Server and Database Administration Guide for Microsoft Dynamics® AX 2009

Microsoft Corporation

Published: November 2009

Microsoft Dynamics is a line of integrated, adaptable business management solutions that enables you and your people to make business decisions with greater confidence. Microsoft Dynamics works like and with familiar Microsoft software, automating and streamlining financial, customer relationship and supply chain processes in a way that helps you drive business success.

U.S. and Canada Toll Free 1-888-477-7989

Worldwide +1-701-281-6500

www.microsoft.com/dynamics

Table of Contents

Check for updated instructions

The information contained in this document was current as of November 2009. The documentation may be updated as new information becomes available. For the most current documentation for system administrators, check the TechNet Library. For the most current documentation for developers, check the MSDN Library.

Getting started with server and database administration

The server and database administration documentation is for IT administrators. This documentation provides information about managing the base components of Microsoft Dynamics AX, including the AOS, the client, and the Business Connector. Information about importing and exporting data and using the Application Integration Framework (AIF) also is included.

For information about setting up Microsoft Dynamics AX, including security considerations, see the **System and Application Setup** Help, available from the Help menu in the Microsoft Dynamics AX client.

Use this table for quick access to the content in this document.

For information about	See
Managing AOS settings	Manage the Application Object Server (AOS)
Managing client settings	Managing the client in Microsoft Dynamics AX
Managing Business Connector	Managing instances of Business Connector
Importing and exporting data	Importing and exporting data
Using the Application Integration Framework (AIF)	Configure and manage AIF
Maintaining Microsoft Dynamics AX and planning for data recovery	Maintenance and data recovery
Optimizing performance	Optimizing performance
Uninstalling Microsoft Dynamics AX components	Modifying or uninstalling Microsoft Dynamics AX

Manage the Application Object Server (AOS)

The Application Object Server (AOS) for Microsoft Dynamics AX executes business logic and processing for clients connecting to Microsoft Dynamics AX.

Designate a batch server

A batch server is an Application Object Server (AOS) instance that processes batch jobs. Batch jobs are used to run tasks, such as printing reports or posting journals, at a specified time and probably on a different computer.

The first AOS to be set up is automatically designated as a batch server. Use multiple batch servers to increase throughput and reduce the amount of time it takes to run batches.

When you set up a batch server, you can specify the times that it is available for batch processing. We recommend excluding a server from batch processing when it is busy with regular transaction processing. For example, you may have servers in different time zones. You can set server schedules so that each AOS is available for user traffic during the day and for batch traffic overnight.

For more information about batches, see the **Applications and Business Processes** Help, available from the Microsoft Dynamics AX Help menu.

1. Click **Administration** > **Setup** > **Server configuration**.
2. Press CTRL+N to add a new batch server.
3. On the **Overview** tab, enter a server ID in the following format: *InstanceName@ServerName*.
4. Select **Is Batch Server** to enable batch processing on the server.
5. On the **Batch server schedule** tab, enter the maximum number of batch tasks that can be run on the AOS instance at one time. The server will continue to pick up tasks from the queue until it reaches its maximum.
6. To specify when the server is available for batch processing, enter a starting time in the **Start time** field and an ending time in the **End time** field. Press CTRL+N to enter an additional time period.

   Note:

   If the server is running a task when its batch processing availability ends, the task will continue running to completion. However, the server will not pick up any more tasks from the queue.
7. On the **Batch server groups** tab, use the arrow buttons to specify the batch groups that can run on the selected server. Batch groups are used to direct batch tasks to specific servers.

Start or stop the Application Object Server Windows service

You can start, stop, or restart the Application Object Server (AOS) Windows service from the Microsoft Windows Services utility.

1. Click **Start** > **Control Panel** > **Administrative Tools** > **Services**.
2. In the details panel (right pane), do one of the following:
   - Select the service that has a name in the format **Dynamics AX Object Server*<$InstanceNumberAOSInstanceName>***. In the toolbar, click the **Start Service**, **Stop Service**, or **Restart Service** icon as appropriate.
   - Right-click the service you want to change, and then click **Start**, **Stop**, or **Restart**.

Manage client connections for an AOS instance

Use the following procedures to accept or reject client connections for an AOS instance.

Set an AOS instance to accept new clients

Use the **Accept new clients** option to allow clients to connect to an AOS instance that was set to reject clients.

1. Open the **Online users** form (**Administration** > **Online users**).
2. On the **Server Instances** tab, select the AOS instance.
3. Click **Accept new clients**.

Set an AOS instance to reject new clients

Use the **Reject new clients** option to stop clients from connecting to an AOS instance.

1. Open the **Online users** form (**Administration** > **Online users**).
2. On the **Server Instances** tab, select the AOS instance.
3. Click **Reject new clients**.

   In-progress client sessions will not be disconnected automatically, but no new connections will be accepted. To end in-progress client sessions, use the **Online users** form. On the **Client Sessions** tab, select a session and click **End sessions**.

Managing configurations (Server)

A configuration is a group of startup and tracing settings for an Application Object Server (AOS) instance that is stored in the registry hive HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Dynamics Server\5.0*AOSinstancename**Configname* or in a configuration file. You may want to create new configurations and configuration files, save, or load configurations to support:

- Moving from a development environment to a production environment. Change the application directory that an AOS instance points to, the database, and whether a system allows debugging.
- Tuning Microsoft Dynamics AX. Save a configuration with the defaults, and then change compression, database tuning, and tracing settings one at a time, and save them as different configurations. Run Microsoft Dynamics AX with each different configuration, and evaluate how it performs.

The table below describes the topics in this section.

Topic	Description
Configuration security (Server)	This topic describes how to manage security for AOS configurations.
Create a new configuration (Server)	This topic describes how to create a new configuration.
Save or export a configuration (Server)	This topic describes how to save or export a server configuration.
Load a configuration (Server)	This topic describes how to load a server configuration from a file.

See Also

Using the command line to manage the AOS

Configuration security (Server)

On the computer running the Application Object Server (AOS), only members of the local Microsoft Windows Power User group or Administrators group can change configuration settings.

Restrict membership in these groups as much as is feasible, to reduce the potential for malicious mischief.

The log directory cannot be changed - the log is always installed to *installationdirectory\log*. Be sure to set the access control list for the directory so that only administrators and the AOS account (the domain account or Network Service account associated with an AOS instance) have write privileges to this directory.

Create a new configuration (Server)

You can create a new configuration for an Application Object Server (AOS) instance to:

- Tune Microsoft Dynamics AX. Save a configuration with the defaults, and then change compression, database turning, and tracing settings one at a time, and save them as different configurations. Start with each configuration in turn, and evaluate how the system performs.
- Move from a development environment to a production environment. Change the application directory that an AOS instance points to, the database it is connected to, and whether a system allows debugging.

You cannot modify the original configuration of a system. To change a configuration, you must create a new one and modify it.

1. Open the Server Configuration utility (**Start** > **Administrative Tools** > **Microsoft Dynamics AX Server Configuration Utility**).

2. In the **Application Object Server** instance box, select the AOS instance to modify.
3. Click **Manage**, and then click **Create configuration**:
4. In the **Create configuration** dialog box, in the **Name** box, type a name.
5. Decide whether you want to copy settings from the active configuration or the original (default) configuration, and then click **OK**.

Save or export a configuration (Server)

You can save startup settings for an Application Object Server (AOS) instance as a configuration stored in the registry or as a configuration file. Saved startup settings enable you to:

- Tune Microsoft Dynamics AX. Change settings, and then tune your system by comparing performance with saved configurations that contain varied compression, database turning, and tracing settings.
- Move a configuration from one server to another.

Save a configuration to the registry

Use this procedure to save changed settings.

1. Open the Server Configuration utility (**Start** > **Administrative Tools** > **Microsoft Dynamics AX Server Configuration Utility**).
2. Make any changes to the currently selected AOS instance and configuration that you want.
3. Click **OK**.

Export a configuration to a file

Use this procedure if you want to copy a configuration to an AOS instance on another computer.

1. Open the Server Configuration utility (**Start** > **Administrative Tools** > **Microsoft Dynamics AX Server Configuration Utility**).
2. Verify that the currently selected AOS instance and configuration are the ones you want to save.
3. Click **Manage**, and then click **Export**. Choose a location and name for the configuration file, and then click **Save**. The file is saved with an .axc extension.

Export all configurations to a file

Use this procedure if you want to copy all configurations for an AOS.

1. Open the Server Configuration utility (**Start** > **Administrative Tools** > **Microsoft Dynamics AX Server Configuration Utility**).

2. Verify that the currently selected AOS instance and configuration are the ones you want to save.
3. Click **Manage**, and then click **Export All**. Choose a location and name for the configuration file, and then click **Save**. The files are saved with an .axc extension.

Save a configuration file with a new name

You may want to save a configuration file under a new name to provide a record of the changes you are making to configurations over time.

1. Open the Server Configuration utility (**Start** > **Administrative Tools** > **Microsoft Dynamics AX Server Configuration Utility**).
2. Verify that the currently selected AOS instance is the one you want.
3. Import a configuration file.
4. Click **Manage**, and then click **Save configuration file**. Choose a location and name for the configuration file, and then click **Save**. The file is saved with an .axc extension.

Load a configuration (Server)

If you are an Administrator or Power User on the computer, you can load a configuration (startup settings) for an Application Object Server (AOS) instance from a configuration stored in the registry or from a configuration file.

Load a configuration from the registry

1. Open the Server Configuration utility (**Start** > **Administrative Tools** > **Microsoft Dynamics AX Server Configuration Utility**).
2. In the **Application Object Server instance** box, select the AOS instance to modify.
3. In the **Configuration** list, click the configuration you want to open.

Import a configuration from a file

Although you can use the **Open** command to view a saved configuration, Microsoft Dynamics AX does not store the settings from the opened file to the registry. To store saved configuration files to the registry, you must use the **Import** command.

1. Open the Server Configuration utility (**Start** > **Administrative Tools** > **Microsoft Dynamics AX Server Configuration Utility**).
2. In the **Application Object Server instance** box, select the AOS instance to modify.
3. Click **Manage**, and then click **Import**.
4. Browse to the configuration file you would like to use (***.axc**), and open it.

See Also

Save or export a configuration (Server)

Connect an Application Object Server instance to a different application

You can connect an Application Object Server (AOS) instance to a different application file location. You may want to do this in the following situations:

- Moving from a development or staging environment to a production environment.
- Upgrading a Microsoft Dynamics AX system.

1. Open the Server Configuration utility (**Start** > **Administrative Tools** > **Microsoft Dynamics AX Server Configuration Utility**).
2. Verify that the currently selected AOS instance and configuration are the ones you want to modify.
3. On the **Application Object Server** tab, in the **Application file location** box, type the path to the application you would like to connect to, and then click **OK**.

Connect an Application Object Server instance to a different bin directory

By default, Microsoft Dynamics AX reads the text displayed in the user interface from the kernel text data file (*.ktd) stored in the bin directory of the Application Object Server (AOS) directory. You can have Microsoft Dynamics AX use a different kernel text data file by modifying a copy of the kernel text data file, choosing a location, and pointing an AOS instance to it.

1. Open the Server Configuration utility (**Start** > **Control Panel** > **Administrative Tools** > **Microsoft Dynamics AX Server Configuration Utility**).
2. Verify that the currently selected AOS instance and configuration are the ones you want to modify.
3. On the **Application Object Server** tab, in the **Alternate bin directory** box, type or browse to the location of the kernel text data file you want to point to, click **Apply**, and then click **OK**.

Run commands at startup (Server)

You can run a command when an Application Object Server (AOS) instance starts. You should exercise caution and test the commands you are using thoroughly in a development or test environment before using them in a Microsoft Dynamics AX production environment. Startup commands are not evaluated before being passed to the AOS instance, and if not correct, can cause the instance to fail to start.

To run a command at startup

1. Open the Server Configuration utility (**Start** > **Administrative Tools** > **Microsoft Dynamics AX Server Configuration Utility**).
2. Verify that the currently selected AOS instance and configuration are the ones you want to modify.
3. On the **Application Object Server** tab, in the **Run advanced startup commands** box, type the command or commands you want to run at startup, and then click **OK**.

See Also

Using the command line to manage the AOS

Change the TCP/IP port an Application Object Server instance runs on

By default, an Application Object Server (AOS) instance runs on port 2712. Subsequent instances are assigned TCP/IP ports dynamically. You can set a different static port for an instance using the Server Configuration Utility.

You may want to set a different port when:

- The firewall you are using restricts you to use specific TCP/IP ports.
- You have multiple AOS instances running on a computer, and always want to know what they are.

1. Open the Server Configuration utility (**Start** > **Administrative Tools** > **Microsoft Dynamics AX Server Configuration Utility**).
2. Verify that the currently selected AOS instance and configuration are the ones you want to modify.
3. On the **Application Object Server** tab, in the **TCP/IP port** box, type the port you want the instance to run on, click **Apply**, and then click **OK**.

Enable printing from a computer running Application Object Server

You can set Microsoft Dynamics AX to allow printing to a printer attached to a computer running an Application Object Server (AOS) instance. You may want to use this type of printing for reports or other data to which you want to restrict access. Options must be set on both the AOS and client to enable printing.

Note:

To print a document to a PDF file, you must have installed an Adobe Printer driver. For details, refer to your Adobe license and documentation (http://www.adobe.com)

Set an Application Object Server instance to enable printing

Note:

This option is set in the Microsoft Dynamics AX Server Configuration Utility.

1. Open the Server Configuration utility (**Start** > **Administrative Tools** > **Microsoft Dynamics AX Server Configuration Utility**).
2. Verify that the currently selected AOS instance and configuration are the ones you want to modify.
3. On the **Application Object Server** tab, click C**onnect to printers on this server**, and then click **OK**.

Set a client to use printing

Note:

This option is set in the Microsoft Dynamics AX Client Configuration Utility.

1. Open the Configuration utility (**Start** > **Control Panel** > **Administrative Tools** > **Microsoft Dynamics AX Client Configuration Utility**).
2. Verify that the currently selected configuration target and configuration are the ones you want to modify.
3. On the **Connection** tab, click **Connect to printers on the server**, and then click **OK**.

Allow debugging

You can set Microsoft Dynamics AX to allow debugging on an Application Object Server (AOS), or on a client that is running the .NET Business Connector or COM Business Connector.

Allow debugging on an Application Object Server instance

Note:

This option is set in the Microsoft Dynamics AX Server Configuration Utility.

1. Open the Server Configuration utility (**Start** > **Administrative Tools** > **Microsoft Dynamics AX Server Configuration Utility**).
2. Verify that the currently selected AOS instance and configuration are the ones you want to modify.
3. On the **Application Object Server** tab, click **Enable breakpoints to debug X++ code running on this server**, and then click **OK**.

Allow debugging of a Business Connector or Client

The following options in the Microsoft Dynamics AX Configuration Utility can used to enable breakpoints for a Business Connector instance or a client:

- **Enable user breakpoints to debug code running in the Business Connector**

  For sessions owned by users for whom debug is enabled within Microsoft Dynamics AX, allow X++ code running in the Business Connector to be interrupted by breakpoints
- **Enable global breakpoints to debug code running in the Business Connector or client**

  For all users, allow X++ code running in the Business Connector or client to be interrupted by global breakpoints.

  Note:

  When this option is set, you will see all breakpoints for currently active clients.

To debug Business Connector, you must set both debug options. To debug the client, set only the second option.

Enable breakpoints for a Business Connector instance

1. Open the Configuration utility (**Start** > **Control Panel** > **Administrative Tools** > **Microsoft Dynamics AX Configuration Utility**).
2. On the **Developer** tab, select **Enable user breakpoints to debug code running in the Business Connector**.
3. Select **Enable global breakpoints to debug code running in the Business Connector or client**, and then click **OK**.

Enable breakpoints for a client

1. Open the Configuration utility (**Start** > **Control Panel** > **Administrative Tools** > **Microsoft Dynamics AX Configuration Utility**).
2. On the **Developer** tab, select **Enable global breakpoints to debug code running in the Business Connector or client**, and then click **OK**.

Encrypt data

By default, Microsoft Dynamics AX secures data sent across a network by using the remote procedure call (RPC) function RPC_C_AUTHN_LEVEL_CONNECT, which validates user credentials at the time a connection is established.

You can also encrypt data if your security needs require it. When you turn on encryption, Microsoft Dynamics AX uses the RPC_C_AUTHN_LEVEL_PKT_PRIVACY call, which provides the highest security level available through RPC.

For more information about RPC security, search for RPC security in MSDN: http://msdn.microsoft.com/.

Note:

Enabling encryption may decrease performance between five and ten percent.

1. Open the Microsoft Dynamics AX Configuration utility (**Start** > **Control Panel** > **Administrative Tools** > **Microsoft Dynamics AX Configuration Utility**).
2. Verify that the currently selected instance and configuration are the ones you want to modify.
3. On the **Connection** tab, click **Encrypt client-to-server communications**, and then click **OK**.

Compress data between clients and an Application Object Server

If you have a slow network, with either low bandwidth or high latency causing slow response times, you may want to increase the compression of the data sent between Microsoft Dynamics AX clients and servers.

Decreasing the data size in each client/server request or response can greatly reduce transmission time because of:

- Better use of limited bandwidth.
- Decreased chance of bit errors.
- Decreased chance of exceeding TCP window size.

The TCP window size is the amount of data received (in bytes) that can be buffered at one time on a connection. The sending host can send only that amount of data before waiting for an acknowledgment and window update from the receiving host.

If the size per client/server request or response is kept below the TCP window size, the sender does not have to wait for an acknowledgment (ACK) when the window size is exceeded. If exceeding the TCP window size cannot be avoided, making as few TCP round trips as possible for each client/server request or response is important.

1. Open the Server Configuration utility (**Start** > **Control Panel** > **Administrative Tools** > **Microsoft Dynamics AX Server Configuration Utility**).
2. Verify that the currently selected Application Object Server (AOS) instance and configuration are the ones you want to modify.
3. In the **Minimum packet size to compress (in KB)** field, choose a packet size.

   Choose the smallest useful packet size to compress. The larger the packet size chosen, the smaller the gains in performance.

Connect an Application Object Server instance to a different database

You can connect an Application Object Server (AOS) instance to a different database. You may want to do this in the following situations:

- Moving from a development or staging environment to a production environment.
- Upgrading a Microsoft Dynamics AX system.

Note:

If you are trying to connect to a database that was not created by Microsoft Dynamics AX Setup, the AOS account (the domain account or Network Service account associated with the AOS service) may not have appropriate rights in Microsoft SQL Server. The AOS account must be a user in the database, and be assigned to the following database roles **db_ddladmin**, **db_datareader**, and **db_datawriter**.

Connect to a different database

1. Open the Server Configuration utility (**Start** > **Administrative Tools** > **Microsoft Dynamics AX Server Configuration Utility**).
2. Verify that the currently selected AOS instance and configuration are the ones you want to modify.
3. On the **Database connection** tab, enter the details of the database you would like to connect to, and then click **OK**.

   For a SQL Server connection, consider the following:

   - To specify a SQL Server named instance, use the format *MyServer**MyInstance*.
   - To specify the local SQL Server instance on this computer, enter *(local)*.

For an Oracle connection, consider the following:

- Choose whether to connect using a net service or custom settings.
- In the **Use this schema** box, specify the schema under which the Microsoft Dynamics AX objects are stored in the database.

Tune database settings

You may want to tune the database settings for Microsoft Dynamics AX to improve performance. Before changing settings, trace the usage of your Microsoft Dynamics AX database to ensure that you have clear understanding of performance under the current settings. To trace Microsoft Dynamics AX database performance, use:

- Tracing from the Microsoft Dynamics AX Server Configuration Utility. For more information, see Set tracing options.
- Windows Performance Monitor, using Microsoft Dynamics AX Object counters.

Test all tuning changes before implementing them in a production environment. In a test or development environment, make a single change and then test your system's performance before making another change.

Tune connections

The following table lists common connection issues, and some adjustments to try in the Server Configuration Utility.

Symptom	Adjustments to try
Results for common queries are returned slowly.	Increase the **Maximum buffer size** value.
Results for ad hoc queries are returned slowly.	Check to see that the appropriate indexes are in place. For the most recent guidance about indexing, check Microsoft Dynamics AX Online.
Transactions are failing frequently.	Decrease the **Transaction retry interval** value.
Data grids for commonly used tables draw slowly.	Increase the **Array fetch ahead** value.

Tune queries

If queries in the system are running slowly, you may want to change settings for literals, string functions, or hints. Microsoft Dynamics AX no longer uses literals by default in form and report queries, or in complex-join queries.

Adjust the use of literals

Microsoft Dynamics AX may pass either parameters (placeholders) or literals (actual values) in queries.

- Parameters allow Microsoft Dynamics AX and the database server to reuse the query when search values change. They are preferred for high-frequency queries.
- Literals allow the database server to optimize the query for a specific piece of information. This provides an optimal query for that piece of information, but the database server must perform the optimization for every query executed. Literals may be used for long running queries such as complex joins.

A developer can override the default use of literals by specifying parameters in their code, or an administrator can override the use of literals in the Server Configuration Utility.

Symptom	Adjustments to try	Anticipated effect
Long-running queries run slowly.	Review the query plan statements sent to SQL Server and consider taking corrective action. Using literals may be one solution. Select **Use literals in join queries from forms and reports**. Select **Use literals in complex joins from X++.**	Long-running queries pass literals to the database. Processing time for long-running queries should go down.

Adjust the use of autogenerated string functions

Microsoft Dynamics AX embeds some string functions in `SELECT` statements automatically. String functions are included to support:

- Treating uppercase and lowercase versions of the same text as the same text (single case) for Oracle installations.
- Left justification or right justification.

