Name	The Hybrid Connection name must be unique and cannot be the same name as the BizTalk Service. You can enter any name but be specific with its purpose. Examples include: <i>PayrollSQLServer</i> <i>SupplyListSharepointServer</i> <i>CustomersOracleServer</i>
Host Name	Enter the fully qualified host name, only the host name, or the IPv4 address of the on-premises resource. Examples include: <i>mySQLServer</i> <i>mySQLServer.Domain.corp.yourCompany.com</i> <i>myHTTPSharePointServer</i> <i>myHTTPSharePointServer.yourCompany.com</i> <i>10.100.10.10</i> If you use the IPv4 address, note that your client or application code may not resolve the IP address. See the Important note at the top of this topic.
Port	Enter the port number of the on-premises resource. For example, if you're using Web Apps, enter port 80 or port 443. If you're using SQL Server, enter port 1433.

5. Select the check mark to complete the setup.

Additional

- Multiple Hybrid Connections can be created. See the [BizTalk Services: Editions Chart](#) for the number of connections allowed.
- Each Hybrid Connection is created with a pair of connection strings: Application keys that SEND and On-premises keys that LISTEN. Each pair has a Primary and a Secondary key.

Link your Azure App Service Web App or Mobile App

To link a Web App or Mobile App in Azure App Service to an existing Hybrid Connection, select **use an existing Hybrid Connection** in the Hybrid Connections blade. See [Access on-premises resources using hybrid connections in Azure App Service](#).

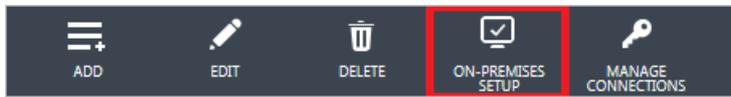
Install the Hybrid Connection Manager on-premises

After a Hybrid Connection is created, install the Hybrid Connection Manager on the on-premises resource. It can be downloaded from your Azure web apps or from your BizTalk Service. BizTalk Services steps:

1. Sign in to the [Azure classic portal](#).
2. In the left navigation pane, select **BizTalk Services** and then select your BizTalk Service.
3. Select the **Hybrid Connections** tab:



4. In the task bar, select **On-Premises Setup**:



5. Select **Install and Configure** to run or download the Hybrid Connection Manager on the on-premises system.
6. Select the check mark to start the installation.

Additional

- Hybrid Connection Manager can be installed on the following operating systems:
 - Windows Server 2008 R2 (.NET Framework 4.5+ and Windows Management Framework 4.0+ required)
 - Windows Server 2012 (Windows Management Framework 4.0+ required)
 - Windows Server 2012 R2
- After you install the Hybrid Connection Manager, the following occurs:
 - The Hybrid Connection hosted on Azure is automatically configured to use the Primary Application Connection String.
 - The On-Premises resource is automatically configured to use the Primary On-Premises Connection String.
- The Hybrid Connection Manager must use a valid on-premises connection string for authorization. The Azure Web Apps or Mobile Apps must use a valid application connection string for authorization.
- You can scale Hybrid Connections by installing another instance of the Hybrid Connection Manager on another server. Configure the on-premises listener to use the same address as the first on-premises listener. In this situation, the traffic is randomly distributed (round robin) between the active on-premises listeners.

Manage Hybrid Connections

To manage your Hybrid Connections, you can:

- Use the Azure portal and go to your BizTalk Service.
- Use [REST APIs](#).

Copy/regenerate the Hybrid Connection Strings

1. Sign in to the [Azure classic portal](#).
2. In the left navigation pane, select **BizTalk Services** and then select your BizTalk Service.
3. Select the **Hybrid Connections** tab:



4. Select the Hybrid Connection. In the task bar, select **Manage Connection**:



Manage Connection lists the Application and On-Premises connection strings. You can copy the Connection Strings or regenerate the Access Key used in the connection string.

If you select Regenerate, the Shared Access Key used within the Connection String is changed. Do the following:

- In the Azure classic portal, select **Sync Keys** in the Azure application.
- Re-run the **On-Premises Setup**. When you re-run the On-Premises Setup, the on-premises resource is automatically configured to use the updated Primary connection string.