When a string function is included in a query, the optimizer may have to choose a less-than-optimal access plan, such as a table scan, for retrieving data. If customers do not require the use of mixed case outside Microsoft Dynamics AX and do not use left justification or right justification, these functions are not required and should be turned off. To improve performance, we recommend that all values be stored left-aligned by default.

Adjust the use of hints

In Microsoft Dynamics AX, you can allow developers to override the index selected by the query optimizer. In most situations, allowing the query optimizer to select an index for a query results in improved performance.

If queries include `INDEX` hints and are running more slowly than expected, clear the **Allow INDEX hints in queries** option.

Changes in the use of hints

If you have upgraded to Microsoft Dynamics AX, the queries in your system may contain outdated Microsoft SQL Server hints. Configuration commands are no longer available to globally enable or disable many of the hints from previous versions. If hints are explicitly specified in an X++ statement, they are added to the SQL Server query that is generated. Otherwise, they are not added.

The following changes have also been made:

- The `OPTION (FAST)`, `LOOP`, and `FORCE ORDER` hints are not applied by default, but are applied if explicitly specified in X++.
- A `FIRSTONLY` hint in X++ is translated into the addition of a `TOP 1` statement to the SQL Server query.
- `FASTFORWARD` cursors are used for all user queries unless a cursor has been marked as `FOR UPDATE`.
- `FOR UPDATE`, `NOLOCK`, and `READPAST`, hints are added to statements depending on the type of the cursor that an X++ query has produced. No interface is available to modify these hints.

Change the concurrency mode

Concurrency settings enable you to reduce locking conflicts on your system. For more information, see the following topics in the Microsoft Dynamics AX 2009 Developer Documentation.

- Performance optimizations: Database design and operations
- Transaction integrity
- Exception handling
- Select statement syntax
- Table properties

Using the command line to manage the AOS

Configuration commands set the options that are used when an Application Object Server (AOS) instance starts. Configuration commands can be run directly from the following locations:

- In a configuration file.
- In the **Configuration Command to run at kernel startup** field in the Server Configuration Utility.
- From a command prompt when starting an AOS instance.

Configuration commands require that you use different syntax if you are setting them in a configuration file, or executing them in the **Configuration Command to run at kernel startup** field or from a command prompt. The syntax variations are provided in the following sections.

General options

This table describes the general options you can use to work with configurations and files.

Command in configuration file	Command from command line	Configuration utility option	Description
This command cannot be set in a file.	-**regConfig**=<*config name*>	Configuration	Specify the name of the current group of settings for this AOS instance.

Application Object Server options

This table describes the options you can use to manage how an AOS functions.

Command in configuration file	Command from command line	Configuration utility option	Description
application,Text,<*applicationname*>	-**application**=<*applicationname*>	**Application instance**	Specify the instance of an application that the AOS instance connects to.
bindir,Text,<*path*>	-**bindir**=<*path*>	**Alternate bin directory**	Specify the location of a directory containing an alternate kernel text data file (one of the Microsoft Dynamics AX label files).

Command in configuration file	Command from command line	Configuration utility option	Description
compressiondisabled,Int,1	-**compressiondisabled**	Option not available in utility	This is a binary command that is not set by default. When this value is absent, data sent between the AOS and its clients is compressed to speed client-server communications. If the value is present, then compression of packets is turned off. To turn on packet compression, remove the value from the configuration file. We recommend that you not disable compression. Disabling compressions can negatively affect system performance and security.
compressionminsize,Text,<*number*>	-**compressionminsize**=<*number*>	**Minimum packet size to compress**	Specify the smallest useful packet size to compress. The larger the packet size chosen, the smaller the gains in speed.
directory,Text,<*path*>	-**directory**=<*spath*>	**Application file location**	Specify the location of the application files for the AOS instance to connect and write to.
exposeserverprinters,Int,1	-**exposeserverprinters**	**Allow clients to connect to printers on this server**	This is a binary command that is not set by default. When this value is present, clients are allowed to connect to printers that are connected to the AOS computer.
port,Text,<*portnumber*>	-**port**=<*portnumber*>	**TCP/IP port**	The TCP/IP port that the AOS instance should use to connect to clients. The default value is 2712.
xppdebug,Text,<*0,1*>	-**xppdebug**=<*0,1*>	**Enable breakpoints to debug X++ code running on this server**	Enable clients to trace their interactions with this AOS instance. The default is off (0).

Command in configuration file	Command from command line	Configuration utility option	Description
caslevel,Text,<*enable/disable/trace*>	-**caslevel**=<*enable/disable/trace*>	Option not available in utility	Code Access Security (CAS) is the mechanism in Microsoft Dynamics AX that is used to protect specific APIs. For a list of these APIs, see "Secured APIs" in the Microsoft Dynamics AX 2009 Developer Documentation. *Enable*, the default setting, activates CAS for all CAS-protected APIs. If a CAS-protected API is invoked without following the correct consumer steps, an error is generated. *Trace* is used to simulate CAS being enabled. An error is not generated if a CAS-protected API is invoked incorrectly. Instead, debug information is written to the Infolog. Use in development or test environments to determine the changes that need to be made to get the system working. **Important:** Do not set the **caslevel** to *Trace* in production environments. *Disable* disables CAS entirely. **Important:** Do not set the **caslevel** to *Disable* in production environments. For more information about securing APIs, see "How to: Secure an API on the AOS" in the Microsoft Dynamics AX 2009 Developer Documentation.

Command in configuration file	Command from command line	Configuration utility option	Description
MaxConcurrentUISessions,Text,<*value*>	-**MaxConcurrentUISessions**=<*value*>		Set the maximum number of concurrent Microsoft Dynamics AX client sessions. The minimum value is 0, the maximum value (and default) is 65535. For details about using this setting to tune AOS performance, see "Tune application object server settings" in the Microsoft Dynamics AX 2009 Server Configuration Help (Start > Administrative Tools > Microsoft Dynamics AX 2009 Server Configuration > Help).
MaxConcurrentGuestSessions,Text,<*value*>	-**MaxConcurrentGuestSessions**=<*value*>		Set the maximum number of concurrent Guest (anonymous user) sessions. The minimum value is 0, the maximum value (and default) is 65535. For details about using this setting to tune AOS performance, see "Tune application object server settings" in the Microsoft Dynamics AX 2009 Server Configuration Help (Start > Administrative Tools > Microsoft Dynamics AX 2009 Server Configuration > Help).
MaxConcurrentWebSessions,Text,<*value*>	-**MaxConcurrentWebSessions**=<*value*>		Set the maximum number of concurrent Enterprise Portal sessions, including Guest sessions. The minimum value is 0, the maximum value (and default) is 65535. For details about using this setting to tune AOS performance, see "Tune application object server settings" in the Microsoft Dynamics AX 2009 Server Configuration Help (Start > Administrative Tools > Microsoft Dynamics AX 2009 Server Configuration > Help).

Command in configuration file	Command from command line	Configuration utility option	Description
MaxConcurrentBCSessions,Text,<*value*>	-**MaxConcurrentBCSessions**=<*value*>		Set the maximum number of concurrent Business Connector sessions, including all Web sessions (all Web sessions come through Business Connector). The default value is 65535. For details about using this setting to tune AOS performance, see "Tune application object server settings" in the Microsoft Dynamics AX 2009 Server Configuration Help (Start > Administrative Tools > Microsoft Dynamics AX 2009 Server Configuration > Help).
MaxMemLoad,Text,<*value*>	-**MaxMemLoad**=<*value*>		Set the maximum amount of memory usage (the maximum percentage of physical memory that is in use on the computer). The default value is 0. For details about using this setting to tune AOS performance, see "Tune application object server settings" in the Microsoft Dynamics AX 2009 Server Configuration Help (Start > Administrative Tools > Microsoft Dynamics AX 2009 Server Configuration > Help).
MaxConcurrentSessions,Int,<*value*>	-**MaxConcurrentSessions**=<*value*>	**Maximum number of client sessions**	Set the maximum number of client sessions this AOS instance will accept. The minimum value is 0, the maximum value (and default) is 65535.
Loadbalance,Int,<0,1>	-**LoadBalance**=<0,1>	**Make this AOS instance part of the load balancing cluster**	Set this AOS instance to load balance client connection requests with other AOS instances that also have load balancing enabled.

Command in configuration file	Command from command line	Configuration utility option	Description
startupcmd,Text,<*command*>	-**startupCmd**=<*command*>	**Command to run at application startup**	Enter a **SysStartupCmd** method to run when this client application starts. For details, see Run commands at startup (Server).
extracmd,Text,<*command*>	-**extracmd**=<*command*>	**Configuration command to run at kernel startup**	Enter any configuration command to run when the kernel starts.

Database connection options

This table describes the options you can use to connect to a database.

Command in configuration file	Command from command line	Configuration utility option	Description
createdsn,Text,<*microsoftsqlserver, oracle*>	**createdsn**=<*microsoftsqlserver, oracle*>	Option not available in utility	Create the data source in the ODBC manager.
dsn,text,<*portnumber*>	**-dsn=**<*portnumber*>	Option not available in utility	Point to a specific data source.
database,Text,<*databasename*>	**-database**=<*databasename*>	**Database to connect to**	Specify the database to connect to.
dbcli,Text,<*ODBC, OCI*>	-**dbcli**=<*ODBC, OCI*>	Option not available in utility	Run Microsoft Dynamics AX in either ODBC or OCI (Oracle) mode. ODBC is the default.
dbserver,Text,<*servername*>	**-dbserver**=<*servername*>	Option not available in utility	SQL Server name.

Database tuning options

This table describes the options you can use to tune database performance.

Command in configuration file	Command from command line	Configuration utility option	Description
connectionidletimeout,Text,<*0,1*>	**-connectionidletimeout**=<*0,1*>	**Leave the connection running when idle**	Retain a connection to the database when no transactions are running.
connectionidletimeout,Text,<*time*>	**-connectionidletimeout**=<*time*>	**Maximum idle time before closing**	Specify the amount of time to leave a database connection idle before closing it.
fetchahead,Text,<*number*>	**-fetchahead**=<*number*>	**Array fetch ahead**	Specify the maximum number of records that the system fetches at the same time. Starts as your local default computer setting of 100.
hint,Text,1	**-hint**=<*0,1*>	**Allow INDEX hints in queries**	Enable any query written with an `INDEX` hint to override the index selected by the database management system.
hint,Text,2	**-hint**=<*0, 2*>	**Include LTRIM in all SELECT statements to remove leading space from right-aligned columns**	Add `LTRIM` to all queries generated by Microsoft Dynamics AX. Using `LTRIM` forces the database to perform a table scan, which can slow query results. Set to 2 to enable this feature, and 0 to disable it.
ignoredatasourceindex,Text,<*0,1*>	**-ignoredatasourceindex**=<*0*, *1*>	**Generate ORDER BY clauses from WHERE clauses**	Set to 1 to override the ordering specified by the index on the data source, using the order of the columns as specified in the `WHERE` clause. This can improve query performance.
newconnectionretrycount,Text,<*number*>	**-newconnectionretrycount**=<*number*>	**Number of connection retries**	Specify the number of times to try connecting to the database before failing.

Command in configuration file	Command from command line	Configuration utility option	Description
newconnectionretrydelayms,Text,<*time*>	**-newconnectionretrydelayms**=<*time*>	**Connection retry interval**	Specify the interval between attempts to connect to the database in milliseconds.
opencursors,Text,<*number*>	**-opencursors**=<*number*>	**Maximum open cursors**	Specify the maximum number of database cursors to keep open for reuse in a connection. Starts as your local computer setting, which defaults to 90.
retry,Text,<*time*>	**-retry**=<*time*>	**Transaction retry interval (in seconds)**	Specify the delay before re-executing a transaction after a deadlock. The default value is 5 seconds.
sqlbuffer,Text,<*number*>	**-sqlbuffer**=<*number*>	**Maximum buffer size**	Specify the maximum size of the data retrieval buffer. The larger the buffer, the greater the number of records transferred at the same time. Starts as your local default computer setting of 24.
sqlcomplexliterals,Text,<*0,1*>	**-sqlcomplexliterals** =<*0,1*>	**Use literals in complex joins from X++**	Specify that Microsoft Dynamics AX use literals rather than parameters for complex joins to optimize performance.
sqlformliterals,Text, <*0,1*>	**-sqlformliterals**=<*0,1*>	**Use literals in join queries from forms and reports**	Specify that Microsoft Dynamics AX use literals rather than parameters in long-running queries to optimize performance.

Tracing options

This table describes the options you can use to trace calls between the AOS, the database and clients.

Note:

The **logdir** directory where the trace files are stored cannot be changed. It is the server installation directory\log.

Command in configuration file	Command from command line	Configuration utility option	Description
TraceStart,Int,1	**-TraceStart**=	**Start trace** **Stop trace**	Specify whether trace should be started or stopped: 0 – stop trace 1 – start trace The default value is 0.
traceeventsenabled,Text, *<1; 100; 101; 200; 201; 202; 203; 204; 205>*	**-TraceEventsEnabled**=*<1; 100; 101; 200; 201; 202; 203; 204; 205>*	See below	Specify the event types to be enabled. You can enable multiple event types using a semi-colon (;) as the delimiter. See below for detailed descriptions. The default value is 1.
traceeventsenabled,Text,1	**-TraceEventsEnabled**=1	**RPC round trips to server**	Trace all remote procedure call (RPC) round trips from any client to the server.
traceeventsenabled,Text,100	**-TraceEventsEnabled**=100	**X++ method calls**	Trace all X++ methods that are invoked on the server.
traceeventsenabled,Text,101	**-TraceEventsEnabled**=101	**Function calls**	Trace all function calls that are invoked on the server.
traceeventsenabled,Text,200	**-TraceEventsEnabled**=200	**Connect and disconnect**	Trace each time the AOS connects and disconnects from the database.
traceeventsenabled,Text,201	**-TraceEventsEnabled**=201	**Transactions: TTSBegin, TTSCommit, TTSAbort**	Trace all transactions that use the `TTSBegin`, `TTSCommit`, and `TTSAbort` statements.

Command in configuration file	Command from command line	Configuration utility option	Description
traceeventsenabled,Text,202	**-TraceEventsEnabled**=202	**SQL statements**	Trace all SQL Server statements that are invoked on the server.
traceeventsenabled,Text,203	**-TraceEventsEnabled**=203	**Bind variables**	Trace all columns that are used as input bind variables. **Note:** SQL Statements (202) must also be on to enable this option.
traceeventsenabled,Text,204	**-TraceEventsEnabled**=204	**Row fetch**	Trace all rows that are fetched using SQL Server. **Note:** SQL Statements (202) must also be on to enable this option.
traceeventsenabled,Text,205	**-TraceEventsEnabled**=205	**Row fetch summary (count and time)**	Count all rows that are fetched, and record the time spent fetching. **Note:** SQL Statements (202) must also be on to enable this option.
tracexppmethodcalldepth,Text,<*number*>	**-TraceXppMethodCallDepth**=<*number*>	**Number of nested calls:**	Specify the maximum call depth to be traced for X++ methods. **Note:** **TraceEventsEnabled** must also be set to 100 to use this command. The default value is 3.
tracemaxfilesize,Text,<*number*>	**-TraceMaxFileSize**=<*number*>	Option not available in utility	Specify the maximum size for each trace file in megabytes (MB). The default value is 10MB.

Command in configuration file	Command from command line	Configuration utility option	Description
tracebuffersize,Text,<*number*>	**-TraceBufferSize**=<*0:64*>	Option not available in utility	Specify the Event Tracing for Windows buffer size, in kilobytes (KB). The maximum size that can be set is 64KB. The default value is 20KB.
traceallowclient,Int,1	**-TraceAllowClient**	**Allow client tracing on Application Object Server instance**	This is a binary command that is not set by default. Specify whether client tracing is allowed on this AOS instance.

Unfamiliar configuration options

In the configuration files generated by Microsoft Dynamics AX, you may see unfamiliar options. Some are legacy options (configuration options from previous versions) that are not in use. Other configuration options remain in both the client or server configuration files, although they only apply to client or server, because in previous product versions the utilities were combined. We recommend that you do not change values for these options; unexpected results may occur.

Value in configuration file	Applies to
company,Text,	Client
client,Text,thin	Legacy
application,Text,standard	Client
broadcast,Text,	Legacy
aol,Text,sys	Client
aolcode,Text,	Client
native,Int,0	Legacy
sqluser,Text,	Legacy
hassqlpwd,Int,0	Legacy
sqlpwd,Text,	Legacy
startupmsg,Text,	Client
localappldoc,Int,0	Legacy
localsysdoc,Int,0	Legacy

Value in configuration file	Applies to
applshare,Int,0	Legacy
applexclusive,Int,1	Legacy
startupcmd,Text,test	Client
hascompwd,Int,0	Legacy
compwd,Text,	Legacy
allowunauth,Int,0	Legacy
windowsauth,Text,1	Legacy
aosencryption,Text,0	Client
xppdebug,Text,1	Legacy
ociuser,Text,	Legacy
hasocipwd,Int,0	Legacy
ocipwd,Text,	Legacy
dbunicodeenabled,Text,1	Legacy

Troubleshoot problems operating the Application Object Server

This topic provides information about how to troubleshoot issues encountered when running or connecting to an Application Object Server (AOS) instance.

If you are experiencing fatal errors in an instance of the AOS, you can help Microsoft solve future problems with the AOS by opening the Microsoft Dynamics AX Server Configuration Utility (**Start** > **Administrative Tools** > **Microsoft Dynamics AX Server Configuration**) and selecting the option **For all AOS instances on this computer, automatically send reports about fatal errors to Microsoft**.

If you enable error reporting, information is sent over a secure (https) connection to Microsoft, where it is stored with limited access. Microsoft uses the reports only to improve Microsoft Dynamics AX, and treats all information as confidential.

Client cannot connect to an AOS instance

When a client cannot connect to an instance of the Application Object Server (AOS), one of the following may be the cause.

AOS is starting

The first time you start a client after the AOS has been installed, the AOS service may still be starting up. On the AOS computer, use the Services manager to determine whether the service has started.

Port is not open

If the Application Object Server is installed on a computer with a firewall, be sure that the port you are trying to connect to is open. The default port is 2712, but additional instances may install on ports 2713, 2714, and so on.

AOS instance cannot connect to new database

If you are trying to connect to a database that was not created by Setup, the AOS account (the domain account or Network Service account associated with an AOS instance) may not have appropriate rights in SQL Server. The AOS account must be a user in the database and be assigned to the database roles **db_ddladmin**, **db_datareader**, and **db_datawriter**.

AOS instance does not start after changing databases

When an AOS instance and SQL Server database are installed on the same computer, Setup creates a dependency—the AOS instance does not start unless the SQL Server database is running.

If you change the AOS instance to use SQL Server on a different computer, you must manually change or delete the dependency using the SC tool (SC.exe) from the command prompt. To make the AOS dependent on a SQL Server instance, open a command prompt window and type the following command:

sc config *AOSinstancename* **depend= RpcSs/***sqlserverinstancename*

To remove the AOS dependency, type the following command:

sc config *AOSinstancename* **depend= RpcSs**

AOS service does not start

If the AOS service does not start automatically, you can attempt to start the service manually in **Control Panel** > **Administrative Tools** > **Services**. If the service still does not start, see the Windows System Event log for error details.

If the event log shows an Oracle error (Oracle 12154), then the TNS name being used to connect the Oracle client (the AOS) with the Oracle server is not contained in the tnsnames.ora file that resides with the Oracle client. Contact your Oracle database administrator for assistance.

Managing the client in Microsoft Dynamics AX

The Microsoft Dynamics AX client is an interface to Microsoft Dynamics AX data and functionality.

This section contains the following topics:

- Managing configurations (Client)
- Managing startup settings (Client)
- Manage Help files and updates
- Troubleshoot problems with Microsoft Dynamics AX clients

Managing configurations (Client)

A configuration is a group of startup and tracing settings for Microsoft Dynamics AX. Configurations are stored in the registry by default, but can also be stored as files. You can manage configurations for your local Microsoft Dynamics AX client or for the system-wide .NET Business Connector.

You may want to create new configurations to:

- Tune or troubleshoot Microsoft Dynamics AX. Change the Application Object Server (AOS) that a client connects to, or set tracing options.
- Move from a development environment to a production environment. Set a startup message and startup company, change the log directory, change the Microsoft Dynamics AX Application Object Server (AOS) that a client points to, encrypt client/server communication, and whether a system allows debugging.
- Save a configuration file for use in a broad deployment of clients.

The table below describes the topics in this section.

Section	Description
Configuration security (Client and Business Connector)	This topic describes how to manage security for client and business connector configurations.
Manage a client configuration	This topic describes how to manage a client configuration.
Manage a configuration for Business Connector	This topic describes how to manage a business connector configuration.

Configuration security (Client and Business Connector)

On a computer with the Microsoft Dynamics AX Configuration Utility installed, any user can view the settings for his or her own sessions.

On the computer running a client or running Business Connector with the Business Connector proxy user only a member of the local Windows Power User group or Administrators group can change configuration settings.

The following are recommended practices for using the Configuration Utility and configuration files securely.

- For clients, deploy the Configuration Utility only in development or test environments - in a production environment do not install it on client computers - instead, place a configuration file on a network shared folder, and use a shortcut to point to it. To install without the Microsoft Dynamics AX Configuration Utility you must perform a silent installation. For details, see "Mass deployment of the client" in the Microsoft Dynamics AX 2009 Implementation Guide.
- For non-administrator Microsoft Dynamics AX users, allow read-only access to the network shared folder that contains the configuration file to prevent accidental changes to the shared configuration file.

Manage a client configuration

If you are an Administrator or Power User on a computer running Microsoft Dynamics AX, you can manage the configurations for your local client through the Microsoft Dynamics AX Configuration Utility. Configuration options are stored in a file, or in the registry hive HKEY_CURRENT_USER/Software/Microsoft/Dynamics/4.0/Configuration/*Confignam*e.

Start the configuration utility

1. Open the configuration utility (**Start** > **Control Panel** > **Administrative Tools** > **Microsoft Dynamics AX Configuration Utility**).
2. In the **Configuration target** list, select **Local client**.

Create a new configuration

You cannot modify the original configuration of a system. To change a configuration, you must create a new one and modify it.

1. Click **Manage**, and then click **Create configuration**.
2. In the **Create configuration** dialog box, in the **Name** box, type a name.

3. Decide whether you want to copy settings from the active configuration or the original (default) configuration, and then click **OK**.

   Note:

   You cannot delete or rename the original configuration. You can only rename or delete configurations that you create.

Copy a configuration

1. In the **Configuration** list, select the configuration you would like to create a copy of.
2. Click **Manage**, and then click **Create configuration**:
3. In the **Create configuration** dialog box, in the **Name** box, type a name.
4. Click **Copy settings from the active configuration**, and then click **OK**.

Rename a configuration

1. In the **Configuration** list, select the configuration you would like to rename.
2. Click **Manage**, and then click **Rename configuration**.
3. In the **Rename configuration** dialog box, in the **New name** box, type a name, and then click **OK**.

Load a configuration

You can either load a configuration that is stored in the local registry or import a configuration file. Although you can use the **Open** command to view a saved configuration, Microsoft Dynamics AX does not store the settings from the opened file to the registry. To store saved configuration files to the registry, you must use the **Import** command.

Load a configuration from the registry

In the **Configuration** list, select the configuration you would like to open.

Import a configuration file

1. Click **Manage**, and then click **Import**.
2. Browse to the configuration file you would like to use (***.axc**), and open it.

Save or export a configuration

You can save startup settings for a client or Business Connector instance as a configuration stored in the registry, or as a configuration file. Saved startup settings enable you to:

- Tune Microsoft Dynamics AX. Change settings, and then tune your system by comparing performance with saved configurations that contain varied compression, database turning, and tracing settings.
- Move a configuration from one client to another.

Save a configuration to the registry

1. Verify that the currently selected configuration is the one you want to save.
2. Make any changes to the currently selected configuration that you want.
3. Click **Apply**, and then click **OK**.

Export a configuration to a file

Use this procedure if you want to copy a configuration to a client on another computer.

1. Verify that the currently selected configuration is the one you want to save.
2. Click **Manage**, and then click **Export configuration to a file**. Choose a location and name for the configuration file, and then click **Save**. The file is saved with an **.axc** extension.

Export all configurations to a file

Use this procedure if you want to copy all configurations from one client.

1. Verify that the currently selected configuration is the one you want to save.
2. Click **Manage**, and then click **Export All**. Choose a location and name for the configuration file, and then click **Save**. The files are saved with an **.axc** extension.

Save a configuration file with a new name

Use this procedure if you want to create a copy of the configuration file you have been using.

1. Import or open a configuration file.
2. Make changes to settings.
3. Click **Manage**, and then click **Save configuration file as**. Choose a location and name for the configuration file, and then click **Save**.

Delete a configuration

1. Verify that the currently selected configuration is the one you want to delete.
2. Click **Manage**, and then click **Delete configuration**.

Managing startup settings (Client)

You can use configurations and configuration files to change many of the settings that a Microsoft Dynamics AX client starts with. You can manage startup settings for your local Microsoft Dynamics AX client or for the system-wide .NET Business Connector. Changing startup settings can be particularly useful in a development environment. You may also want to create a configuration file to move from a development environment to a production environment.

The table below describes the topics in this section.

Section	Description
Change the log location	Provides a description of how to change the Microsoft Dynamics AX log location.
Set company to open on startup	Provides a description of how to set the company to open on startup.
Run commands at startup (Client)	Provides a description of how to run commands at the startup of Microsoft Dynamics AX.
Display message at startup	Provides a description of how to display a message at the startup of Microsoft Dynamics AX.
Connect a client to a different Application Object Server instance	Provides a description of how to connect a client to a different Application Object Server instance.
Enable printing from a computer running Application Object Server	Provides a description of how to enable printing from a computer running an Application Object Server.
Encrypt data	Provides a description of how to encrypt data sent between the Application Object Server and the client.

See Also

Managing configurations (Client)

Change the log location

You can change where Microsoft Dynamics AX places the log files by using the Microsoft Dynamics AX Configuration Utility.

1. Open the Configuration Utility (**Start** > **Control Panel** > **Administrative Tools** > **Microsoft Dynamics AX Configuration Utility**).

2. Verify that the currently open configuration target and configuration are the ones you want to modify.
3. On the **General** tab, in the **Log directory** box, type the path in which you want to store the log files, and then click **OK**.

Set company to open on startup

You can set Microsoft Dynamics AX to always open a particular company's data when it starts by using the Microsoft Dynamics AX Configuration Utility.

1. Open the Configuration Utility (**Start** > **Control Panel** > **Administrative Tools** > **Microsoft Dynamics AX Configuration Utility**).
2. Verify that the currently open configuration target and configuration are the ones you want to modify.
3. On the **General** tab, in the **Company** box, enter the three letter company identifier, and then click **OK**.

Run commands at startup (Client)

You can run a command when Microsoft Dynamics AX starts. You should exercise caution and test the commands you are using thoroughly in a development or test environment before using them in a production environment. Startup commands are not evaluated before being passed to Microsoft Dynamics AX and, if not correct, can cause the system to fail to start.

1. Open the Configuration Utility (**Start** > **Control Panel** > **Administrative Tools** > **Microsoft Dynamics AX Configuration Utility**).
2. Verify that the currently open configuration target and configuration are the ones you want to modify.
3. On the **General** tab, in the S**tartup command** box, type the command or commands you want to run at startup, and then click **OK**.

Display message at startup

You can set Microsoft Dynamics AX to display a message box when it starts by using the Microsoft Dynamics AX Configuration Utility.

To set a startup message

1. Open the Configuration Utility (**Start** > **Control Panel** > **Administrative Tools** > **Microsoft Dynamics AX Configuration Utility**).

2. Verify that the currently open configuration target and configuration are the ones you want to modify.
3. On the **General** tab, in the **Startup message** box, enter the message you want to display, and then click **OK**.

   When the client is started, the text displays in a message box, with an **OK** button below.

Connect a client to a different Application Object Server instance

You can connect a client to a different Application Object Server (AOS) instance from the Microsoft Dynamics AX Configuration Utility.

Connect to a different AOS instance

1. Open the Configuration Utility (**Start** > **Control Panel** > **Administrative Tools** > **Microsoft Dynamics AX Configuration Utility**).
2. Verify that the currently open configuration target and configuration are the ones you want to modify.
3. On the **Connection** tab, in the **Application Object Server instance** list, choose the AOS instance to connect to, and then click the **up** arrow until it is first in the list.

   Note:

   Microsoft Dynamics AX connects to the first AOS instance in the list unless it is unavailable, then it connects to the next instance in the list.
4. Click **OK**.

Add an instance to the list of available AOS instances

1. On the **Connection** tab, click **Add**.
2. In the **Application Object Server** box, type the name of the computer running the AOS instance you want to connect to.
3. Optional. In the **Instance name** box, type the name of the AOS instance.
4. In the **TCP/IP port** box, type the TCP/IP port that the AOS instance is running on, and then click **OK**.

Enable printing from a computer running Application Object Server

You can set Microsoft Dynamics AX to allow printing to a printer attached to a computer running an Application Object Server (AOS) instance. You may want to use this type of printing for reports or other data to which you want to restrict access. Options must be set on both the AOS and client to enable printing.

Note:

To print a document to a PDF file, you must have installed an Adobe Printer driver. For details, refer to your Adobe license and documentation (http://www.adobe.com)

Set an Application Object Server instance to enable printing

Note:

This option is set in the Microsoft Dynamics AX Server Configuration Utility.

1. Open the Server Configuration utility (**Start** > **Administrative Tools** > **Microsoft Dynamics AX Server Configuration Utility**).
2. Verify that the currently selected AOS instance and configuration are the ones you want to modify.
3. On the **Application Object Server** tab, click C**onnect to printers on this server**, and then click **OK**.

Set a client to use printing

Note:

This option is set in the Microsoft Dynamics AX Client Configuration Utility.

1. Open the Configuration utility (**Start** > **Control Panel** > **Administrative Tools** > **Microsoft Dynamics AX Client Configuration Utility**).
2. Verify that the currently selected configuration target and configuration are the ones you want to modify.
3. On the **Connection** tab, click **Connect to printers on the server**, and then click **OK**.

Manage Help files and updates

Microsoft Dynamics AX Help content is updated periodically to address feedback and product changes. This topic describes how to specify the location where Microsoft Dynamics AX client workstations access updated Help content and how to download updated Help content.

Specify where to locate updated Help content

You can specify a Web page or a file share where users on client workstations can download updated content. The default option is a Web page, but you might specify a file share if you want to manage Help content versions across multiple client workstations, or if you want to restrict Web downloads during peak business hours.

Use this procedure to configure a workstation to download updated Help content from a Web page or a file share.

1. In the Microsoft Dynamics AX client, click **Administration** > **Setup** > **Help Update parameters**.
2. Select the appropriate option and enter the location.
3. Close the form to save changes.

Download updated Help content

Updated Microsoft Dynamics AX Help files are available in English and a limited set of additional languages.

1. Find the most recent downloadable Help files on the Web.
2. Download and install the updated Help files to the location you specified earlier in this topic.
3. On each client workstation, click **Help** > **Check for Updates to the Help**.

Troubleshoot problems with Microsoft Dynamics AX clients

Use the following information to troubleshoot problems with Microsoft Dynamics AX clients.

Cannot connect to AOS

When a client cannot connect to an instance of the Application Object Server (AOS), one of the following problems may be occurring.

AOS is starting

The first time that you start a client after the AOS has been installed, the AOS service may be still be starting up. On the AOS computer, use the **Services** control panel (**Start** > **Administrative Tools** > **Services**) to determine whether the service has started.

Port is not open

If the AOS is installed on a computer with a firewall, make sure that the port that the AOS uses is open. The default port is 2712, but additional AOS instances will be installed on ports 2713, 2714, and so on.

Problems with Help

If Help is not working as expected, one of the following problems may be occurring.

Help files do not display correctly

If clients use a configuration file to point to an alternate Help location and the topics in the Help files do not appear, you may need to modify security settings that prevent Help from loading over a network. For details, see Knowledge Base article 896358. (Go to http://support.microsoft.com and search for **896358**.)

Cannot find the Help files for the required language

The Help files for the language that you require may not be installed. To install additional Help files, run the Microsoft Dynamics AX Setup program again, select to install the Client, and then select any additional languages that you require.

Managing instances of Business Connector

This section contains information about the Microsoft Dynamics AX Business Connector. The Business Connector enables applications to interact with Application Object Server instances. This section contains the following topics:

About the .NET Business Connector

The .NET Business Connector enables applications to interact with Application Object Server instances by providing a set of managed classes that facilitate access to X++ functionality in Microsoft Dynamics AX.

The Business Connector is installed and used with several components, including Enterprise Portal and the Application Integration Framework. Enterprise Portal, for example, allows users to access Microsoft Dynamics AX functionality and data using a Web browser. When a user connects to Enterprise Portal, the Business Connector is the key component that bridges Enterprise Portal and Microsoft Dynamics AX.

The .NET Business Connector can also be installed as a stand-alone component and used to develop third-party applications that integrate with Microsoft Dynamics AX.

To support integration with Microsoft Windows SharePoint Services and to enhance product security, the .NET Business Connector requires Windows integrated authentication.

Install and uninstall the Business Connector using **Setup.exe** on the Microsoft Dynamics AX DVD. For more information about installing the Business Connector, see the Microsoft Dynamics AX Installation Guide.

Set the .NET Business Connector proxy account

Some components require that the .NET Business Connector be configured to connect to Microsoft Dynamics AX with a proxy account. The use of a proxy enables the .NET Business Connector to connect on behalf of Microsoft Dynamics AX users when authenticating with an AOS instance.

The Business Connector proxy is a Microsoft Windows domain account that is configured from the initialization checklist, or in the **Administration** > **Setup** > **Security** > **System accounts** form.

Work with a system administrator to create a new account for the Business Connector before you install it. We recommend that the account be set up as follows:

- Must be a Windows domain account
- Must be a dedicated account (used only by Business Connector)
- Must have a password that does not expire
- Must not have interactive logon rights
- Must not be a Microsoft Dynamics AX user.

Important:

If a malicious user learns the Business Connector proxy credentials (name and password), that user could gain unauthorized access to sensitive information, and potentially damage the Microsoft Dynamics AX application. For this reason, only Microsoft Dynamics AX administrators should know the proxy credentials.

To set up and configure the Business Connector proxy, you must perform the following steps.

1. Create the proxy account in Active Directory.
2. Add the proxy account to the IIS local Windows group.
3. Configure the IIS application pool.
4. Install the .NET Business Connector.
5. Specify the Business Connector proxy user in Microsoft Dynamics AX.

Create the proxy account in Active Directory

1. Create a unique user in Active Directory in the form *domain\username*, for example, domain\bcproxy. This user must not have the same name as an existing Microsoft Dynamics AX user. For the procedure to add a new user, see the Active Directory documentation.
2. Assign a password to the user.
3. Select the **Password does not expire** option.
4. Select the **No interactive logon rights** option.
5. Close Active Directory.

Add the proxy account to the IIS local Windows group

For Web applications, you must add the Business Connector proxy account to the IIS local Windows group. If you are using Windows SharePoint Services, you must also add the account to the Windows SharePoint Services local Windows group.

1. Open the Computer Management application (**Start** > **Administrative Tools** > **Computer Management**).
2. Expand the Groups folder under Local Users and Groups.

3. Add the Business Connector proxy account to the following groups:
   - IIS_WPG (IIS Worker Process Group)
   - STS_WPG (STS Worker Process Group), if running Windows SharePoint Services

Specify the Business Connector proxy user in Microsoft Dynamics AX

1. Start Microsoft Dynamics AX (**Start** > **All Programs** > **Microsoft Dynamics** > **Microsoft Dynamics AX**).
2. Open the **System service accounts** form: **Administration** > **Setup** > **Security** > **System service accounts**.
3. In the **Business Connector Proxy** section of the form, enter the alias and the domain of the user.
4. Click **OK**.

Configuration security (Client and Business Connector)

On a computer with the Microsoft Dynamics AX Configuration Utility installed, any user can view the settings for his or her own sessions.

On the computer running a client or running Business Connector with the Business Connector proxy user only a member of the local Windows Power User group or Administrators group can change configuration settings.

The following are recommended practices for using the Configuration Utility and configuration files securely.

- For clients, deploy the Configuration Utility only in development or test environments - in a production environment do not install it on client computers - instead, place a configuration file on a network shared folder, and use a shortcut to point to it. To install without the Microsoft Dynamics AX Configuration Utility you must perform a silent installation. For details, see "Mass deployment of the client" in the Microsoft Dynamics AX 2009 Implementation Guide.
- For non-administrator Microsoft Dynamics AX users, allow read-only access to the network shared folder that contains the configuration file to prevent accidental changes to the shared configuration file.

Manage a configuration for Business Connector

To modify the configuration options for a Business Connector that is running non-interactively (not being run in conjunction with a client), you must be a member of the Administrators or Power Users group on the local computer.

Note:

Although the option to change the Business Connector non-interactive configuration becomes available when Business Connector is installed, if a Microsoft Dynamics AX client is also installed on the computer, the configuration settings for the client will be used for it and for the Business Connector if Business Connector is running under the currently logged-in account.

Start the configuration utility

1. Open the configuration utility (**Start** > **Control Panel** > **Administrative Tools** > **Microsoft Dynamics AX Configuration Utility**).
2. In the **Configuration target** list, select **Business Connector (non-interactive use only)**.

Create a new configuration

You cannot modify the original configuration of a system. To change a configuration, you must create a new one and modify it.

1. Click **Manage**, and then click **Create configuration**:
2. In the **Create configuration** dialog box, in the **Name** box, type a name.
3. Decide whether you want to copy settings from the active configuration or the original (default) configuration, and then click **OK**.

Copy a configuration

1. In the **Configuration** list, select the configuration you would like to create a copy of.
2. Click **Manage**, and then click **Create configuration**:
3. In the **Create configuration** dialog box, in the **Name** box, type a name.
4. Click **Copy settings from the active configuration**, and then click **OK**.

Rename a configuration

1. In the **Configuration** list, select the configuration you would like to rename.
2. Click **Manage**, and then click **Rename configuration**.
3. In the **Rename configuration** dialog box, in the **New name** box, type a name, and then click **OK**.

Load a configuration

You can either load a configuration that is stored in the local registry or import a configuration file. Although you can use the **Open** command to view a saved configuration, Microsoft Dynamics does not store the settings from the opened file to the registry. To store saved configuration files to the registry, you must use the **Import** command.

Load a configuration from the registry

In the **Configuration** list, select the configuration you would like to open.

Import a configuration file

1. Click **Manage**, and then click **Import**.
2. Browse to the configuration file you would like to use (***.axc**), and open it.

Save or export a configuration

You can save startup settings for a client or Business Connector instance as a configuration stored in the registry, or as a configuration file. Saved startup settings enable you to:

- Tune Microsoft Dynamics AX. Change settings, and then tune your system by comparing performance with saved configurations that contain varied compression, database turning, and tracing settings.
- Move a configuration from one client to another.

Save a configuration to the registry

1. Verify that the currently selected configuration is the one you want to save.
2. Make any changes to the currently selected configuration that you want.
3. Click **OK.**

Export a configuration to a file

Use this procedure if you want to copy a configuration to a client on another computer.

1. Verify that the currently selected configuration is the one you want to save.
2. Click **Manage**, and then click **Export configuration to a file**. Choose a location and name for the configuration file, and then click **Save**. The file is saved with an **.axc** extension.

Export all configurations to a file

Use this procedure if you want to copy all configurations from one client.

1. Verify that the currently selected configuration is the one you want to save.
2. Click **Manage**, and then click **Export All**. Choose a location and name for the configuration file, and then click **Save**. The files are saved with an **.axc** extension.

Save a configuration file with a new name

Use this procedure if you want to create a copy of the configuration file you have been using.

1. Import or open a configuration file.
2. Make changes to settings.
3. Click **Manage**, and then click **Save configuration file as**. Choose a location and name for the configuration file, and then click **Save**.

Delete a configuration

1. Verify that the currently selected configuration is the one you want to delete.
2. Click **Manage**, and then click **Delete configuration**.

Importing and exporting data

The following topics show how you can import and export data using Microsoft Excel worksheets or migrate data into Dynamics AX from another ERP system.

- Exporting to and importing from Microsoft Office Excel
- Importing and exporting large data files
- Importing base data and customized data

The Planning data migration section in the Microsoft Dynamics AX Administrator Help offers additional guidelines to follow when planning to import data into Microsoft Dynamics AX.

Securing data during export and import

When you export or import Microsoft Dynamics AX tables, store them in a folder that has appropriately-restricted permissions. Allowing access to the data from the following tables can potentially expose confidential or security information:

- SysConfig
- SysUserInfo
- AccessRightsList
- UserGroupInfo
- UserInfo
- UserGroupInfo

If you allow table data to be edited and then imported to the system, you risk providing increased access to the Microsoft Dynamics AX system.

Import and export definition groups

Definition groups define the tables and formats that are used for data export and import.

If you use the Excel template wizard, the definition group is created by the wizard.

You can create the following types of definition groups:

- **Standard** – Typical data export/import.
- **Excel** – Import using Microsoft Office Excel spreadsheets.
- **Custom** – Import using custom import facilities.

Create definition groups for import and export

Definition groups define the tables to export data from or import data to.

1. Click **Administration** > **Periodic** > **Data export/import** > **[EMenuItem: Display, SysExpImpGroup]**.
2. Press CTRL+N.
3. In the **Create table definition group** dialog box, type an identification (ID) and a name for the definition group.

   Note:

   To reset values on all tabs to their original settings, click **Default** in this dialog box. To clear all values on the **Options** and **Include table groups** tabs, click **Clear**.
4. Select a **Type**:
   - **Standard** – Normal data export/import.
   - **Excel** – Import using Microsoft Excel spreadsheets.
   - **Custom** – Import using custom import facilities.
5. Click the **Options** tab and select to export:
   - **Note** – Document management notes.
   - **Include system tables**
   - **Include cross-reference tables**
   - **Include database log tables**
   - **Include shared tables** – If set up as virtual companies, data shared between companies such as postal codes.
6. Click the **Include table groups** tab and decide which table groups should be included.

Configure tables for definition groups

1. Click **Administration** > **Periodic** > **Data export/import** > **Definition groups**.
2. Select the relevant definition group in the list, and then click the **Table setup** button to open the **Table setup** form.
3. The form looks different depending on the type of the definition group:
   - For **Standard** definition groups:

Tab page	Standard
Overview	Add or remove tables, and *for each table*: • Define whether notes are included when data is imported or exported. The **Note** option is relevant only if you add a document of the type **Note** by using the document management system. • Select whether a query should be used to limit the data that is exported from each table, and set up the query by selecting **Export criteria**.