Use Group Policy to control the on-premises resources used by a Hybrid Connection

1. Download the [Hybrid Connection Manager Administrative Templates](#).
2. Extract the files.

3. On the computer that modifies group policy, do the following:

- Copy the .ADMX files to the %WINROOT%\PolicyDefinitions folder.
- Copy the .ADML files to the %WINROOT%\PolicyDefinitions\en-us folder.

Once copied, you can use Group Policy Editor to change the policy.

Next

[Connect Azure Web Apps to an On-Premises Resource](#)

[Connect to on-premises SQL Server from Azure Web Apps](#)

[Hybrid Connections Overview](#)

See Also

[REST API for Managing BizTalk Services on Microsoft Azure](#)

[BizTalk Services: Editions Chart](#)

[Create a BizTalk Service using Azure classic portal](#)

[BizTalk Services: Dashboard, Monitor and Scale tabs](#)

Release Notes for Azure BizTalk Services

6/27/2017 • 14 min to read • [Edit Online](#)

IMPORTANT

Microsoft Azure BizTalk Services (MABS) is being retired, and replaced with Azure Logic Apps. If you currently use MABS, then [Move from BizTalk Services to Logic Apps](#) provides some guidance on moving your integration solutions to Logic Apps.

If you're brand new to Logic Apps, then we suggest getting started here:

- [What are Logic Apps](#)
- [Create your first logic app, or quickly get started using a pre-built template](#)
- [View all the available connectors](#) you can use in a logic app

The release notes for the Microsoft Azure BizTalk Services contain the known issues in this release.

What's new in the November update of BizTalk Services

- Encryption at Rest can be enabled in the BizTalk Services Portal. See [Enable Encryption at Rest in BizTalk Services Portal](#).

Update History

October Update

- Organizational accounts are supported:
 - **Scenario:** You registered a BizTalk Service deployment using a Microsoft account (like user@live.com). In this scenario, only Microsoft Account users can manage the BizTalk Service using the BizTalk Services portal. An organizational account cannot be used.
 - **Scenario:** You registered a BizTalk Service deployment using an organizational account in an Azure Active Directory (like user@fabrikam.com or user@contoso.com). In this scenario, only Azure Active Directory users within the same organization can manage the BizTalk Service using the BizTalk Services portal. A Microsoft account cannot be used.
- When you create a BizTalk Service in the Azure classic portal, you are automatically registered in the BizTalk Services Portal.
 - **Scenario:** You sign into the Azure classic portal, create a BizTalk Service, and then select **Manage** for the very first time. When the BizTalk Services portal opens, the BizTalk Service automatically registers and is ready for your deployments.
See [Registering and Updating a BizTalk Service Deployment on the BizTalk Services Portal](#).

August 14 Update

- Agreement and bridge decoupling – Trading partner agreements and bridges are now decoupled in the BizTalk Services Portal. You now create agreements and bridges separately, and at runtime bridges resolve to an agreement based on the values in the EDI message. See [Create Agreements in Azure BizTalk Services](#), [Create an EDI bridge using BizTalk Services Portal](#), [Create an AS2 bridge using BizTalk Services Portal](#), and [How do bridges resolve agreements at runtime?](#)
- The option to create templates for agreements is discontinued.

- For the send-side agreement, you can now specify different delimiter sets for each schema. This configuration is specified under protocol settings for send side agreement. For more information, see [Create an X12 Agreement in Azure BizTalk Services](#) and [Create an EDIFACT Agreement in Azure BizTalk Services](#). Two new entities are also added to the TPM OM API for the same purpose. See [X12DelimiterOverrides](#) and [EDIFACTDelimiterOverride](#).
- Standard XSD constructs, including Derived Types, are now supported. See [Use standard XSD constructs in your maps](#) and [Use Derived Types in Mapping Scenarios and Examples](#).
- AS2 supports new MIC algorithms for message signing and new encryption algorithms. See [Create an AS2 Agreement in Azure BizTalk Services](#).

Know Issues

Connectivity Issues after BizTalk Services Portal Update

If you have the BizTalk Services Portal open while BizTalk Services is upgraded to roll in changes to the service, you might face connectivity issues with the BizTalk Services Portal.

As a workaround, you may restart the browser, delete the browser cache, or start the portal in private mode.

Visual Studio IDE cannot locate the artifact if you click an error or warning in a BizTalk Services project

Install the Visual Studio 2012 Update 3 RC 1 to fix the issue.