- For **Excel** definition groups:

Tab page	Excel

Overview	Add or remove tables, and *for each table*:
	• In the **Export status** column, define how export is performed. You can select **Export to**, **Exported in part**, or **Exported in total**.
	Note: If the status is Exported in part or Exported in total, then the table will not be included in any later exports.
	• In the **Import status** column, define how import is performed. You can select **Delete and import**, **Import**, **Imported in part**, or **Imported in total**.
	Note: If the status is **Imported in part** or **Imported in total**, then the table will not be included in any later import**.**
	• In the **File name** column, select the file from which data is imported or to which data is exported.
	• In the **Use export criterion** column, select whether special criteria should be used for the export, for example only customer account numbers within a certain range or customers from certain ZIP codes.
	When it is selected, the actual criteria are defined by clicking the **Export criteria** button.
	• In the **Exclude table** column, select whether to exclude a table from export or import instead of removing it from the definition group.
	• In the **Validation level** column, select whether data should be validated, and if so, whether it is validated after each field or after each table.
	  • If **Table** is selected, then `validateWrite` is executed.
	  • If **Field** is selected, then `validateField` and `validateWrite` are executed.
	This tab also displays status after data has been exported or imported.

Conversion	Setting up this tab is optional. Add X++ code to manipulate data before it is added to tables in Microsoft Dynamics AX during import.
Import criteria	Setting up this tab is optional. Add X++ code to manipulate data before it is added to the current record in Microsoft Dynamics AX during import. This code is executed after any code on the Conversion tab.

Tab page	Excel
Log files	Setting up this tab is optional. This tab is a status page for import or export success. You can select whether to view a log file for information about data that was not imported.
Preview	Preview the data that is to be exported or imported by using the settings that are defined on the other tabs. If no file exists, you receive an error message.

- For **Custom** definition groups:

  Note:

  You can use **Custom** definition groups for import only.

Tab page	Custom

Overview	Add or remove tables, and *for each table*:
	• In the **Import status** column, define how import is performed. You can select **Delete and import**, **Import**, **Imported in part**, or **Imported in total**.
	Note: If the status is **Imported in part** or **Imported in total**, then the table will not be included in any later import.
	• In the **File name** column, select the file from which data is imported.
	• In the **Validation level** column, select whether data should be validated, and if so, whether it is validated after each field or after each table.
	  • If **Table** is selected, then `validateWrite` is executed.
	  • If **Field** is selected, then `validateField` and `validateWrite` are executed.
	This tab also displays status after data has been imported.

Tab page	Custom
General	• Use **Field separator** to specify how the fields are separated in the import file. If a separator is not defined, then the start and end positions must be defined by using **Field setup**. The number of fields in each record is automatically calculated when you select the field separator. • The **Validation level** field is the same as the one on the **Overview** tab. We recommend that you use either **Field** level or **Table** level validation. • In the **Unique field** drop-down list, select the field that is unique for each record. After a record with that identifier is imported, duplicate records are skipped.
Conversion	Setting up this tab is optional. Add X++ code to manipulate data before it is added to tables in Microsoft Dynamics AX during import.
Import criteria	Setting up this tab is optional. Add X++ code to manipulate data before it is added to the current record in Microsoft Dynamics AX during import. This code is executed after any code on the **Conversion** tab.
Log files	Setting up this tab is optional. This tab is a status page for import success. You can select whether to view a log file for information about data that was not imported.
Preview	Preview the data that is to be exported or imported by using the settings that are defined on the other tabs. If no file exists, you receive an error message.

4. To set up individual fields for each table, click the **Field setup** button.

   The form looks different depending on the type of the definition group. For complete guidelines about how to set up individual fields, click the Help button in the **Field setup** form.
5. Close the form to save the setup.
6. To start the actual export or import, click **Export to** or **Import** on the **Data export/import** dialog box (**Administration** > **Periodic** > **Data export/import**).

Exporting data

To determine the best method for exporting data, consider the purpose of the export, the amount of data, and the type of the data that you want to export. Microsoft Dynamics AX provides the following export mechanisms:

- Export default data
- Export data to Microsoft Office Excel
- Export standard data

Export default data

Default data is customer-independent data, such as address formats, time intervals, and unit conversions. Use this method to export non-company specific data into a data file (*.dat) and a definition file (*.def). The data file contains the data that you have exported. The definition file contains information about table name and ID, field name and ID, record ID information, transaction IDs, and company account.

A table is part of the default data when its `TableContents` property is set to `Default data`.

The export is handled by the **`SysDefaultDataExport`** class that extends the **`SysDefaultDataExportBase`** class. For a listing of the tables defined as default data tables, see the export method on the **`SysDefaultDataExportBase`** class.

If tables are to be added to the list, use the **`exportPost`** method on the **`SysDefaultDataExportBase`** class.

Tip:

Setting up this kind of information is a prerequisite for all Microsoft Dynamics AX installations. After you set up one Microsoft Dynamics AX installation, use the **Export default data** form to export default data. Use the export files as a basis to import into new installations.

Export default data

1. Make sure that the current company account is the one that you want to export data from.

   Data is exported from the current company. However, data that is not company specific is included in the export.

2. Click **Administration** > **Periodic** > **Data export/import** > **Default data** > **Export to**.

3. In the **File name** field, use the browse button to select a location for the file that contains the export data, and type a file name.

   Important:

   - Be sure to save exported files in a folder that has appropriately restricted permissions. In particular, allowing access to the data from the following tables can potentially expose confidential or security information: SysConfig, SysUserInfo, AccessRightsList, UserGroupInfo, UserInfo, and UserGroupInfo.
   - If unsecured data from these tables is edited and then imported, you run the risk of potentially providing increased access to the Microsoft Dynamics AX system.

4. Select the **Note** check box to export document management notes.
5. Click **OK**.

The results of the export is a data file (*.dat) and a definition file (*.def). If you import the data file and the definition file to another Microsoft Dynamics AX installation, these files must be in the same folder.

See Also

Import default data

Export data to Microsoft Office Excel

This topic describes how to populate the Microsoft Office Excel template that is generated in Microsoft Dynamics AX.

This method enables you to export populated or empty Microsoft Office Excel (*.xls) files (templates). Empty *.xls files can be populated in Microsoft Office Excel and imported back into Microsoft Dynamics AX. Populated .xls files are populated with data from Microsoft Dynamics AX and then exported as *.xls files. Both empty and populated *.xls files that are exported from Microsoft Dynamics AX contain information about the table identification (ID), field name, table name, header notes, and Help text from related fields.

Important:

Export to Microsoft Office Excel shares the Excel worksheet size limitations of 65,536 rows by 256 columns.

Export data to Excel

1. Use the **Microsoft Office Excel Template Wizard** to create templates for the required Microsoft Dynamics AX tables, as described in Create a template in Microsoft Office Excel.

   The **Microsoft Office Excel Template Wizard** helps you create one or more templates in Microsoft Office Excel where you can enter data, and then import the data into Microsoft Dynamics AX.

   The wizard creates a worksheet in the workbook for each table that you select.

   The wizard then lets you select whether to include current company data in the export, export supporting data, and create a Microsoft Office Excel project file.
2. Open the Excel workbook that contains the templates and review the data.

Export standard data

The standard data format is generally used when you are exporting large amounts of data. This method supports the fast export of large amounts of data.

Use standard data format to export large binary or comma-separated files. The standard export yields two files: a data file (*.dat) and a definition file (*.def). The data file contains the data that you have exported. The definition file contains information about table name and ID, field name and ID, record ID information, transaction IDs, and company account.

Before you can export standard data, you must first create a definition group as described in Create definition groups for import and export.

Export standard data

1. Click **Administration** > **Periodic** > **Data export/import** > **[EMenuItem: Display, SysExpImpGroup]**.
2. Select the definition group that you want to export, and then click **Export to**.

   The **Export options** dialog box opens.
3. For **File name**, use the browse button to select a location for the file that contains the export data, and type a file name.

   Important:

   - Be sure to save exported files in a folder that has appropriately restricted permissions. In particular, allowing access to the data from the following tables can potentially expose confidential or security information: SysConfig, SysUserInfo, AccessRightsList, UserGroupInfo, UserInfo, and UserGroupInfo.
   - If unsecured data from these tables is edited and then imported, you run the risk of potentially providing increased access to the Microsoft Dynamics AX system.

4. Select the file type:
   - **Binary** – Use a compressed file format. This option is used most frequently if the data will be imported back into Microsoft Dynamics AX.
   - **Comma** – Do not include container fields in the export. This comma-separated format is typically used to export data for viewing in a spreadsheet.
5. Select the **Execute on AOS** check box to have the report executed by the AOS server instead of by the client.

   This option typically improves performance.
6. Click the **OK** button to start the export.

Importing data

To determine the best method for importing data, consider the purpose of the import, the amount of data, and the type of the data that you want to import. Microsoft Dynamics AX provides the following import mechanisms:

- Import default data
- Import custom data and data from other systems
- Create a template in Microsoft Office Excel
- Enter data in Microsoft Office Excel
- Import data from Microsoft Office Excel

Import default data

Default data is customer-independent data, such as address formats, time intervals, and unit conversions.

Tip:

Setting up this kind of information is a prerequisite for all Microsoft Dynamics AX installations. After you set up one Microsoft Dynamics AX installation, use the **Export default data** form to export default data. Use the export files as a basis to import into new installations.

Importing default data is one of the first steps that you should complete when you set up a Microsoft Dynamics AX implementation. However, you can import default data at a later stage.

If you try to import data into tables that already hold data, the import wizard issues a warning. You can choose to overwrite the existing data or append the imported data to the existing data. We recommend that you do not delete existing data.

Note:

The import typically includes data that is not company-specific, such as user group and user information. If there is more than one company, importing this data also affects other companies.

Import default data

1. Make sure that the current company account is the one that you want to import data into.
2. Click **Administration** > **Periodic** > **Data export/import** > **Default data** > **Import Wizard**.
3. In the **Import default data** wizard, click **Next** to begin the wizard.
4. In the **File name** page:
   a. Locate the file that contains the default data.
   b. Select the **Execute on AOS** check box to have the import executed by the AOS server instead of by the client.

      This option will typically improve performance.
   c. If you want to import data into tables that are not company accounts-specific, select **Include system and shared tables**.
   d. Click **Next** to move to the next page.
5. Click **Next** in the **Generate table list** page.

   The wizard then provides an overview of the tables in the import file.
6. In the **Select tables** page, you can clear tables that should not be imported. The import is not performed until you acknowledge that the tables presented are the correct set to import.

   When you have made a selection, click **Next** to move to the next page.

   Note:

   If you import a text field from the Microsoft Office Excel spreadsheet with an integer greater than 2.1 million, it is not imported into Microsoft Dynamics AX, and the column displays in red. Use an integer less than 2.1 million.
7. Click **Next** in the **Generate table list** page.
8. In the **Select tables to be deleted** page, select the tables to delete from the Microsoft Dynamics AX system before you import the default data.

   This page shows a list of tables that contain records. By selecting a table, the records will be replaced by the imported data. If you do not select to delete the table, then you will not import default data to these tables.

   When you have made your selection, click **Next** to move to the next page.
9. Click **Finish** to save your changes and start the import, or click **Cancel** to exit the wizard.

See Also

Export default data

Import custom data and data from other systems

Import data that has been exported from any other business management application. As a prerequisite for data import, you must have one or more files holding data that have been exported from another system.

Note:

A period must be used as the decimal symbol to be recognized by Microsoft Dynamics AX.

Working with data import

1. Make sure that the current company account is the one that you want to import data into.

   Note:

   Do not import data into the non-company specific DAT company account.
2. Click **Administration** > **Periodic** > **Data export/import** > **Definition groups**.
3. Select a definition group and then click **Table setup** to set import options, as described in Configure tables for definition groups, and close the form.

   Note:

   Before actually performing an import, click the **Preview** tab on the **Table setup** form to preview the result of the import.
4. In the **Definition groups** form, select the definition group, and then click **Import**.
5. When the import is complete, the Infolog provides information about the number of records that have been imported. The **Table setup** form also shows import information about the **Log files** tab.

See Also

Import data from Microsoft Office Excel

Create a template in Microsoft Office Excel

1. Click **Administration** > **Periodic** > **Data export/import** > **Excel spreadsheets** > **Template Wizard**.

   Read the first page, and then click **Next >**.
2. Type a name for the Microsoft Office Excel workbook and include the full path, or browse to locate an existing Microsoft Office Excel workbook, and then click **Next >**.

   Note:

   If you select an existing file, the content is overwritten.

3. On the **Select tables** page, select the tables to include in the workbook:
   - Include an available object by selecting the table in the **Available objects** pane, and clicking the (**>**) directional arrow. Press CTRL to select multiple tables.

     If there are tables that you want to use in the template but they do not appear in the **Available objects** pane, then select **Show all tables**.
   - Remove selected objects by highlighting the table in the **Selected objects** pane, and clicking the (**<**) directional arrow. Press CTRL to select multiple tables.
   - Remove all previously selected objects by clicking the (**<<**) directional arrow.
4. Click **Next >** to generate the field list on the **Generate field list** page, and then click **Next >** again to select the fields that you want to use.
5. On the **Select fields** page, select the fields from the tables that you want to be shown in the template.

   The shaded check boxes indicate that a field is either mandatory or part of a unique index and therefore necessary to maintain data consistency. Fields marked with a yellow padlock are system fields and are not selected by default.
6. On the **Import definition group** page, select **Create import definition group?** to create an import definition group based on the template.

   A definition group contains definitions for each worksheet in a workbook and is used when importing the workbook to Microsoft Dynamics AX.

   The definition group is called "EXL00000xx" where "xx" is a consecutive number.
7. On the **Export data** page, select which of the following actions you want to take:
   - **Export data** – Export data from the current company to the Excel workbook.
   - **Create supporting tables worksheet** – Include supporting tables. Supporting tables are typically populated with data.
   - **Create a Microsoft Office Excel project file** – Create an Excel project file. The project file references the exported Excel workbook.
8. Click **Finish** to complete the wizard.

   Note:

   By default, the **View workbook after creation?** field is selected and the workbook opens after the wizard is completed. Clear this option if you do not want to open the workbook now.

Enter data in Microsoft Office Excel

Data for the Microsoft Office Excel worksheets can be entered manually or imported from another system.

Important:

The template cannot be changed. If it is changed, the import will fail.

Format details

- For each Microsoft Dynamics AX table selected, a template is presented on a separate worksheet.
- Field names are displayed in the first row.
- Fields occur in the same order that they appear in the table. Do not change this order. If you change the order, the data will not import correctly.
- Array field names contain the symbol '@'. For example, *Dimension@Department*, *Dimension@Cost center*, and *Dimension@Purpose*.
- Mandatory fields in Microsoft Dynamics AX have their corresponding columns in Microsoft Office Excel highlighted in yellow.

Working with data in Microsoft Office Excel

Data import from Microsoft Office Excel into Microsoft Dynamics AX is optimal when the data types in the Excel worksheet match the ones in the Microsoft Dynamics AX fields. When a template based on Microsoft Dynamics AX is created, the cells in the worksheet have an Excel format called 'Text'. The contents of cells with text format are treated as text even when a number is in the cell. This format is useful because numbers frequently contain spaces, parentheses, or dashes, which can produce poor data if they are saved in the 'Number' format.

When Excel data is imported, Microsoft Dynamics AX converts the data to the required format only if the entered data type is compatible with the required type. For example, a string type entered in an Excel cell that should be imported into a Microsoft Dynamics AX field of integer type is not imported into Microsoft Dynamics AX. This also means that wherever enum values are expected, the actual string value must be entered instead of the representative integers.

Examples

- If you enter telephone numbers, Telex numbers, swift numbers, postal (ZIP) codes or other numbers that contain parentheses and dashes, Microsoft Dynamics AX accepts them as strings.

- If you enter a number such as 10000 into a field where a real number is expected, Microsoft Dynamics AX converts it to 10,000.00 by inserting the separators. However, if you enter a string, such as "abc" in the same field, the data is not imported.
- To import the `DayWeekMonth` enum with the value Week, you must enter 'Week' in the Excel worksheet.

Notes:

- You cannot import data entered in Excel in a format that is incompatible with the data type in Microsoft Dynamics AX.
- Mandatory rows left blank in Excel that are imported into Microsoft Dynamics AX integer fields are "zero-filled" in Microsoft Dynamics AX. The best way to avoid unwanted data in Microsoft Dynamics AX is to enter valid data in all yellow-highlighted columns in the worksheet.

Import data from Microsoft Office Excel

Data can be easily imported from Microsoft Office Excel by using the import feature of Microsoft Dynamics AX. To reduce the potential of error, the import feature include a wizard that generates Microsoft Office Excel workbook templates and a dialog box that controls the import of data into Microsoft Dynamics AX after the data has been entered into the workbook.

Note:

Microsoft Windows Regional options date format and the date format in Microsoft Office Excel on the computer that is performing the import must be the same. If the formats are not the same, the date field will be empty.

Import data from Microsoft Excel

1. Click **Administration** > **Periodic** > **Data export/import** > **Excel spreadsheets** > **Template Wizard**.

   The **Microsoft Office Excel Template Wizard** helps you create one or more templates in Microsoft Office Excel where you can enter data, and then import the data into Microsoft Dynamics AX.

   The wizard creates a worksheet in the workbook for each table that you select.

   Complete the wizard, as described in Create a template in Microsoft Office Excel. While completing the wizard, select to create a definition group.
2. Open the Excel workbook that contains the templates and enter necessary data, as described in Enter data in Microsoft Office Excel.

   Note:

   Protect data by adding password protection to the worksheet or workbook using Excel's password protection feature.

3. Click **Administration** > **Periodic** > **Data export/import** > **Default data** > **Definition group** and select the definition group that was created by the Template wizard. Click **Import**.

   The **Microsoft Office Excel import** dialog box appears.

4. Select the Excel workbook to import data from and then click **OK**.

   Based on the worksheet name, the import allocates the contents of each worksheet into the corresponding table in Microsoft Dynamics AX.

Tips for importing data from Excel

- After you create a Microsoft Office Excel template using the Template wizard, you can import data from third-party systems into Excel and from Excel into the Microsoft Dynamics AX template. The easiest way to do this is:
  - Put the data from your third-party system into a different Excel spreadsheet.
  - Map the contents of the Excel spreadsheet to the Microsoft Dynamics AX template. The mapped contents appear in the template spreadsheet.
- The Microsoft Office Excel template lists each field to be imported into Microsoft Dynamics AX. When populating the Excel spreadsheet, many fields are associated with another Microsoft Dynamics AX table. Therefore, make sure that you use the correct codes.

  For example, Customer groups may be defined as DOM = Domestic and FOR = Foreign; DOM or FOR should be entered on the spreadsheet. The values for these types of user-defined fields are not available from Excel. However, lists of the setup values can be printed from Microsoft Dynamics AX.
- Data import from Microsoft Office Excel into Microsoft Dynamics AX is optimal when the data types in the Excel worksheet match the ones in the Microsoft Dynamics AX fields. When a template based on Microsoft Dynamics AX is created, the cells in the worksheet have an Excel format called 'Text'. The contents of cells with text format are treated as text even when a number is in the cell. This format is useful because numbers frequently contain spaces, parentheses, or dashes. This can produce poor data if they are saved in the 'Number' format.

  When Excel data is imported, Microsoft Dynamics AX converts the data to the required format only if the entered data type is compatible with the required type. For example, a string type entered in an Excel cell that should be imported into a Microsoft Dynamics AX field of integer type is not imported into Microsoft Dynamics AX. This also means that wherever enum values are expected, the actual string value must be entered instead of the representative integers.

- Begin entering data starting at row 7.

  Caution:

  Rows 2 through 6 in the workbook are used to map the fields back to Microsoft Dynamics AX and are hidden deliberately. Do not delete them.

  - Verify that all required fields are populated before importing.
  - Complete any additional fields that your old system may not have had available.
  - Edit records as necessary to clean up the database.

    Note:

    Remember that Microsoft Dynamics AX has already created the mapping of this template to the Microsoft Dynamics AX database. Therefore, do not delete or rearrange columns. If you are no longer using a column, hide it.

- Before populating the whole spreadsheet, try importing a few records to verify the import is working correctly. Verify the import status before re-importing. The status is used to determine whether the re-import should replace or append the existing records. Check the status by clicking the **Table setup** button in the **Definition** group.

See Also

Create a template in Microsoft Office Excel

Enter data in Microsoft Office Excel

Import users from Active Directory

To add an additional layer of security to your computing environment, Microsoft Dynamics AX requires that all users be listed in Active Directory directory services on your domain controller before they can be enabled on the Microsoft Dynamics AX**User** form. If users are not enabled on this form, they cannot access Microsoft Dynamics AX.

Active Directory for Microsoft Windows catalogs information about all the objects on a network, including people, computers, and printers, and distributes that information throughout the network. Security is integrated with Active Directory through logon authentication and access control. Active Directory is a feature of Microsoft Windows Server 2003 and Microsoft Windows Server 2000. For more information, see Windows Server 2003 Active Directory or Windows Server 2000 Active Directory. For more information about how to implement Active Directory with Microsoft Dynamics AX, see the Microsoft Dynamics AX Implementation Guide.

Note:

Existing Active Directory structures do not require modifications to be used to support Microsoft Dynamics AX users within the domain. If your customer has a site with Active Directory domains, and all the domains in the forest are set up with two-way trust, the application will recognize all the users in the domain as soon as they have been imported.

After a user is listed in Active Directory, you can add that user to Microsoft Dynamics AX manually or you can import multiple Active Directory users into Microsoft Dynamics AX using the procedure included in this topic. (For more information about adding users manually, see the System and Application Setup Help, available from the Microsoft Dynamics AX help menu.)

Administrator permissions

There is no requirement for the Microsoft Dynamics AX administrator to be a Windows domain administrator to import users from Active Directory.

When a domain administrator in Active Directory is logged in to Microsoft Dynamics AX as a Microsoft Dynamics AX administrator and tries to import Active Directory users, the administrator can see all users in Active Directory and can import them into Microsoft Dynamics AX successfully.

If a Microsoft Dynamics AX administrator who is not a domain administrator in Active Directory tries to import Active Directory users, only a subset of the users in Active Directory will appear. This occurs because of security functionality in the Active Directory Group Policy Objects (GPO).

To allow Microsoft Dynamics AX administrators rights to Active Directory, you must grant **Authenticated users** security group membership to the Microsoft Dynamics AX administrators. They can then see the complete list of Active Directory users during import.

Import users from Active Directory

1. From a Microsoft Dynamics AX client, click **Administration** > **Users**.
2. On the **Overview** tab, click **Import** to access the **Active Directory Import Wizard**.
3. Complete the wizard.

Alias ID duplicates

When you import users from Active Directory into Microsoft Dynamics AX, the wizard tries to create Microsoft Dynamics AX users by creating Microsoft Dynamics AX user IDs from the Active Directory aliases. But, Microsoft Dynamics AX user IDs are limited to five characters, whereas the Active Directory alias can be up to 255 characters. If the first five characters of the Active Directory alias are the same for more than one user, then the wizard then generates alternative Microsoft Dynamics AX user IDs for these users and displays them.

When generating alternative user IDs, if the user alias has more than five characters, then the first four characters from the first name and a single character from the last name are used. If there are still duplicates, then the first three characters of the first name and two characters from the last name are used.

You can change any of the user IDs.

When you approve the new user IDs, the users are created in Microsoft Dynamics AX.

Import data on startup

You can import data into Microsoft Dynamics AX without user interaction by starting Microsoft Dynamics AX from the command line with an XML file as an input parameter. The XML file specifies the data to be imported. Results are written to a log file or shown in the Infolog.

Running the import process without user interaction is particularly useful in test environments.

Create an XML file

1. On the Microsoft Dynamics AX client computer, create an XML file.
2. Add tags to the XML file by entering the following parameters:
   - Microsoft Dynamics AX version.
   - Name and location of the log file.
   - The data file to be imported, along with the appropriate attributes.

     Note:

     The XML file syntax is described in the documentation for the **SysAutoRun** class. For more information, see the Developer Help, available from the Microsoft Dynamics AX Help menu. In addition, see the sample file in the following section.
3. Open a command prompt and type the following:

```
ax32.exe -StartupCmd=AutoRun_c:\PathToFile\FileName.XML
```

Sample XML input file

```
  <?xml version="1.0" encoding="utf-8" ?>
<AxaptaAutoRun exitWhenDone="true" version="5.0" logFile="D:\AX\axautodata.log">
<CompanyAccounts>
  <Company name="Demo Company" id="DMO" overwrite="true" />
  </CompanyAccounts>
  <DataImport companyId="DMO" file="\\ServerName\ShareName\DemoData.dat" />
  </AxaptaAutoRun>
```

Migrating data

This section contains information about how to migrate data from your current system to Microsoft Dynamics AX. The section contains the following topics:

- Plan data migration
- Migrate customer, vendor, and item data by using the Excel Template Wizard
- Migrate historical transaction data
- Migrate open transactions

Plan data migration

If you plan to migrate data to Microsoft Dynamics AX from another enterprise resource planning (ERP) system, planning is critical. It is impossible to offer a step-by-step guide for planning data migration. However, you should consider the following issues when planning the process.

- Create a backup before importing any data into Microsoft Dynamics AX. The system administrator should create a hard copy backup. The Microsoft Dynamics AX administrator can create an online backup in Microsoft Dynamics AX by copying the company's information to a newly created company.
- Determine whether data can be exported from the current system, and determine how to export data.

  If the data can be exported into a Microsoft Office Excel workbook, then Microsoft Dynamics AX Excel templates can be used to import the data into Microsoft Dynamics AX. For more information about importing by using the Excel Template Wizard, see Create a template in Microsoft Office Excel.

  If the current data cannot be exported into a workbook, then you can create a custom import file, or you can enter data manually. A custom import file allows you to define criteria, set delimiters, and map each field to Microsoft Dynamics AX.

- Determine whether you will import master data (customers, vendors, items, and ledger accounts). Required setup for master records must be completed before you import data.

  For more information, see Migrate customer, vendor, and item data by using the Excel Template Wizard.
- Financial data is imported into a Microsoft Dynamics AX journal, but the data is not imported directly into data files.

  Because of the complex file structures in Microsoft Dynamics AX, financial data should never be imported directly into the data file, because this action will cause inconsistencies in other files.

  After the financial data has been converted into a journal, the journal can be reviewed and posted. When the journal is posted, all of the necessary Microsoft Dynamics AX files are updated.
- Almost any table can be imported into Microsoft Dynamics AX, but it may be easier or more effective to enter some tables manually.

  For example, many of the setup forms, such as **Terms of payment**, do not have many records, so it is faster to enter them manually.
- Consider cleaning up the database. For example, determine whether old records can be deleted or archived, whether the current database contains duplicate records, and whether you want to change numbering schemes.

Migrate customer, vendor, and item data by using the Excel Template Wizard

Before you import master data (such as customer, vendor, and item records) into Microsoft Dynamics AX, complete the required setup for the records that you will import. Because field names and values in Microsoft Dynamics AX may be different than those in the current system, we recommend that you create a record manually to understand Microsoft Dynamics AX fields and how they correspond to fields in your current system.

After you decide how fields and values from the old system correspond to the fields and values in Microsoft Dynamics AX, use the **Template Wizard** to create a Microsoft Office Excel template.

Do not convert all records at once or attempt to work with a blank template. Instead, we recommend that you create a few records in Microsoft Dynamics AX and run though the conversion process. Ensure that the template meets your needs and that the template converts the data as you expect it to. As you create the template by using the wizard, verify the template by populating it with a few records and importing the data. If the template does not meet your expectations, modify it and try again.

Start the **Microsoft Office Excel Template Wizard** by clicking **Administration** > **Periodic** > **Data export/import** > **Excel spreadsheets** > **Template Wizard**.

Create a vendor template by using the template wizard

1. Start the **Microsoft Office Excel Template Wizard**.
2. Create a workbook called *Vendor*s, and then click **Next**.
3. Select **VendTable** from the list of available objects, and then click **Next**.
4. After the list of fields has been generated, click **Next**.
5. Select the fields to use in the template, and then click **Next**.

   Note:

   Mandatory fields are marked with a red padlock symbol and cannot be removed from the list.
6. Select whether you want to create an import definition group for the workbook that contains the template, and then click **Next**.

   The import definition group contains a definition for each worksheet in the workbook. The definition group can be used when you import the workbook.
7. Select whether you want to export data to the workbook, and then click **Next**.
8. Click **Finish**.
9. Use Excel to map Microsoft Dynamics AX fields to those in the current system and to populate the template with data.

Create a customer template by using the template wizard

1. Start the **Microsoft Office Excel Template Wizard**.
2. Create a template called *Customers,* and then click **Next**.
3. Select **CustTable** from the list of available objects, and then click **Next**.
4. Select the fields to use in the template, and then click **Next**.

   Note:

   Mandatory fields are marked with a red padlock symbol and cannot be removed from the list.
5. Select whether you want to create an import definition group for the workbook that contains the template, and then click **Next**.

   The import definition group contains a definition for each worksheet in the workbook. The definition group can be used when you import the workbook.
6. Select whether you want to export data to the workbook, and then click **Next**.
7. Click **Finish**.
8. Use Excel to map Microsoft Dynamics AX fields to those in the current system and to populate the template with data.

Create inventory templates by using the template wizard

For inventory records, you must create three templates: Item table, Inventory module parameters, and Warehouse items.

1. Start the **Microsoft Office Excel Template Wizard**.
2. Enter the name of the template that you are creating, and then click **Next**.
3. Select the table to create a template for.
   - If you are creating the Item table template, select **InventTable** from the list of available objects, and then click **Next**.
   - If you are creating the Inventory module parameters template, select **InventTableModule** from the list of available objects, and then click **Next**.
   - If you are creating the Warehouse items template, select **InventItemLocation** from the list of available objects, and then click **Next**.
4. Select the fields to use in the template, and then click **Next**.

   Important For the Inventory module parameters template, make sure that the following objects are selected: **Invent**, **Purch**, and **Sales**. Mandatory fields are marked with a red padlock symbol and cannot be removed from the list.
5. Select whether you want to create an import definition group for the workbook that contains the template, and then click **Next**.

   The import definition group contains a definition for each worksheet in the workbook. The definition group can be used when you import the workbook.
6. Select whether you want to export data to the workbook, and then click **Next**.
7. Click **Finish**.
8. Use Excel to map Microsoft Dynamics AX fields to those in the current system and to populate the template with data.

See Also

Import data from Microsoft Office Excel

Enter data in Microsoft Office Excel

Migrate historical transaction data

We recommend that you import transaction history data for General Ledger balances only. To convert transaction histories for the subsidiary modules would be time-consuming, because it is likely that the current system does not have the same fields, file layouts, and posting methodologies as Microsoft Dynamics AX. We recommend that you maintain either a reference version of your current system or hard copy records to preserve your history and transaction details.

Migrate open transactions

Open transactions are unpaid or partly settled customer and vendor invoices that exist before you migrate to Microsoft Dynamics AX. How you process open transactions depends on the needs of your business.

The following information provides guidelines about how to process open transactions. You may be required to transfer more information or less information, depending on local accounting rules.

Important:

Because the transactions were not created in Microsoft Dynamics AX, you will lose all historical references to transactions. For more information about historical data in a migration, see Migrate historical transaction data.

Consider the following guidelines when you work with open transactions:

- Because open transactions must be processed manually, try to minimize the number of open transactions before migrating to Microsoft Dynamics AX. Settle as many accounts receivable and accounts payable as possible before transferring transactions.
- Partly settled invoices should indicate only the original total amount, represented as a text string, and the current open balance.
- Create journals for customers and vendors, and enter open transactions by using one line for each unsettled invoice. Post directly to the Accounts Receivable/Accounts Payable (AR/AP) accounts. In addition, use the AR/AP ledger control account as the offset account.

  This method of creating journals allows each journal line to be balanced and the net AR/AP ledger control account posting to be 0 (zero). This method requires the General Ledger opening balance to include the control account totals.

See Also

Migrate customer, vendor, and item data by using the Excel Template Wizard

Configure and manage AIF

Application Integration Framework (AIF) enables companies to integrate Microsoft Dynamics AX and communicate with external business processes and partners through the exchange of XML over various transport media. AIF enables both business-to-business and application-to-application integration scenarios.

This information is for IT staff and administrators who are responsible for installing, configuring, and troubleshooting AIF integrations with external systems. In this guide, you will find the following information.

Topic	Description
Exchanging documents electronically using AIF	General information about AIF data exchange including topics about planning, security, and troubleshooting.
Adapter-based exchanges in AIF	Information on configuring and maintaining data exchanges using one of the AIF adapters: file system, Message Queuing (MSMQ), or BizTalk.
Web services-based exchanges in AIF	Information on configuring and maintaining data exchanges using the AIF Web services.
Manage document exchanges in AIF	Covers how to maintain AIF integrations and research problems when they arise.

For more information about AIF for software developers, see "Integrating Other Applications with Microsoft Dynamics AX" in the Microsoft Dynamics AX SDK Help.

See Also

What's new in Application Integration Framework (AIF)

Exchanging documents electronically using AIF

The ability to integrate Microsoft Dynamics AX with other systems inside and outside the enterprise is a common requirement. Application Integration Framework (AIF) is the framework that enables integration through the exchange of data through formatted XML. This formatted XML is referred to as a document, and each document contains defined data and business logic.

In AIF, data is exchanged with external systems through electronic documents. An exchange starts with a document (based on a document class) defined by using Microsoft Dynamics AX business logic.

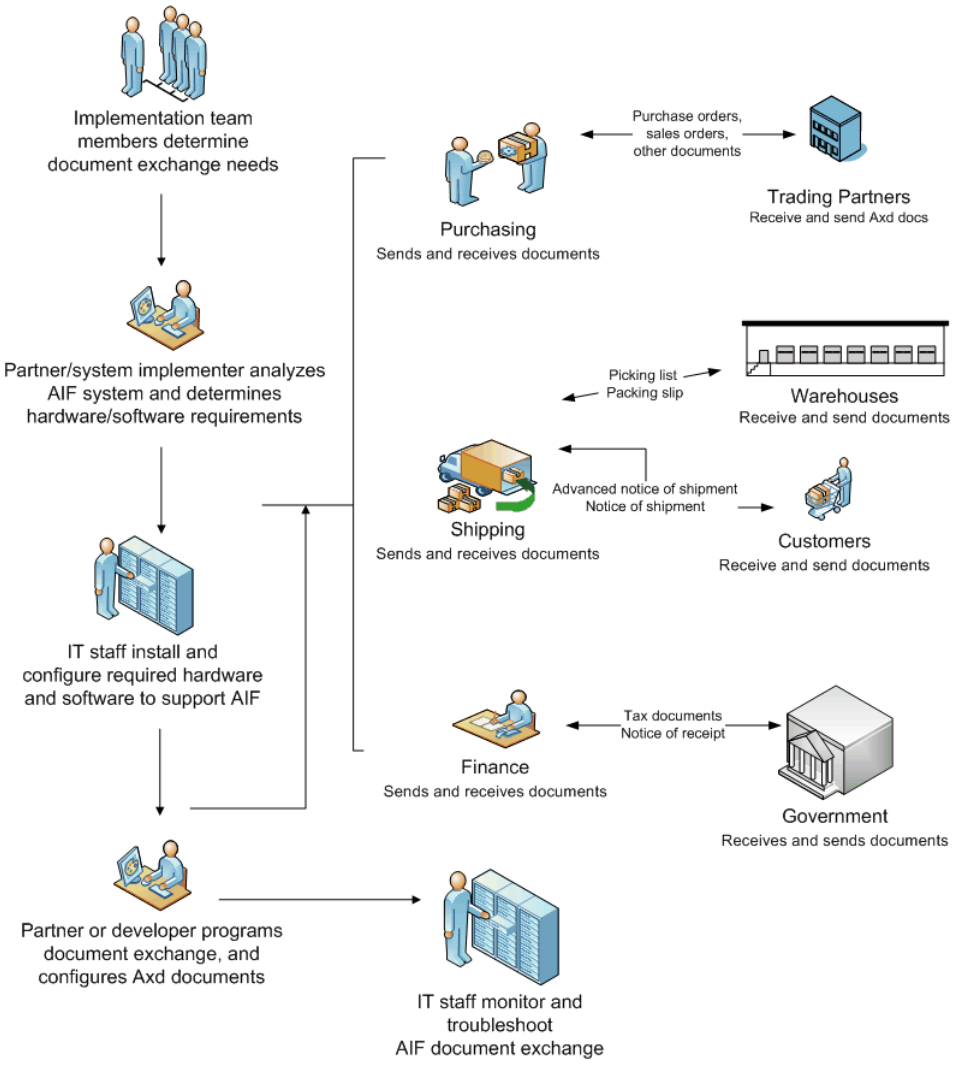
Microsoft Dynamics AX ships with over 70 standard documents that support common business processes. AIF also provides the ability to customize existing documents or create your own documents. For more information about the standard documents, see topic "Documents that ship with Microsoft Dynamics AX" in the Microsoft Dynamics AX SDK Help. For developer information about how to create new AIF documents, see topic "Creating New Documents" in the Microsoft Dynamics AX SDK Help.

Integration Process

Whether you require integration with internal legacy systems or external trading partners, the integration process involves common key steps:

1. In a typical integration scenario, users who have business expertise first determine the document exchange needs. These are requirements from a business perspective. The business users work with the implementation team to specify:
   - What data is to be exchanged
   - Any business logic related to that data
   - The external systems with which data is to be exchanged
   - The conditions under which data is sent from or received by Microsoft Dynamics AX
2. The partner or system implementer works with the customer and their IT staff to determine the hardware and software requirements for AIF. They analyze the existing environment and recommend any new hardware or software that must be installed.
3. The customer IT staff install and configure any required hardware and software to support AIF.
4. The partner or customer developer programs the document exchange. They may make customizations to the AIF documents or create new documents to meet the requirements of the business users. How AIF is configured depends in part on the network environment. Therefore, the developer may work with IT staff when implementing an integration.
5. IT staff monitor the document exchanges and troubleshoot any errors that are generated.

A high-level view of the process of integrating Microsoft Dynamics AX with other systems



What's new in Application Integration Framework (AIF)

Application Integration Framework (AIF) enables companies to integrate Microsoft Dynamics AX and communicate with external business processes and partners through the exchange of XML over various transport media. AIF enables both business-to-business and application-to-application integration scenarios. Enhancements to AIF in Microsoft Dynamics AX 2009 include the following:

- Create, read, update, and delete (CRUD) operations are now supported. The ability to update and delete data through AIF enables companies to fully integrate Microsoft Dynamics AX in their business processes.
- The programming model for AIF supports document services that encapsulate business logic and are the interface between Microsoft Dynamics AX and external systems. The document services can be customized by adding your own methods. Microsoft Dynamics AX now supports the ability to expose business logic implemented in X++ as Windows Communication Foundation (WCF) services. All services can be published though AIF asynchronous adapters or through WCF.
- AIF provides functionality for consuming external Web services from within X++. This enables the integration with any external data provided as a Web service.
- Performance improvements include the ability to scale up and handle more messages through parallel message processing and the addition of multiple AOSs.
- New document services for additional commonly-used documents.

Update and Delete

AIF now supports the ability to update and delete data in Microsoft Dynamics AX through document exchange. Two new actions enable the ability to update or delete a single record or multiple records. The Axd Wizard is now the AIF Document Service Wizard and enables developers to create custom documents that support updating and deleting data.

Programming Model Changes

Documents are now exposed as services making them more flexible and customizable. Service operations can now be consumed from external AIF clients as well as directly from X++ code. The AIF framework has been updated to enable developers to expose X++ business logic as WCF services. ASMX Web services have been replaced with WCF services enabling new functionality such as message encryption.

Consume Web Services

Microsoft Dynamics AX 2009 provides the ability to consume Web services from within X++. Creating a service reference to an external Web service generates a .NET service proxy that enables the service to be available through IntelliSense within X++. The external Web service can be consumed by Microsoft Dynamics AX using the .NET service proxy through CLR interop.

Performance

AIF now supports the ability to scale up message processing when using the AIF adapters by adding AOSs. You can now define whether messages are processed sequentially or in parallel. If parallelism is implemented, messages can be processed in any order by multiple AOSs. After implementing parallel message processing, specific messages can still be designated for sequential processing.

New Documents

AIF includes support for an enlarged set of frequently used documents. Documents new to Microsoft Dynamics AX 2009 include the following.

Document	Supported service operations (Create, Read, Update, Delete)
Address	CRUD
Absence Request	R
Bill of Materials (BOM)	CR
Business Sector	C
Cash Discount	R
Contact Person	CRUD
Credit Card	C
Customer	CRUD
Customer Group	R
Customer Payment Journal	C
Customer Quotation	C
Expense Report	CRU
Fixed Asset	CRU
Fixed Asset Condition	R

Document	Supported service operations (Create, Read, Update, Delete)
Fixed Asset Group	R
Fixed Asset Location	R
Fixed Asset Major Type	R
General Journal	CR
Item	CR
Item Dimension (Color)	R
Item Dimension (Configuration)	R
Item Dimension (Size)	R
Item Dimension Combination	R
Leads	C
Payment Terms	R
Product Groups	R
Product Picking List	C
Project Hour Journal	C
Return Order Acknowledgment	R
Return Order Document	CR
RFQ	R
RFQ Reply	C
Route Card	C
Sales Confirmation	R
Sales Quotation	R
Sales Forecast DMP	R
Service Agreement	R
Service Order	CR
Shipping Methods	R
Transfer Order	CR
Travel Card	C

Document	Supported service operations (Create, Read, Update, Delete)
Unit	R
Unit Conversion	R
Vendor	CR
Vendor Group	CR
Vendor Payment	R
Warehouse	R
Work Center	CR
Worker Attendance	CRUD

In addition to these new documents, the sales order document has been updated to support multi-site functionality as well as the ability to update and delete data.

Partners and customers can easily customize and extend the existing application programming interfaces (APIs) by using the Microsoft Dynamics AX software development kit (SDK). The SDK includes a service wizard that developers can use to easily create their own custom document services.

See Also

Configure and manage AIF

Planning for AIF integration

Planning is an integral part of any data integration effort. When integrating Microsoft Dynamics AX with other systems, one of the initial steps is the planning phase. In this phase, the implementation team must define high-level requirements and make integration design decisions. After these requirements are defined, the partner, IT staff, and development staff can then work together to define the best way to implement the exchange in AIF.

The integration design decisions that must be made fall into two primary categories: data and configuration.

- Data - At the core of data integration is the data itself. While planning your data integration, many decisions must be made about the data that is being exchanged and the associated business rules. This phase often involves the expertise and knowledge of business users because they understand the meaning of the data and define the requirements for integration. This category addresses the "who, what, and when" requirements of the integration.

- Configuration - Configuration requirements define the environment of the data exchange. Factors that affect these requirements include the following: the network configuration, the hardware and software configuration of the external system, and the level of trust between Microsoft Dynamics AX and the external system. This category addresses the "how" requirements of the integration.

Before writing any code or configuring document exchanges, the implementation team should consider the following questions.