Custom binding project reference

Consider the following situations with a BizTalk Services project in a Visual Studio solution:

- In the same Visual Studio solution, there is a BizTalk Services project and a custom binding project. The BizTalk Service project has a reference to this custom binding project file.
- The BizTalk Service project has a reference to a custom binding/behavior DLL.

You 'Build' the solution in Visual Studio successfully. Then, you 'Rebuild' or 'Clean' the solution. After that, when you rebuild or clean again, the following error occurs:

Unable to copy file to "bin\Debug\FileName.dll". The process cannot access the file 'bin\Debug\FileName.dll' because it is being used by another process.

Workaround

- If [Visual Studio 2012 Update 3](#) is installed, you have the following two options:
 - Restart Visual Studio, or
 - Restart the solution. Then, perform only a Build on the solution.
- If [Visual Studio 2012 Update 3](#) is not installed, open Task Manager, click the Processes tab, click the MSBuild.exe process, and then click the End Process button.

Routing to BasicHttpRelay endpoints is not supported from bridges and BizTalk Services Portal if non-printable characters are promoted as HTTP headers

If you use non-printable characters as part of promoted properties for messages, those messages cannot be routed to relay destinations that use the BasicHttpRelay binding. Also, the promoted properties that are available as part of tracking are URL-encoded for blobs and un-encoded for destinations.

MDN is sent asynchronously even if the Send asynchronous MDN option is unchecked

Consider this scenario – If you select the **Send asynchronous MDN** checkbox and, specify a URL to send the async MDN to, and then uncheck the **Send asynchronous MDN** checkbox again, the MDN is still sent to the specified URL even though the option to send async MDNs is not selected.

As a workaround, you must clear the specified URL before unchecking the **Send asynchronous MDN** checkbox and then deploy the AS2 agreement.

Whitespace characters beyond a valid interchange cause an empty message to be sent to the suspend endpoint

If there are whitespaces beyond an IEA segment, the disassembler treats this as end of current interchange and looks at the next set of whitespaces as a next message. Since this is not valid interchange, you might observe that one successful message is sent to the route destination and one empty message is sent the suspend endpoint.

Tracking in BizTalk Services Portal

Tracking events are captured up to the EDI message processing and any correlation. If a message fails outside of the Protocol stage, Tracking will show as successful. In this situation, refer to the LOG section under the **Details** column in **Tracking** for error details. The X12 Receive and Send settings ([Create an X12 Agreement in Azure BizTalk Services](#)) provide information on the Protocol stage.

Update Agreement

The BizTalk Services Portal allows you to modify the Qualifier of an Identity when an agreement is configured. This can result in inconsistency properties. For example, there is an agreement using ZZ:1234567 and ZZ:7654321 the Qualifier. In the BizTalk Services Portal profile settings, you change ZZ:1234567 to be 01:ChangedValue. You open the agreement and 01:ChangedValue is displayed instead of ZZ:1234567. To modify the Qualifier of an identity, delete the agreement, update **Identities** in the partner profile and then recreate the agreement.

AZURE.WARNING This behavior impacts X12 and AS2.

AS2 Attachments

Attachments for AS2 messages are not supported in send or receive. Specifically, attachments are silently ignored and the message body is processed as a regular AS2 message.

Resources: Remembering Path

When adding **Resources**, the dialog window may not remember the path previously used to add a resource. To remember the previously-used path, try adding the BizTalk Services Portal web site to **Trusted Sites** in Internet Explorer.

If you rename the entity name of a bridge and close the project without saving changes, opening the entity again results in an error

Consider a scenario in the following order:

- Add a bridge (for example an XML One-Way Bridge) to a BizTalk Service project
- Rename the bridge by specifying a value for the Entity Name property. This renames the associated .bridgeconfig file with the name you specified.
- Close the .bcs file (by closing the tab in Visual Studio) without saving the changes.
- Open the .bcs file again from the Solution Explorer.
You will notice that while the associated .bridgeconfig file has the new name you specified, the entity name on the design surface is still the old name. If you try to open the Bridge Configuration by double-clicking the bridge component, you get the following error:

'<old name>' Entity's associated file '<old name>.bridgeconfig' does not exist To avoid running into this scenario, make sure you save changes after you rename the entities in a BizTalk Service project.

BizTalk Service project builds successfully even if an artifact has been excluded from a Visual Studio project

Consider a scenario where you add an artifact (for example, an XSD file) to a BizTalk Service project, include that artifact in the Bridge Configuration (for example, by specifying it as a Request message type), and then exclude it from the Visual Studio project. In such a case, building the project will not give any error as long as the deleted artifact is available on the disk at the same location from where it was included in the Visual Studio project.

The BizTalk Service project does not check for schema availability while configuring the bridges

In a BizTalk Service project, if a schema that is added to the project imports another schema, the BizTalk Service project does not check whether the imported schema is added to the project. If you try to build such a project, you do not get any build errors.

The response message for a XML Request-Reply Bridge is always of charset UTF-8

For this release, the charset of the response message from an XML Request-Reply Bridge is always set to UTF-8.

User-Defined Datatypes

The BizTalk Adapter Pack adapters within the BizTalk Adapter Service feature can utilize user-defined datatypes for adapter operations. When using user-defined datatypes, copy the files (.dll) to drive:\Program Files\Microsoft BizTalk Adapter Service\BAServiceRuntime\bin\ or to the Global Assembly Cache (GAC) on the server hosting the BizTalk Adapter Service service. Otherwise, the following error may occur on the client:

```
<s:Fault xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
<faultcode>s:Client</faultcode>
<faultstring xml:lang="en-US">The UDT with FullName "File, FileUDT, Version=Value, Culture=Value,
PublicKeyToken=Value" could not be loaded. Try placing the assembly containing the UDT definition in the Global
Assembly Cache.</faultstring>
<detail>
  <AFConnectRuntimeFault xmlns="http://Microsoft.ApplicationServer.Integration.AFConnect/2011"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
    <ExceptionCode>ERROR_IN_SENDING_MESSAGE</ExceptionCode>
  </AFConnectRuntimeFault>
</detail>
</s:Fault>
```

IMPORTANT

It is recommended to use GACUtil.exe to install a file into the Global Assembly Cache. GACUtil.exe documents how to use this tool and the Visual Studio command line options.

Restarting the BizTalk Adapter Service Web Site

Installing the **BizTalk Adapter Service Runtime*** creates the **BizTalk Adapter Service** web site in IIS that contains the **BAService** application. **BAService** application internally uses relay binding to extend the reach of on-premise service endpoint to the cloud. For a service hosted on-premises, the corresponding relay endpoint will be registered on the Service Bus only when the on-premises service starts.

If you stop and start an application, the configuration for auto-starting an application is not honored. So when **BAService** is stopped, you must always restart the **BizTalk Adapter Service** web site instead. Do not start or stop the **BAService** application.

Special characters should not be used for address and entity names of LOB components

You should not use special characters for address and entity names of LOB components. If you do so, you will get an error while deploying the BizTalk Service project. For certain characters like '%', the BizTalk Adapter Service website might go into a stopped state and you will have to manually start it.

Test Map with Get Context Property

If a Transform contains a **Get Context Property** Map Operation, **Test Map** will fail. As a temporary workaround, replace the **Get Context Property** Map Operation with a String Concatenate Map Operation containing dummy data. This will populate the target schema and allow you test other Transform functionality.

Test Map Property does not display

The **Test Map** properties do not display in Visual Studio. This can occur if the **Properties** window and the **Solution Explorer** window are not simultaneously docked. To resolve this, dock the **Properties** and the **Solution Explorer** windows.

Date Time Reformat drop-down is grayed out

When a Date Time Reformat Map Operation is added to the design surface and configured, the Format drop-down list may be grayed out. This can happen if the computer Display is set **Medium – 125%** or **Larger – 150%**. To resolve, set the display to **Smaller – 100% (default)** using the steps below:

1. Open the **Control Panel** and click **Appearance and Personalization**.
2. Click **Display**.

3. Click **Smaller – 100% (default)** and click **Apply**.

The **Format** drop-down list should now work as expected.

Duplicate agreements in the BizTalk Services Portal

Consider the following scenario:

1. Create an agreement using the Trading Partner Management OM API.
2. Open the agreement in the BizTalk Services Portal in two different tabs.
3. Deploy the agreement from both the tabs.
4. As a result, both the agreements get deployed resulting in duplicate entries in the BizTalk Services Portal

Workaround. Open any one of the duplicate agreements in the BizTalk Services Portal, and undeploy.