Category	Question	Design impact
Data	What data elements are involved in the exchange and what screens do those elements come from? Are there any calculated values?	Helps determine which AIF documents support the business needs.
Data	Is the data being sent from Microsoft Dynamics AX to an external system or is the data received by Microsoft Dynamics AX from an external system?	Helps determine how a document exchange is configured.
Data	Does the external system request data from Microsoft Dynamics AX or is there an event in the application that triggers the sending of data to the external system?	This information helps determine how a document exchange is configured.
Data	Are records in Microsoft Dynamics AX being created, updated, or deleted?	Helps determine whether the AIF documents will need any customizations.
Data	What are the business rules associated with the data? For example, if data is created or updated, which data elements are mandatory? If data is deleted, what are the conditions under which a record can be deleted?	Helps determine whether any customizations must be made to existing AIF documents.
Data	Do the documents that ship with Microsoft Dynamics AX contain the data that must be exchanged?	Helps determine whether any customizations have to be made to existing AIF documents or if new documents must be created.
Data	Do the relevant documents support the actions that must be performed on the data (read, create, update, or delete)?	If the existing documents do not support some of the data integration requirements, developers may need to make customizations.

Category	Question	Design impact
Data	Does the data need to be transformed by Microsoft Dynamics AX? This could be transformations that need to be performed before data is sent or when data is received. What is the extent of the data transformations?	Enables the team to determine whether AIF value mapping or XSLT transformations should be used.
Configuration	Does the local Microsoft Dynamics AX system have any restrictions on how data is exchanged?	Determines how a document exchange is configured. For example, if there is a requirement to use Message Queuing as a transport mechanism, then the MSMQ adapter would be used and the exchange would be asynchronous.
Configuration	Does the external system have any restrictions on how data is exchanged?	Determines how a document exchange is configured.
Configuration	Are there any constraints on the data? For example, is the document exchange limited to a particular vendor or customer?	Determines how a document exchange is configured.
Configuration	Is the external system an in-house system or external trading partner?	Impacts how users and security are configured.

Note:

This planning information is a guideline for what you may need to consider when planning your data integration. For more information about your specific implementation, contact your partner.

See Also

Exchanging documents electronically using AIF

Use AIF to integrate with external systems

Use AIF to integrate with external systems

Application Integration Framework (AIF) provides an extensible framework that supports multiple asynchronous transports, as well as synchronous transport using Web services, to exchange documents in XML format with external systems.

An exchange starts with a service based on a document, that is, a document class defined using Microsoft Dynamics AX business logic. The document is serialized into XML and header information is added to create a message, which may then be transferred into or out of the Microsoft Dynamics AX system.

Your Microsoft Dynamics AX system is called the local endpoint within AIF. The other entity taking part in an exchange is called the endpoint. Endpoints and all the other elements of an exchange are set up and configured using forms found when you click **Basic** > **Setup** > **Application Integration Framework**.

There are two methods for exchanging data in AIF:

- Web services - A data exchange in which a Microsoft Dynamics AX service is consumed an external system.
- Adapters - A data exchange in which Microsoft Dynamics AX adapters are used to communicate with the external system. Microsoft Dynamics AX adapters support the following transport mechanisms:
    - File system
    - Message Queuing (MSMQ)
    - BizTalk Server

Web services-based exchanges

Using Web services for data exchange requires the installation and configuration of Web services for application integration and Microsoft Internet Information Services (IIS) 7.0. Exchanges configured to use Web services are processed synchronously and do not use the Microsoft Dynamics AX queues. AIF allows multiple connections; that is, your Microsoft Dynamics AX system can support the processing of document exchanges with multiple partners simultaneously.

To help ensure the highest level of security, deploy Web services on your intranet only. Deployment of Web services outside your intranet requires additional middleware known as a trusted intermediary to ensure security. For more information about AIF security, see Security considerations for AIF and Security considerations for AIF Web services.

For more information about data exchanges using Web services, see Web services-based exchanges in AIF and Configure document exchanges with Web services in AIF.

Adapter-based exchanges

An adapter-based exchange uses an adapter to convert the document into the proper format for exchange by means of a particular transport mechanism, such as Message Queuing (MSMQ). Adapter-based exchanges are asynchronous because they involve moving the document into a queue where it waits for processing by a Microsoft Dynamics AX batch job. Adapter-based exchanges require configuration of an adapter and a channel for use by AIF.

For asynchronous, adapter-based exchanges, you configure and control the Microsoft Dynamics AX batch jobs that process documents in the AIF queues.

Microsoft Dynamics AX includes the functionality to enable connections with the following asynchronous adapters:

- File system
- Message Queuing (MSMQ)
- BizTalk Server

Note:

Although adapter-based exchanges are asynchronous, if you use the BizTalk adapter, it is possible to configure the data exchange to be synchronous or asynchronous.

For more information about data exchanges using adapters, see Adapter-based exchanges in AIF and Configure document exchanges with adapters in AIF.

Send and receive documents and data

Regardless of which transport method you use, AIF can be used to either send data into Microsoft Dynamics AX (inbound) or retrieve it (outbound). An example of an inbound exchange would be an external system sending a sales order to be saved to the Microsoft Dynamics AX database. An example of an outbound exchange would be an external system sending a request for a purchase order and receiving the purchase order back. The inbound and outbound exchanges can be categorized as follows:

- **Send data** - Microsoft Dynamics AX sends documents to an external system.
- **Send data in response to requests** - Microsoft Dynamics AX receives requests for documents from another authorized system, retrieves the requested information (a document or a list of documents) from the Microsoft Dynamics AX database, and returns it to the requesting system, with appropriate filtering and security. The request message would contain the entity keys or a query that specifies the data that the external system is requesting.
- **Receive and create data** - Microsoft Dynamics AX receives documents from another authorized system and creates new records in the Microsoft Dynamics AX database.

By using outbound exchanges in AIF, you can send documents and data to your trading partners. You receive documents and data from endpoints in an inbound exchange.

Send documents and data

Sending a document can be initiated by clicking a button on a form, such as the **Send electronically** button on the **Chart of accounts** form. For more information, see topics "How to: Send and receive electronic documents automatically" and "How to: Send documents manually" in the Application and Business Processes Help.

Receive documents and data

When documents are received in an inbound transfer, data is added, updated, deleted or changed in the Microsoft Dynamics AX database. For this reason, you should carefully consider how to ensure the security of your Microsoft Dynamics AX system when configuring the Microsoft Dynamics AX users associated with an endpoint. Be sure that endpoint users are trusted by your business organization.

See Also

Adapter-based exchanges in AIF

Configure document exchanges with adapters in AIF

Web services-based exchanges in AIF

Configure document exchanges with Web services in AIF

Security considerations for AIF

Security of the Application Integration Framework (AIF) is critical when you exchange data with other systems. Determine who can access AIF through Microsoft Dynamics AX security. Users can access an AIF service with appropriate permissions to the service security key.

The AIF security design assumes that all inbound messages come from trusted sources. This means that AIF should not be deployed directly over the Internet or extranets. If AIF messages must be received from endpoints over the Internet, use middleware and trusted intermediaries (Microsoft Dynamics AX users or user groups authorized to act on behalf of an AIF endpoint).

AIF ships with a default endpoint that can be used for data exchange without any additional configuration as long as the submitting user is a valid Microsoft Dynamics AX user and has access to the service security key. If additional authorization, security verification, or constraints are required, then administrator should configure specific endpoints.

Configure users on endpoints

When configuring users on an endpoint, keep in mind that these Microsoft Dynamics AX users may represent outside interests and must have permissions set appropriately. For more information about configuring Microsoft Dynamics AX users, see the following topics in the System and Application Setup Help:

- Managing access to Microsoft Dynamics AX

- Manage security permissions for user groups and domain combinations
- Manage user groups
- Manage users

You must also set the appropriate security keys and record-level security for any users that are granted access to Microsoft Dynamics AX through AIF, to help prevent unauthorized data access. For more information, see "Manage record level security" in the Application and Business Processes Help.

Certain actions cause data to be written directly to the Microsoft Dynamics AX database without manual end-user verification (for example, creating exchange rates). When configuring endpoints and creating new actions, be careful to restrict access to trusted and reliable partners and applications. For more information about the behavior of individual documents, see "Standard Axd Documents" in the Microsoft Dynamics AX SDK Help.

Trusted intermediaries

Trusted intermediaries are middleware applications that reside between external endpoints and AIF. That is, they are Microsoft Dynamics AX users (or user groups) that are authorized to submit inbound requests on behalf of the endpoint. A trusted intermediary prevents an unauthorized user from accessing AIF and is typically used in a business-to-business data exchange scenario.

BizTalk Server installations and Electronic Data Interchange (EDI) services are examples of systems that can act as trusted intermediaries and submit inbound requests. These systems can be designated as trusted intermediaries if they can be trusted to reliably and accurately identify who has submitted the requests they forward to AIF. Part of the trusted intermediaries' responsibility is to ensure that messages from untrusted third parties are never allowed access to your system.

If AIF is used strictly for in-house integrations, and the users who submit documents to AIF exist as Microsoft Dynamics AX users, then trusted intermediaries are not required. However, if you need to receive messages from systems or trading partners outside your Microsoft Dynamics AX system (that is, from Microsoft Dynamics AX users with the **External** check box selected on the **User** form), trusted intermediaries must always be used. To use them, you should be aware of the following requirements.

A trusted intermediary must:

- Be created as a valid Microsoft Dynamics AX internal user (the **External** check box is not selected on the **User** form).
- Be known to you as representing a valid partner or a trusted system.
- Be configured within AIF by selecting the **Use trusted intermediary** check box on the **Users** tab in the **Endpoints** form and adding the Microsoft Dynamics AX internal user that represents the trusted intermediary. For more information about configuring users in Microsoft Dynamics AX, see "Setting up and maintaining security" in the System and Application Setup Help.

- Be set up and configured to submit messages on behalf of Microsoft Dynamics AX external endpoints.
- Ensure that the information about the `SourceEndpointUser` (specified for inbound transfers in clear text in the message header) is accurate and cannot be changed by the external endpoint.
- Verify that the request message or inbound document was submitted by the authorized user, and not by an attacker using "spoofing" to impersonate the authorized user. The trusted intermediary system must authenticate the submitting user. If the user is not authenticated then the request must be rejected by the trusted intermediary.

Security best practices

Follow these additional security-related recommendations when configuring AIF. For more information about security and Web services in AIF, see Security considerations for AIF Web services.

- Ensure that transport channels are secure and can be accessed only by authenticated and authorized users. When transmitting messages to the file system or Message Queuing (MSMQ), be sure that these channels (the file shares and queues themselves) are secure and can be accessed only by authorized users. This can be done using security software such as SecureFTP that encrypts data and ensures that only authorized users can access a file location. When sending and receiving data through BizTalk Server, only authorized users should be able to create orchestrations or submit documents. Authentication and encryption are particularly important for business-to-business scenarios in which data is being transmitted over the public Internet.
- Ensure that data sent to and from AIF channels is secure and encrypted. All data transmissions should be secured so that no one can read or modify data during transmission. This can be done using security software such as SecureFTP.
- The Microsoft Dynamics AX system administrator should limit access to modifying the AIF configuration only to the Microsoft Dynamics AX Administrators group. For more information, see "Manage user groups" in the System and Application Setup Help.
- The Microsoft Dynamics AX system administrator should restrict access to AIF by assigning users permissions to only the security keys that they need. In general, the following types of users will need access to AIF:
  - Action generation - Developers
  - General AIF configuration - IT staff
  - Queue management - IT staff and potentially Microsoft Dynamics AX users
- Be aware that all actions involving inbound documents are executed within Microsoft Dynamics AX under the context of a valid Microsoft Dynamics AX user. If a `DestinationEndpoint` is provided in the message, then the `SourceEndpointUser` must be a valid Microsoft Dynamics AX user. If no `DestinationEndpoint` is provided in the message, then AIF will use the default endpoint.

- Associate AIF endpoints only with trusted Microsoft Dynamics AX users. Currently AIF does not authenticate the actual endpoint identification. Instead, it authenticates the user associated with the endpoint. Only users specifically configured on the endpoint are allowed to perform actions associated with the endpoint.
- Be sure to secure the file system location where you export messages from the **Queue manager** form. These messages could contain confidential information
- To restrict an endpoint to sending or receiving data only for specific customers, vendors, or warehouses and avoid spoofing attacks, use endpoint constraints.

  Add external components only from a trusted and reliable source, for example, a Microsoft Partner or independent software vendor (ISV). External components include pipeline components (X++ classes called by AIF pipeline processing), document classes, and adapter classes.
- Before adding an XSLT as part of pipeline processing, ensure that the XSLT is secure and capable of handling documents with incorrect or malicious data. Thoroughly test any transformations to ensure that they do not contain code that will run and create an error on the system that can be exploited.
- By default, scripting is disabled on the XSLT transform component. If scripting is turned on, this can open up your system to scripting attacks. To check whether scripting is enabled:
  a. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
  b. Click **Action policies**.
  c. In the **Endpoint Action Policies** form, select an action, and then click **Inbound Pipeline** or **Outbound Pipeline**.
  d. In the **Pipeline components** form, select a transformation or value substitution record, and then click **Configure**.
  e. In the **Pipeline XSLT transform** form, the **Scripting enabled** field should be cleared.

See Also

Configure document exchanges with adapters in AIF

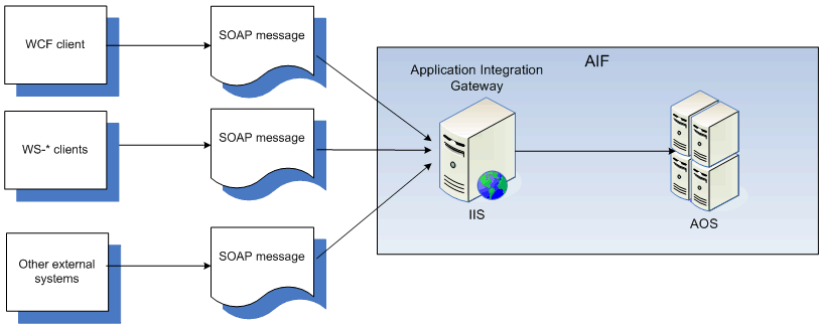
Configure document exchanges with Web services in AIF

Security considerations for AIF Web services

Application Integration Framework (AIF) supports Windows Communication Foundation (WCF) Web services. In AIF, each document is represented by a service that is exposed as a WCF service and hosted in Internet Information Services (IIS). AIF uses standard WCF processing to receive and process SOAP requests.

Web services security architecture

Security in AIF is enforced through a combination of WCF, IIS, Active Directory, and user security within Microsoft Dynamics AX. The authentication process that occurs when a client calls a Microsoft Dynamics AX WCF service is shown in the following figure.



AIF Web services security architecture

Note:

This figure contains an example of application-to-application integration in which all the systems are trusted users on an intranet. If the clients were outside the intranet, there would be a trusted intermediary (such as BizTalk Server or electronic data interchange (EDI) services) in between the client and AIF. For more information about trusted intermediaries, see Security considerations for AIF.

In the preceding illustration, the Web services authentication process is as follows:

1. The client calls a service method, such as the `Customer.read` method, and passes the entity key of the requested customer in a SOAP message.
2. The request is received by IIS where the AIF services are hosted. IIS retrieves the user credentials, depending on the authentication mechanism specified in the service configuration. IIS then tries to map the security credentials onto a valid domain user. By default, Microsoft Dynamics AX configures WCF to use the `basicHttpBinding` binding with message security so the user credentials are contained in the message SOAP header. IIS authenticates the user as a valid user in Active Directory.

3. The request is then passed to AIF which performs further authentication by verifying that the user:
   a. Is a valid Microsoft Dynamics AX user
   b. Has access to the service through the appropriate security key
4. After AIF determines that the user has access to the service, the message is processed. At run time, standard AIF security ensures that the user has access to the data exposed by the service.

Security best practices

Follow these additional security-related recommendations when configuring AIF Web services:

- It is unsafe to deploy AIF using Web services outside the intranet without installing additional middleware such as BizTalk Server to ensure proper security. As installed with Microsoft Dynamics AX, AIF Web services are intended for intranet deployment only.
- Deploy Web services in the intranet only and configure the Web server so that it is not facing the Internet. By default, AIF services implement the `basicHttpBinding` binding configured to use WCF message-level security. Administrators should follow the standard WCF configuration in IIS. For more information about security in WCF, see Securing Services.
- After installation, be sure that you have restricted access to the Web service share as described in "Install AIF Web Services" in the Microsoft Dynamics AX Installation Guide.
- Secure outbound Web services so that only authorized users can send data from Microsoft Dynamics AX by setting security key permissions on the AIF services. This is done by assigning users permissions to specific services. To give users access to services, follow the standard method of assigning user permissions to security keys. For more information, see topic "Manage security permissions for user groups and domain combinations" in the System and Application Setup Help.
- In Microsoft Dynamics AX 2009, the WCF services for the AIF documents are secured by security keys. The Microsoft Dynamics AX administrator must explicitly grant users and groups permissions to the appropriate security key in order for users to access the service.
- All data exchanged with external Web services (when consuming an external Web service from X++, for example), must be over secure channels to prevent tampering, spoofing, and so on. Confidential and business critical information should only be exchanged with external Web services through communication channels that provide for secure authentication, message confidentiality, and integrity.
- You should never consume unknown or untrusted external Web services. When consuming a Web service, you should always be certain that the consumed service is the correct service (by using a secure identification/authentication mechanism).
- For more security best practices, see Security considerations for AIF.

Documents included with Microsoft Dynamics AX

Microsoft Dynamics AX includes over 70 standard documents and each document supports a common business process. Application Integration Framework (AIF) lets you customize these standard documents or create your own custom documents to suit your individual business processes.

For more information about the documents that are included with Microsoft Dynamics AX, see "Standard Axd Documents" in the Microsoft Dynamics AX SDK Help.

Troubleshoot AIF

This topic describes how to troubleshoot common issues with document exchange using the Application Integration Framework (AIF).

For more information about troubleshooting AIF Web services installation, see "Troubleshooting AIF Web services installation" in the Microsoft Dynamics AX Installation Guide.

Set up an endpoint to use inbound Web services

You may see the message in the Web server event log:

"The requested operation cannot be performed because the required security key doesn't exist."

This occurs if the endpoint user does not have access to Business Connector.

When you configure an endpoint for inbound Web services, you must:

1. Set up a Microsoft Dynamics AX user (or user group) as an endpoint user or trusted intermediary for that endpoint.
2. Give the user group access to the Business Connector.

Give the endpoint user access to Business Connector

1. Click **Administration** > **Setup** > **User groups** and select the user group for the endpoint, or the user group that contains the user for the endpoint.
2. Click **Permissions**.
3. On the **Permissions** tab, in the table under the **Viewing** field:
   a. Select **Business Connector**.
   b. Select **Full control**.
   c. Click **Cascade**.

After installing Enterprise Portal, AIF Web services do not work

For the AIF Web services to coexist with Windows SharePoint Services (WSS) and Enterprise Portal on the same computer, the virtual directory that AIF is using for Web services must be excluded from the SharePoint managed path.

To exclude the AIF virtual directory from the SharePoint managed path list:

1. Launch the SharePoint Central Administration page (**Start** > **All Programs** > **Administrative Tools** > **SharePoint Central Administration** or, from the browser on a remote computer, type the URL for the pages on the administration port. For example: *http://servername:port*).
2. Click **Configure virtual server settings** and click the name of the site that you are managing.
3. Under **Virtual Server Management**, click **Define managed paths**.
4. Under **Add a New Path**, enter the AIF virtual directory path, and then select **Excluded path**, and click **OK**.

Re-register ASP.NET when setting up inbound Web services

If you cannot see the ASP.NET tab when viewing Properties for the virtual directory in the IIS Services Manager (see topics "Install AIF Web services" and " Troubleshooting AIF Web services installation" in the Microsoft Dynamics AX Installation Guide), or if you see a "Page not found" error after clicking **Browse** when validating Web services, you must re-register ASP.NET in IIS using the following steps:

1. Click **Start** > **Run**.
2. Type `cmd`.
3. In the Command Prompt window, type: `cd` *your-system-directory*`\Microsoft.NET\Framework\v2.0.50727` and press ENTER.
4. Type `aspnet_regiis.exe -u` and press ENTER.
5. Type `aspnet_regiis.exe -i -enable` and press ENTER.
6. Type `iisreset` and press ENTER.
7. Close the Command Prompt window.
8. After uninstalling and reinstalling ASP.NET and resetting IIS, the ASP.NET tab is available, and you can select **ASP version 2.0**. This allows you to browse the Web services after you enable and generate them on the **AIF Services** form. For more information, see Configure services

Calling Web services in Active Directory results in an access error

If you are calling Web services and receive an error even though the calling user has access, this may be due to an issue between Windows authentication and Kerberos security in an Active Directory environment.

Symptoms of this problem occur when you call a Web service URL such as `http://`*<URL>*`/SalesOrderService.asmx?WSDL` and you receive an error even though the user calling the Web service has security access to the Web service. If you look in the Event Viewer on the calling machine, you may see a Kerberos error such as:

"The kerberos client received a KRB_AP_ERR_MODIFIED error from the server host/<Computername>.<DNS-ComputerDomain>. The target name used was HTTP/<computername>. This indicates that the password used to encrypt the kerberos service ticket is different than that on the target server. Commonly, this is due to identically named machine accounts in the target realm and the client realm. Please contact your system administrator."

This error is caused by an issue with name resolution in the network environment. To call the Web service, you must replace the name of the server where the Web services are hosted with an IP address, for example, `http://`*<IP Address>*`/SalesOrderService.asmx?WSDL`.

Issues when using Message Queuing (MSMQ) to exchange documents

- If the error message "The transaction context is invalid" appears for an inbound message, verify that the Message Queuing queue that receives the message is located on the same computer as the Application Object Server (AOS) running the AIF batch jobs. For more information on these batch jobs, see Start and stop the asynchronous AIF services.
- If the error message "The specified format name does not support the requested operation. For example, a direct queue format name cannot be deleted." appears when a message is sent, verify that the queue you are sending to is a public queue and its address (on the **Channels** form) is given in short name format (*computer-name**queue-name*).
- If it seems that outbound messages are sent (that is, they no longer appear in the list on the **Overview** tab in the **Queue manager** form, and no entries are created in the Exception Log for an error condition), but the messages are not received by the target queue, ensure that the target queue's access control list (ACL) is set properly by following these steps.
    a. On the **Security** tab of the Properties window for the queue, **Allow** should be selected for **Send Message**, **Get Permissions**, and **Get Properties** for the Anonymous Logon user.
    b. Verify that the target queue's **Authenticated** property is cleared on the **General** tab in the **Message Queuing** folder (click **Start** > **Administrative Tools** > **Computer Management**).
- For inbound messages, if there is a message in the Exception Log that states "The user is not authorized to perform this action", check the **Queue manager** form for any inbound messages in an error state by clicking **Basic** > **Periodic** > **Application Integration Framework** > **Queue manager**. If the **Submitting user** field on the **Details** tab is blank, verify that either the inbound queue's **Authenticated** property is selected or that all incoming messages are signed and authenticated.

Error received processing a message with Web services or the BizTalk adapter

While processing a message using Web services or the BizTalk adapter, you may see the error "The requested operation cannot be performed because the required security key doesn't exist."

This error may occur if the user has not been granted execute permissions on the Business Connector security key. For more information about setting permissions, see "Manage security permissions for user groups and domain combinations" in the System and Application Setup Help.

Adapter-based exchanges in AIF

A transport adapter is a software component that enables data to be exchanged asynchronously in Application Integration Framework (AIF). An asynchronous exchange is one in which documents are placed in a queue to wait for processing by a transport mechanism.

The transport adapters that are included with Microsoft Dynamics AX include:

- Microsoft Message Queuing (MSMQ) - Messages are sent and received through Message Queuing queues.
- File system - Messages are sent and received through file system directories.
- BizTalk Server - Messages are sent and received through BizTalk orchestrations.

For example, you may have a scenario where the chart of accounts is sent from Microsoft Dynamics AX to two external systems. The first system receives messages by checking a specific file system directory for files while the second system uses Microsoft Message Queuing (MSMQ) to receive messages. In this case, whether the chart of accounts is sent to the first system or the second system, both messages are first placed in the queue in Microsoft Dynamics AX. For more information about how to configure a data exchange using adapters, see Configure document exchanges with adapters in AIF.

To start and stop processing in the queues, use the Microsoft Dynamics AX batch functionality. For more information, see Start and stop the asynchronous AIF services. In the previous scenario, when the batch jobs start, they will pick up the messages in the queue. The batch jobs will send the first message to the appropriate directory and send the second message to the appropriate Message Queuing queue. How the messages are processed is defined by how the endpoint is configured and how the channel for that endpoint is configured.

See Also

Configure document exchanges with adapters in AIF

AIF security concepts for user credentials

This topic describes the types of users that you need to set up for data exchange in the AIF.

Before you begin

You must understand the following concepts before you configure the AIF:

- The difference between a Microsoft Dynamics AX user (internal user) and an external user. For more information, see Create new users or the online help for the **Users** form (**Administration** > **Users**).
- The concept of trusted intermediaries. For more information, see Security considerations for AIF.
- The security features that BizTalk Server uses to authenticate inbound messages. For more information, see Inbound Message Authentication.
- The security features that BizTalk Server uses to authenticate messages between processes. For more information, see Authentication of Messages Between Processes.
- The Application Integration Framework (AIF) messages. For more information, see AIF Messages.

Use the information in the following section to understand and plan the AIF configuration.

- AIF users
- AIF security concepts for BizTalk adapter
- Considerations for the endpoint user configuration

See Also

Security considerations for AIF

Security considerations for AIF Web services

AIF users

This topic explains the different types of users in the Application Integration Framework (AIF).

AIF Users

There are several types of users in AIF. The applicable type of user depends on the transport and the mode of communication (synchronous or asynchronous).

- Source endpoint user
- Submitting user
- Trusted intermediary

- Proxy user
- Gateway user (only used with the AIF BizTalk adapter)

Source endpoint user

The source endpoint user is the user who sends the message. AIF processes the request in the context of the source endpoint user in Microsoft Dynamics AX. The inbound message contains the source endpoint user as shown in the following code sample:

```
<Header>
<MessageId>{FFF3E75E-A75C-4228-ABF0-8E3EA2483EB4}</MessageId>
<SourceEndpointUser>MYDOMAIN\MYUSER</SourceEndpointUser>
<SourceEndpoint>CustomerCompany</SourceEndpoint>
<DestinationEndpoint>MyCompanyLEP</DestinationEndpoint>
```

Use the **Endpoints** form to configure the endpoint to allow the source endpoint user to submit the message, as explained in Considerations for the endpoint user configuration. The source endpoint user must be an internal or external Microsoft Dynamics AX user.

Submitting user

The submitting user submits the message into Microsoft Dynamics AX and is determined by the transport mechanism used, as detailed in the following table. The submitting user must be an internal user. You may have to configure the **Endpoints** form to allow the submitting user as a trusted intermediary, as explained in Considerations for the endpoint user configuration.

The following table explains the process used by different transports to determine the submitting user.

Data exchange method	Submitting user
BizTalk adapter	The submitting user is determined as explained in AIF security concepts for BizTalk adapter.
File system adapter	The submitting user is the owner of the message request file as returned by the Windows function GetFileSecurity (OWNER_SECURITY_INFORMATION).
MSMQ adapter	The submitting user is the sender of the MSMQ message as set on the SenderId property of the MSMQ message.
Web services	The submitting user is the Windows identity of the caller.

Trusted intermediary

The trusted intermediary is a logical intermediary between the document request originating party (external endpoint) and AIF. The trusted intermediary is authorized to submit inbound requests on behalf of the external endpoint. The trusted intermediary must be an internal account. For more information, see Considerations for the endpoint user configuration.

Proxy user

The use of a proxy enables the .NET Business Connector to connect on behalf of Microsoft Dynamics AX users when authenticating with an Application Object Server (AOS) instance. For more information, see Set the Business Connector proxy user.

Gateway user

The gateway user is used by the BizTalk adapter for asynchronous messaging when messages are sent and received from the AIF gateway queue. Asynchronous messaging occurs when the send and the receive ports are one-way ports. The gateway user must be an internal user with permission to access the gateway queue. The gateway user is usually the Admin user. In the BizTalk orchestration, use the **Microsoft Dynamics AX 2009 Transport Properties** window to configure the gateway user.

See Also

Configure the file system for AIF

Configure Message Queuing for AIF

Configure BizTalk for AIF

Configure document exchanges with Web services in AIF

AIF security concepts for BizTalk adapter

This topic describes how BizTalk adapter and BizTalk server determine different users for a data exchange. For a comprehensive and step-by-step guidance to exchange data with AIF using BizTalk Server, see Application Integration Framework (AIF) BizTalk adapter configuration for data exchange.

Source endpoint user

The BizTalk adapter uses the following process to determine the source endpoint user:

1. If the message has an envelope, get the source endpoint user from envelope header.
2. If the message header is missing, get the source endpoint user from the BizTalk message context (DynamicsAx5.SourceEndpointUser).
3. If the user is configured in neither the header nor the BizTalk message context, use the submitting user as the source endpoint user.

Submitting user

The submitting user is always set by the BizTalk Server. If you view the message details using the BizTalk Server administration console, the submitting user is the value of the Originator Security ID message property. This value is not configurable. For information about the security features that BizTalk Server uses to authenticate the inbound messages, see Inbound Message Authentication. For information about the security features BizTalk Server uses to authenticate messages between processes, see Authentication of Messages Between Processes.

Gateway user

The gateway user is used by the BizTalk adapter for asynchronous messaging when the messages are sent and received from the AIF gateway queue. Asynchronous messaging occurs when the send and the receive ports are one way ports. The gateway user must be an internal user with the permission so the gateway queue. The gateway user is usually the Admin user. In the BizTalk orchestration, use the **Microsoft Dynamics AX 2009 Transport Properties** window to configure the gateway user.

Proxy user

The proxy user setting has no dependency on the endpoint user, submitting user or gateway user. The Business Connector proxy account is a Microsoft Windows domain account that enables the Business Connector to act on behalf of Microsoft Dynamics AX users when the users authenticate with the Application Object Server (AOS) via a BizTalk application. The proxy user account must be same as the user account in the Business Connector Proxy section of the **System service accounts** form. The configuration of the **Microsoft Dynamics AX 2009 Transport Properties** window in the BizTalk Server administration console determines how the BizTalk adapter selects the proxy user.

1. If the **Authentication Type** field is **Host User**, the service account for the BizTalk Server is used as the proxy user.
2. If the **Authentication Type** field is **Proxy User** , the values of the **Proxy User** and **Proxy Password** fields from the **Microsoft Dynamics AX 2009 Transport Properties** window are used as the proxy user.

Note:

If the authentication type is single sign on (SSO), the proxy user is not used as the Microsoft Dynamics AX authentication is done using the SSO user credentials. For more information on SSO, refer to the BizTalk Server documentation.

Considerations for the endpoint user configuration

This section describes how to determine the endpoint user and the trusted intermediary for the **Endpoints** form.

Considerations for the endpoint user configuration

The **Endpoints** form is used to configure endpoints for data transfer. You must specify users for all named endpoints. You must also specify a trusted intermediary when the source endpoint user is an external user or when the source endpoint user is different from the submitting user as explained in a later section of this topic. Use this topic to understand how to configure users for named endpoints. This topic is not applicable to the default endpoint as the default endpoint allows all internal users to submit messages to AIF.

Determine the endpoint users

Use the following table to determine the source endpoint user and the trusted intermediary:

Condition	Configuration
The source endpoint user is the same as the submitting user.	Configure the source endpoint user as the endpoint user on the **Endpoints** form.
The source endpoint user differs from the submitting user.	Configure the source endpoint user as the endpoint user on the **Endpoints** form. Configure the submitting user as the trusted intermediary on the **Endpoints** form.

Note:

The submitting user must be an internal user. The source endpoint user can be an internal or external user.

Use the following table to determine the endpoint user and the submitting user:

User ID	User type	Can be an endpoint user	Can be a submitting user
InternalUserA	Internal user	Yes	Yes
InternalUserB	Internal user	Yes	Yes
ExternalUserC	External user	Yes	No

The following table explains configuration of the endpoint user and trusted intermediaries in the **Endpoints** form:

Message source endpoint user	Request submitted by	Endpoint user	Trusted intermediary
InternalUserA	InternaluserA	InternalUserA	Not required
InternalUserA	InternalUserB	InternalUserA	InternalUserB
ExternalUserC	InternalUserB	ExternalUserC	InternalUserB

AIF performance

When data is transferred in Application Integration Framework (AIF) using one of the adapters as the transport, messages move in and out of the system through the Microsoft Dynamics AX queue. This queue is known as the gateway queue.

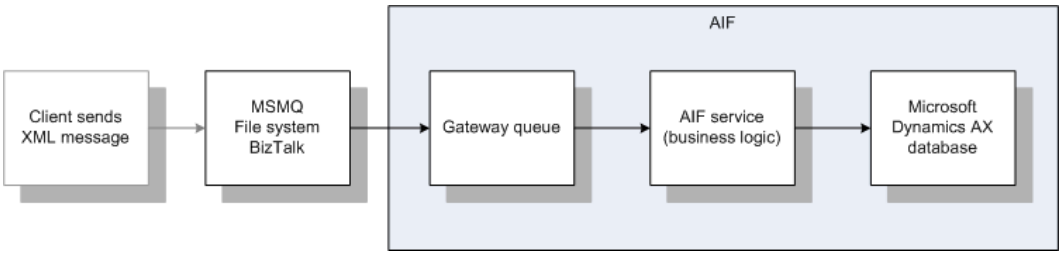
When using asynchronous adapters, some implementations may experience a performance impact when there is a high volume of messages sent to or from a particular endpoint. This performance impact is only experienced with asynchronous processing.

Inbound message processing

The processing flow for an inbound message is shown in the following diagram. The external system sends a message to Message Queuing (MSMQ), the file system, or BizTalk. The AIF gateway receive service polls for messages and then brings them into the gateway queue. The AIF inbound processing service takes them from the gateway queue and the business logic in the AIF document service processes the message and then saves it to the database.

Moving messages out of the gateway queue and invoking business logic to process the messages is sequential by default. That is, for each AIF destination endpoint, only one message can be processed at any given time.

The transition of the message from the gateway queue and calling business logic can be a potential performance bottleneck. Even with multiple AOSs, all messages bound for the same endpoint are processed from the gateway queue sequentially, even if this processing order is not necessary from a business process perspective.



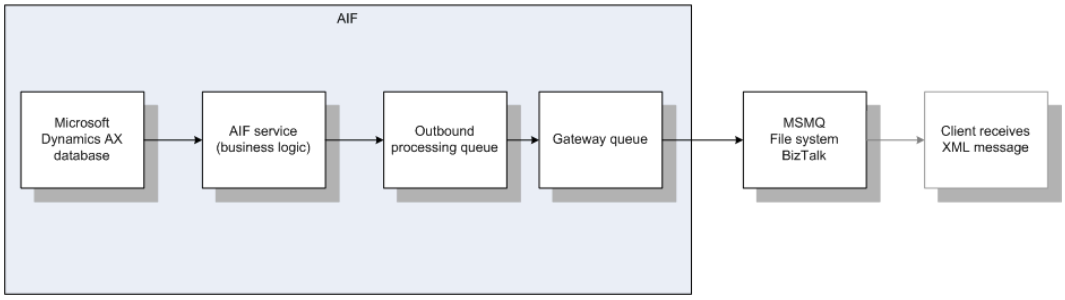
The conceptual message path for an inbound message

Outbound message processing

The processing flow for an outbound message is shown in the following diagram. The AIF outbound processing service sends a message with data from the database to the outbound processing queue. The AIF gateway send service then moves the messages from the outbound processing queue and sends them to Message Queuing (MSMQ), the file system, or BizTalk. The external system then receives the message from the transport.

When outbound messages are moved from the outbound processing queue to the gateway queue, all messages that are bound for the same destination endpoint are processed sequentially, by default. That is, for each AIF destination endpoint, only one message can be processed at any given time.

The transition of the message from the outbound processing queue to the gateway queue can be a potential performance bottleneck. Even with multiple AOSs, all messages bound for the same destination endpoint are processed sequentially, even if this processing order is not necessary from a business process perspective.



The conceptual message path for an outbound message

Improve performance

To improve performance for data exchange using asynchronous adapters, AIF supports parallelism. Parallelism specifies that inbound messages are processed by one or more AOSs without regard to the order in which they are received or produced. This enables you to scale out message processing by adding multiple AOSs. Note that you can enable parallelism only for inbound channels.

Enable parallelism

Parallel processing in AIF is implemented on a per channel basis. Follow these steps to implement parallel processing.

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Channels**.
2. Select a channel and select the **Parallel processing** check box.

Conversational parallelism

Conversational parallelism—also known as ordered parallelism—means that you can specify certain messages to be processed sequentially in a channel even when parallelism is enabled for that channel. This is done by including a special XML element called `<ConversationId>` in the messages that require sequential processing. All messages with the same `ConversationId` will then be processed sequentially.

For more information about this message header element, see "Document Schema Overview" and "Message Header" in the Microsoft Dynamics AX SDK Help.

Note:

If you do not select the **Parallel processing** field in the **Channels** form, all inbound messages for a particular endpoint will be processed sequentially and the `<ConversationId>` element in a message is ignored.

Number of messages processed in parallel

When you enable parallelism, the number of messages processed in parallel by the AOS is set to 1000 by default. Inbound messages that contain a `ConversationId` element value are processed sequentially and not in parallel. The `MaximumInboundParallelMessages` macro defines the number of inbound messages that are processed in parallel. The `MaximumOutboundParallelMessages` macro defines the number of outbound messages that are processed in parallel.

You can change this number in the AIF macro in the AOT. You will need a developer license to access the AOT. To change the number of message processed in parallel, follow these steps.

1. Open the AOT and expand the **Macros** node.

2. Double-click the **Aif** macro to open it.
3. Find the following macro values and modify the number of messages according to your performance requirements.

```
#define.MaximumInboundParallelMessages      (1000)

#define.MaximumOutboundParallelMessages   (1000)
```

See Also

Create a channel

Configure document exchanges with adapters in AIF

In Application Integration Framework (AIF), the following steps are used to set up and manage a document exchange using adapters:

- Configure the prerequisites.
- Configure the exchange.
- Configure other AIF settings, if required.
- Maintain integration with external software systems.

Prerequisites

When you configure a document exchange that uses an adapter, you must first perform these tasks to configure the transport mechanism that you are using:

- For exchanges that use the file system, see Configure the file system for AIF.
- For exchanges that use Message Queuing, see Configure Message Queuing for AIF.
- For exchanges that use BizTalk, see Configure BizTalk for AIF.

Configure the exchange

To quickly configure a document exchange, follow the steps in the minimal configuration. If you want to customize or modify the document exchange, follow the steps in the extended configuration.

Minimal configuration

To set up a document exchange with the minimal configuration, perform the following steps:

1. Configure an adapter. In this step, specify which transport adapter the data transfer uses: file system, Message Queuing, or BizTalk. See Configure an adapter. This step is **required**.
2. Configure a channel. Specify the transport details for the exchange. See Creating and configuring channels and Create a channel. This step is **required**.

3. Enable the service. After you enable the service, the service operations (actions) are available for use by the endpoint. See Configure services. This step is **required**.
4. For inbound transfers, create the XML message and place it in the appropriate location as defined in the channel. This step is **required**.
5. Configure the Microsoft Dynamics AX batch jobs to send and receive messages. For inbound transfers, these jobs move messages from external locations (the file system, the Message Queuing queue, or BizTalk Server) into the internal queues and then save the data into the Microsoft Dynamics AX database. For outbound transfers, these jobs move messages into the internal queues and then to the external locations (the file system, the Message Queuing queue, or BizTalk Server). See Start and stop the asynchronous AIF services. This step is **required**.

Extended configuration

The extended configuration provides more flexibility and options for configuring the AIF components that make up the document exchange. This configuration allows you to create your own endpoints, configure actions for those endpoints, and specify endpoint action data policies. To set up a document exchange with the extended configuration, perform the following steps:

1. Create a local endpoint. See Create and configure local endpoints. This step is **required** if you are going to create an endpoint in a subsequent step.
2. Configure an adapter. In this step, you specify which transport adapter the data transfer uses: file system, Message Queuing, or BizTalk. See Configure an adapter. This step is **required**.
3. Create a channel. In this step, you specify the transport details for the exchange. See Creating and configuring channels and Create a channel. This step is **required**.
4. Enable the service. After you enable the service, the service operations (actions) are available for use by the endpoint. See Configure services. This step is **required**.
5. Create and configure an endpoint for the external system. This is only necessary if you do not use the default endpoint that ships in AIF. See Creating and configuring endpoints. This step is **required** if you are not going to use the default endpoint.
6. Configure the endpoint action policy to associate the desired actions with the endpoint. See Configure endpoint action policies. This step is **required** if you are not going to use the default endpoint.
7. For inbound transfers, create the XML message and place it in the appropriate location as defined the channel. This step is **required**.
8. Configure the Microsoft Dynamics AX batch jobs to send and receive messages. For inbound transfers, these jobs move messages from external locations (the file system, the Message Queuing queue, or BizTalk Server) into the internal queues and then save the data into the Microsoft Dynamics AX database. For outbound transfers, these jobs move messages into the internal queues and then to the external locations (the file system, the Message Queuing queue, or BizTalk Server). See Start and stop the asynchronous AIF services. This step is **required**.

Other AIF settings

The following forms can be used when configuring a data exchange regardless of whether you are using the minimal configuration or the extended configuration:

- Use the **Global settings** form to configure global defaults for configuring adapters, actions, resource locks for batch processing, and schema validation, as well as the default encoding format for documents. For more information, see Configure global settings for document exchange.
- Use the **Pipeline components** form to configure optional document transformations, including XSLT style sheet mapping or value substitutions. For more information, see Creating and configuring a pipeline.
- Use the **Value Mapping** form to set up optional predefined value mapping that is available for certain documents. For more information, see About value mapping.

Maintain integration with external software systems

After you have configured a document exchange using AIF, you will need to maintain that integration and possibly troubleshoot any errors that occur. This may include the following tasks:

- Checking error logs and message queues to monitor traffic.
- Stopping and restarting the framework when necessary.
- Reconfiguring the channel and endpoint if conditions change.
- Viewing the document history and deleting old messages.

For more information, see Manage document exchanges in AIF.

See Also

Adapter-based exchanges in AIF

Configure the file system for AIF

In Application Integration Framework (AIF), you can exchange documents with external systems by sending them to or reading them from a file system directory.

When you use the file system adapter to exchange documents, you must add a channel for each file folder (directory) used in the exchange. If a new folder is required, you can use Windows Explorer to create the folder before you set up the channel or you can create the folder when you configure the channel.

Note:

When you submit multiple documents to the file adapter, they are processed in sequence based on file name. The file name is used because the timestamp on each file is not granular enough to be useful. Use sequential file names such as PO_0001 and PO_0002 to control the order of processing.

Change the default owner for objects for an inbound file system transfer

When you use the file system for an inbound transfer, the default owner of objects in the inbound folder must be the user who is sending the files. AIF determines user validity by comparing the sending user to the endpoint users. This ensures that a file is owned by the user who sends it, and not by the Administrators group that the user belongs to.