Bridges do not use updated certificate even after a certificate has been updated in the artifact store

Consider the following scenarios:

Scenario 1: Using thumbprint-based certificates for securing message transfer from a bridge to a service endpoint

Consider a scenario where you use thumbprint-based certificates in your BizTalk Service project. You update the certificate in the BizTalk Services Portal with the same name but a different thumbprint, but do not update the BizTalk Service project accordingly. In such a scenario, the bridge might continue to process the messages because the older certificate data might still be in the channel cache. After that, message processing fails.

Workaround: Update the certificate in the BizTalk Service project and redeploy the project.

Scenario 2: Using name-based behaviors to identify certificates for securing message transfer from a bridge to a service endpoint

Consider a scenario where you use name-based behaviors to identify certificates in your BizTalk Service project. You update the certificate in the BizTalk Services Portal but do not update the BizTalk Service project accordingly. In such a scenario, the bridge might continue to process the messages because the older certificate data might still be in the channel cache. After that, message processing fails.

Workaround: Update the certificate in the BizTalk Service project and redeploy the project.

Bridges continue to process messages even when the SQL database is offline

The BizTalk Services bridges continue to process messages for a while, even if the Microsoft Azure SQL Database (which stores the running information like deployed artifacts and pipelines), is offline. This is because BizTalk Services uses the cached artifacts and bridge configuration. If you do not want the bridges to process any messages when the SQL Database is offline, you can use the BizTalk Services PowerShell cmdlets to stop or suspend the BizTalk Service. See [Azure BizTalk Service Management Sample](#) for the Windows PowerShell cmdlets to manage operations.

Reading the XML message within a bridge's custom code component includes an extra BOM character

Consider a scenario where you want to read an XML message within a bridge's custom code. If you use the .NET API `System.Text.Encoding.UTF8.GetString(bytes)` an extra BOM character is included in the output at the beginning of the message. So, if you do not want the output to include the extra BOM character, you must use

```
System.IO.StreamReader().ReadToEnd()
```

Sending messages to a bridge using WCF does not scale

Messages sent to a bridge using WCF does not scale. You should instead use `HttpWebRequest` if you want a scalable client.

UPGRADE: Token Provider error after upgrading from BizTalk Services Preview to General Availability (GA)

There is an EDI or AS2 Agreement with active batches. When the BizTalk Service is upgraded from Preview to GA,

the following may occur:

- Error: The token provider was unable to provide a security token. Token provider returned message: The remote name could not be resolved.
- Batch tasks are canceled.

Workaround: After the BizTalk Service is updated to General Availability (GA), redeploy the agreement.

UPGRADE: Toolbox shows the old bridge icons after upgrading the BizTalk Services SDK

After you upgrade an earlier version of the BizTalk Services SDK, which had old icons representing the bridges, the toolbox continues to show the old icons for the bridges. However, if you add a bridge to BizTalk Service project designer surface, the surface shows the new icon.

Workaround. You can work around this issue by deleting the .tbd files under
:\Users<user>\AppData\Local\Microsoft\VisualStudio\11.0.

UPGRADE: BizTalk Portal update from Preview to GA might show an error indicating that the EDI capability is not available

If you are logged into the BizTalk Services Portal while the BizTalk Services is upgraded from Preview to GA, you might get the following error on the portal:

This capability is not available as part of this edition of Microsoft Azure BizTalk Services. To use these capabilities switch to an appropriate edition.

Resolution: Log out from the portal, close and open the browser, and then log into the portal.

UPGRADE: New tracking data does not show up after BizTalk Services is upgraded to GA

Assume a scenario where you have an XML bridge deployed on BizTalk Services Preview subscription. You send messages to the bridge and the corresponding tracking data is available on the BizTalk Services Portal. Now, if the BizTalk Services Portal and BizTalk Services runtime bits are upgraded to GA, and you send a message to the same bridge endpoint deployed earlier, the tracking data does not show up for messages sent after upgrade.

Pipelines versus Bridges

Throughout this document, the term 'pipelines' and 'bridges' are used interchangeably. Both essentially mean the same thing, which is, a message processing unit deployed on BizTalk Services.

Concepts

[BizTalk Services](#)