1. Click **Start** > **Programs** > **Administrative Tools** > **Local Security Policy**.
2. In the console tree, click **Local Policies**, and then click **Security Options.**
3. In the details pane, double-click **System Objects: Default owner for objects created by members of the administrator's group**, and then change the owner from **Administrator's group** to **Object creator**.
4. Log off from the computer, and then log on.

Change the owner of an inbound folder

The account that AOS service is running under must have all permissions (create, read, write, delete) on the inbound folder. Therefore, you must change the default owner of the inbound folder to the AOS service account with the following steps.

1. Create the folder for the inbound file system transfer.
2. Right-click the folder in Windows Explorer, and then click **Properties**.
3. On the **Security** tab, give Full Control permissions to the AOS account. If the outbound folder is on a separate machine, change the share permissions to allow the AOS account to have Full Control permissions, and then click **Advanced**.
4. On the **Owner** tab, set the **Current owner of this item:** field to be the logged-in user who will perform the document exchange.
5. Click **Apply**.
6. Click **OK** to exit the menus.
7. Log off from the computer, and then log on.

Change the owner of an outbound folder

The account that the AOS service is running under must have all permissions (create, read, write, delete) on the outbound folder. Therefore, you must change the default owner of the outbound folder to the AOS service account with the following steps.

1. Right-click the folder in Windows Explorer, and then click **Properties**.
2. On the **Security** tab, give Full Control permissions to the AOS account. If the outbound folder is on a separate machine, change the share permissions to allow the AOS account to have Full Control permissions, and then click **Advanced**.
3. On the **Owner** tab, set the **Current owner of this item:** field to be the logged-in user who will perform the document exchange.
4. Click **Apply**.

5. Click **OK** to exit the menus.
6. Log off from the computer, and then log on.

See Also

Configure document exchanges with adapters in AIF

Configure document exchanges with Web services in AIF

Creating and configuring channels

Configure an adapter

Configure Message Queuing for AIF

An adapter for Microsoft Message Queuing (also known as MSMQ) is included with your Microsoft Dynamics AX installation. To send documents by using Message Queuing, you must install it on a computer on the network and create at least one public queue.

Note:

Whenever the word "queue" is used in this topic, it refers to a queue created and maintained using Message Queuing.

To receive documents by using Message Queuing, you must install Message Queuing and create at least one queue on a computer that has the Application Object Server (AOS) installed, and that has been configured to run the Application Integration Framework (AIF) batch jobs. For more information about these batch jobs, see Start and stop the asynchronous AIF services.

To configure the MSMQ adapter (`AifMSMQAdapter`) for use in a channel, see Configure an adapter and Create a channel.

Signed messages

If you need to send signed messages, such as to other Microsoft Dynamics AX installations, you must run the AOS service under a domain account. By default, the AOS service runs under the Network Service account. Using the default service account can be an issue when you use AIF because when the AOS is running as Network Service, it cannot send signed messages. For more information, see "Install an Application Object Server (AOS) instance" in the Microsoft Dynamics AX Installation Guide.

When you send a signed message, AIF validates that the sending user is a current user in the Active Directory directory service. The Network Service account is not in Active Directory so the message signing process will fail. Microsoft Dynamics AX requires that incoming messages be signed. Therefore, if you want to send documents from one Microsoft Dynamics AX installation to another, the messages must be signed. If the receiver of the documents does not require a signature, you can send documents unsigned as long as the AOS runs under the Network Service account.

The rules for message signature and the AOS account are as follows:

- Unsigned messages can be sent whether the AOS is running as the Network Service account or a domain user.
- Signed messages can only be sent if the AOS is running as a domain user.

Install Message Queuing on a Windows Server 2003 computer

To configure Message Queuing to receive documents, you must install Message Queuing on a computer that has AOS installed. This computer must also be configured to run the AIF batch jobs that execute the AIF services.

1. Click **Start** > **Settings** > **Control Panel** > **Add or Remove Programs** > **Add/Remove Windows Components**.
2. Select **Application Server**.
3. Click **Details**.
4. Select **Message Queuing**.
5. Click **Details**.
6. Select **Active Directory Integration** and **Common**.
7. Click **OK** twice.
8. Click **Next** to install.

Create a queue

A queue for inbound messages can be either private or public. A queue for outbound messages must be public.

1. Click **Start** > **Settings** > **Control Panel** > **Administrative Tools** > **Computer Management**.
2. Under **Services and Applications**, expand the **Message Queuing** folder.
3. To create a new public queue, right-click the **Public Queues** folder and then click **New** > **Public Queue**.
4. To create a new private queue, right-click the **Private Queues** folder and then click **New** > **Private Queue**.

   Any queue used for receiving messages must be located on the same computer as the AOS that is configured to run the batch jobs that operate on the AIF queues. For more information, see Start and stop the asynchronous AIF services.
5. Enter a name for the queue.
6. Click the **Transactional** check box.
7. Click **OK**.

Configure a queue for sending or receiving documents

AIF requires all inbound messages to be authenticated. When using Message Queuing, authenticated messages are passed to AIF through authenticated or unauthenticated queues. AIF does not provide authentication for outbound messages. Therefore, you must configure outbound queues to be unauthenticated.

1. Click **Start** > **Settings** > **Control Panel** > **Administrative Tools** > **Computer Management**.
2. Under **Services and Applications**, expand the **Message Queuing** folder.
3. Right-click the queue that you created and then click **Properties**.
4. If you are configuring an inbound queue, on the **General** tab, select **Authenticated**. If you are configuring an outbound queue, clear **Authenticated**.

   Inbound messages must be authenticated. That is, inbound messages that are not authenticated fail to reach their destination (the local endpoint).
5. On the **Security** tab, set the access control lists (ACL) appropriately for all queues.

   When configuring the ACLs for newly created Message Queuing queues, be sure that the current, logged-in user retains Full Control over each queue. If the current, logged-in user is denied certain privileges, they may become locked out of the queue and unable to make changes.

   By default, Everyone and Anonymous Logon users can send messages to any newly created queue. Only the creator of the queue and an administrator can receive messages from the queues.

   For inbound queues, select **Allow** on **Receive Message** and **Peek Message** for the AOS account (the domain account or Network Service account associated with the AOS instance). Select **Allow** on **Send Message**, **Get Properties**, and **Get Permissions** for endpoint users only. For more information about how to configure endpoint users and trusted intermediaries, see Configure an endpoint.

   For outbound queues, set the ACLs to select **Allow** on **Send Message**, **Get Properties**, and **Get Permissions** for the Anonymous Logon user. Select **Allow** on **Receive Message** and **Peek Message** for endpoint users only.

   If the inbound queue is not on the same computer as the AOS, then two additional entries added to the queue's ACL. First, add the account for the AOS computer (for example, *domain-name**computer-name$*), and select **Allow** on **Peek Message** and **Receive Message**. Second, grant the **Peek Message** and **Receive Message** permissions for the Anonymous Logon user.
6. On the **Security** tab, click **Advanced**.
7. On the **Owner** tab, set the **Current owner of this item:** field to be the logged-in user who performs the document exchange.

8. Click **OK** to exit the menus.
9. Log off and then log back on to the computer.

See Also

Configure document exchanges with adapters in AIF

Configure document exchanges with Web services in AIF

Configure BizTalk for AIF

Application Integration Framework (AIF) ships with a BizTalk adapter that enables the exchange of data between your Microsoft Dynamics AX system and external systems using BizTalk Server. Although this type of exchange uses the BizTalk transport adapter (as opposed to Web services), the exchange can be configured to be synchronous or asynchronous. Asynchronous exchanges use the Microsoft Dynamics AX queues to move messages in and out of the system.

To configure a document exchange using the BizTalk adapter, you must have Microsoft BizTalk Server 2006 and Visual Studio 2005 installed and you must install the AIF BizTalk adapter, if you have not already done so. After you configure AIF, you must also configure a BizTalk assembly (application) to communicate with AIF.

Note:

Only Visual Studio 2005 is supported by BizTalk Server 2006.

Using BizTalk as the transport mechanism in an exchange is useful when you need to make changes or transformations to the data during a transfer or if you have any integration requirements that are not covered by the logic in the AIF documents.

Setting up an AIF exchange with BizTalk includes the following steps:

1. Set up the AIF environment. See How to: Configure AIF for use with BizTalk Server.
2. Create the BizTalk assembly. See How to: Create the BizTalk assembly.
3. Write the BizTalk application. For more information about configuring a document exchange with BizTalk, see the white paper Configure AIF BizTalk Adapter for Data Exchange.

How to: Configure AIF for use with BizTalk Server

The steps to configure Application Integration Framework (AIF) for use with Microsoft BizTalk Server include:

1. Check for prerequisites.
2. Set up the Business Connector proxy.
3. Install the BizTalk adapter on the server.
4. Configure the BizTalk transport adapter.

Check for prerequisites

The following software must be installed to enable a BizTalk document exchange in Microsoft Dynamics AX:

- Microsoft BizTalk Server 2006. For installation instructions, refer to BizTalk Server documentation.
- Visual Studio 2005. This is used to create the BizTalk orchestration that communicates with AIF.

Be sure the following are available in the domain:

- A core Microsoft Dynamics AX installation. This installation must be completed before you install the BizTalk adapter.
- Active Directory directory service configured in native mode.

Set up the Business Connector proxy

The Business Connector proxy is a Microsoft Windows domain account that is used to connect to Microsoft Dynamics AX for applications that require “act-on-behalf-of” functionality for external users or users who have an intermittent network connection.

The Business Connector proxy account should not be set up as a Microsoft Dynamics AX user account. If you have already set up the Business Connector proxy account, you do not need to set it up again.

To set up the Business Connector proxy account, see "Install the .NET Business Connector" in the Microsoft Dynamics AX Installation Guide.

Install the BizTalk adapter

The BizTalk adapter is installed on the application integration server. For more information on installing the BizTalk adapter, see "Install the BizTalk adapter" in the Microsoft Dynamics AX Installation Guide.

Enable the BizTalk adapter

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Transport adapters**.
2. Press CTRL+N.
3. In the **Adapter class** list, select **AifBizTalkAdapter**. There may be a slight delay while Microsoft Dynamics AX scans for adapters.
4. Type a name for the adapter in the **Name** field.
5. To make the adapter available for use in a channel, click **Active**.
6. Close the form.

Configure a channel to use the BizTalk adapter

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Channels**.
2. Press CTRL+N to create a new channel.
3. Select **AifBizTalkAdapter** in the **Adapter** field.
4. Enter the unique identification information for the new channel, including an identifier in the **Channel ID** field and a friendly name in the **Name** field.
5. To activate the channel and allow it to participate in document exchanges, click **Active**.
6. The **Direction** field defaults to **Both** specifying that this channel can be used to send or receive messages. This field cannot be updated.
7. In the **Address** field, enter the name of the BizTalk group that contains the servers with which the adapter communicates. This group name does not need to correspond to a BizTalk group name; however, if all your servers are in the same BizTalk group, you may want the AIF group to match the BizTalk group for convenience.
8. Click **Configure**.
9. Enter the name of the BizTalk server that can connect to this channel. This is the machine name of the BizTalk server. You can enter multiple BizTalk servers in the grid and associate them with the new channel. Press CTRL+N to add each server to the list.
10. Press CTRL+S to save the channel.

Next steps

After you configure AIF for use with BizTalk, you must:

- Create the BizTalk assembly in Visual Studio. For more information, see How to: Create the BizTalk assembly.
- Configure document exchanges. For more information, see Configure document exchanges with adapters in AIF.

For more information about configuring a document exchange with BizTalk, see the white paper Configure AIF BizTalk Adapter for Data Exchange.

See Also

Configure BizTalk for AIF

How to: Create the BizTalk assembly

How to: Create the BizTalk assembly

After the BizTalk adapter has been configured in Application Integration Framework (AIF), you are now ready to create the BizTalk assembly. This is the BizTalk application that communicates with AIF to exchange documents. The BizTalk assembly is written in Visual Studio. To use the

BizTalk application to exchange documents, you must import the Microsoft Dynamics AX schemas for the specific documents that will be exchanged.

Note:

Only Visual Studio 2005 is supported by BizTalk Server 2006.

For more information about configuring a document exchange with BizTalk, see the white paper Configure AIF BizTalk Adapter for Data Exchange.

Create the BizTalk assembly

Create the BizTalk application in Visual Studio 2005.

1. Open Microsoft Visual Studio 2005 and click **File** > **New** > **Project**.
2. Under **Project types**, click the **BizTalk Projects** node.
3. Select **Empty BizTalk Server Project**.
4. In the **Name** field, enter a name for the BizTalk project.
5. In the **Location** field, enter a directory location for the project.
6. In the **Solution Name** field, keep the default solution. You can enter an alternate name if you want the solution name to be different from the project name.
7. Click **OK**.

Add a reference to the AIF schemas

Add a reference to the Microsoft Dynamics AX schemas assembly from the BizTalk project. A reference to this assembly enables you to import a schema for a particular document.

1. In the Visual Studio 2005 Solution Explorer, right-click the **References** node and select **Add Reference**.
2. Click the **Browse** tab.
3. Browse to the Microsoft Dynamics AX install location of <*Microsoft Dynamics AX Installation Directory*>`\Client\Bin\ Microsoft.Dynamics.BizTalk.Adapter.Schemas.dll`.
4. Double-click the assembly (.dll file). It will appear in the **Selected projects and components** grid.
5. Click **OK**. The assembly reference appears under the References node in the Visual Studio Solution Explorer.

Import the schemas

After a reference to the AIF document schemas has been created, you must import the document schemas. In these steps, you will select the appropriate schema for each action based on whether the exchange is inbound or outbound and whether it is synchronous or asynchronous.

1. In the Visual Studio 2005 **Solution Explorer**, right-click the project and select **Add** > **Add Generated Items**.
2. Under **Categories**, the **Add Adapter Metadata** node should be selected. In the **Templates** pane, select **Add Adapter Metadata**.
3. Click **Add**. The Add Adapter Wizard appears.
4. Select the **Microsoft Dynamics AX 2009** adapter and click **Next**.
5. In the **Server name** field, enter the name of the server running the AOS. The name entered in this field should be the machine name of the server.
6. In the **TCP/IP Port** field, enter the port of the AOS server and click **Next**.

   To find the server port, open the Microsoft Dynamics AX 2009 Server Configuration Utility by clicking **Start** > **Administrative Tools** > **Microsoft Dynamics AX 2009 Server Configuration Utility**. Select a configuration in the **Configuration** field to find the port number in the **TCP/IP port** field in the **Settings** frame.
7. In the Schema Import Wizard, you will see a list of the document schemas for document services that are enabled in AIF. For example, if you have enabled the sales order service, you will see it listed as `SalesOrderService`. Expand the service node and select the actions for which you want to import schemas.

   Note:

   In order to import document service schemas into a BizTalk project, you must have at least one service enabled in Microsoft Dynamics AX.
8. Click **Finish**. The schemas for the selected documents can be seen in the Solution Explorer (.xsd file). An orchestration has also been created.

Select the message schema

For each message type that will be processed in the orchestration, you must specify the schema in the multi-part messages types property.

1. In Solution Explorer, double-click the orchestration (.odx file) to open it.
2. In the **Orchestration View**, expand the **Types** node and then expand the **Multi-part Message Types** node. There will be a node for each schema imported, for example, `SalesOrderService_create_Request`, `SalesOrderServer_create_Response`, and so on.
3. Expand each of the child nodes, click the **Body** node, and open the **Type** property drop-down list.

4. In the drop-down list, expand the **Schemas** node and click **Select from referenced assembly**. The **Select Artifact Type** dialog box appears.
5. Expand the **Microsoft.Dynamics.BizTalk.Adapter.Schemas** node and click the **{ } DynamicsAx5** node
6. In the **Type Name** field, select the corresponding schema type, for example, `EntityKey`, and click **OK**.

See Also

Configure BizTalk for AIF

How to: Configure AIF for use with BizTalk Server

Configure global settings for document exchange

In the **Integration Framework global settings** form, you can set global defaults for AIF components including:

- Actions
- Resource locks
- Schema validation
- Default encoding format for documents
- Response cache messages

Set the maximum resource locking interval

The default value for the resource locking interval is 30 minutes. The locking interval only needs to be modified if you have multiple Application Object Server (AOS) instances running. The maximum resource locking interval determines how often the various services lock resources when processing messages. This setting is used by the gateway service to lock channels for inbound and outbound processing, by the inbound processing service to lock channels for inbound processing, and by the outbound processing service to lock endpoints for outbound processing.

An example of resource locking is if you have two AOS instances, AOS1 and AOS2. When AOS1 retrieves messages from the channel, it locks the channel so that AOS2 cannot access those messages and send them out twice. If AOS1 fails, that channel remains locked for the time specified in the global settings. If AOS1 remains unavailable, when the lock expires after 30 minutes, AOS2 can access those messages and deliver them. It is recommended that you monitor the volume of exchanges for your particular system and then set the maximum locking interval.

- If the locking interval is too short, the locks expire faster. When a lock expires and another AOS is available, the same channel or endpoint is processed again. This can lead to wasted resources because the channel or endpoint is processed continuously when no messages are present.

- If the locking interval is too long, it takes the system longer to recover in the event of a system failure.

You should change the maximum locking interval after monitoring the volume of exchanges for your particular system.

For adapter-based exchanges, AIF implements a scheme for locking endpoints and channels to guarantee that messages are processed in a particular order. After one of the AIF services (`AifInboundProcessingService`, `AifOutboundProcessingService`, `GatewaySendService`, and `GatewayReceiveService`) begins processing messages related to a channel or an endpoint, a lock is set for that resource. If the service stops processing the resource before all messages have been transferred, the lock on the resource expires after the maximum locking interval, and another service can begin processing that resource. For more information about the AIF batch services, see Start and stop the asynchronous AIF services.

The default value for the maximum resource locking interval is 30 minutes. If your installation only runs one instance of the AIF services, and those services are interrupted while processing channels or endpoints, the maximum time before the services can begin processing again is 30 minutes.

However, if you are exchanging documents using more than one channel or more than one endpoint and you use multiple AOS instances installed on multiple computers, you may need to set the maximum locking interval for channels and endpoints.

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Global settings**.
2. Enter the time in minutes in the **Maximum resource locking interval (minutes)** field.
3. Press CTRL+S to save.

Set the default encoding format

The default encoding format specifies the encoding of the documents exchanged with an endpoint. When a new endpoint is created, the encoding format for that endpoint defaults to the specified value in the global settings.

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Global settings**.
2. In the **Default encoding format** field, select a supported encoding format from the list to be used as the default on the **Endpoints** form.
3. Press CTRL+S to save.

Set up validation for outbound documents

If you select this option, the XML format for all outbound messages will be validated against the document class schema. For more information about document schemas, see "Document Schema Overview" in the Microsoft Dynamics AX SDK Help.

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Global settings**.

2. Select the **Validate outbound schema** field.
3. Press CTRL+S to save.

Note:

Choosing to validate the schema for every outbound document may negatively impact performance.

Set the response cache lifetime

The response cache lifetime specifies the time in hours that message responses are cached. This setting is only used when you set the `EnableIdempotence` property for a particular service operation to true in the AOT. If a service operation has idempotency enabled, the response message for any message that calls that operation is cached. If a duplicate message is received, AIF generates an error and sends the original response back as part of the generated error.

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Global settings**.
2. In the **Response cache lifetime (hours)** field, enter the number of hours that response messages must be cached.
3. Press CTRL+S to save.

See Also

Use AIF to integrate with external systems

Create and configure local endpoints

The local endpoint refers to your Microsoft Dynamics AX installation. A local endpoint is the origination for sent messages and the destination for received messages. There may be more than one local endpoint, depending on how many companies are configured for your Microsoft Dynamics AX installation.

The name of the local endpoint identifies you to external systems. Therefore, we recommend that the name for the local endpoint be representative of the Microsoft Dynamics AX company name that participates in the exchange.

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Local endpoints**.
2. Press CTRL+N to create a new local endpoint record.
3. Select a company from the **Company** field.
4. Enter a name for the endpoint in the **Local endpoint** field.

   The name for the local endpoint should be representative of the Microsoft Dynamics AX company name that participates in the exchange.
5. Press CTRL+S to save the data.

See Also

Configure document exchanges with adapters in AIF

Configure document exchanges with Web services in AIF

Creating and configuring actions

In Application Integration Framework (AIF), there are two types of actions:

- ServiceOperation
- SendXML

A ServiceOperation action defines an action that is available from a service. A SendXML action defines an action that is independent of any service or class and is used in X++ to send XML (outbound).

The **Actions** form displays all of the actions that are available.

ServiceOperation Actions

The services that ship with Microsoft Dynamics AX come with one or more of the actions in the following table. These actions are ServiceOperation actions and are read-only in the **Actions** form.

Service operation	Description
`create`	Accepts XML data, writes a record to the database, and returns the ID of the new record.
`delete`	Accepts ID values and deletes the associated records.
`find`	Accepts query criteria and returns the matching data.
`findKeys`	Accepts query criteria and returns IDs of the matching data.
`read`	Accepts a list of IDs and returns the associated data.
`update`	Accepts a list of IDs and data and updates the associated records in the database.

Each service does not implement all of the available actions. Rather, each service implements actions based on the requirements of the business process that the service supports. Follow these steps for more information about the service operations that are supported by a service:

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Services**
2. Select a service and click > **Service Operations**. The **Service Operations** form lists all of the available operations.
3. To view the parameters for a service operation, select the operation and click **Parameter schema**.

SendXML action type

You can use the SendXML action type when you want to create an action that is not a service operation. In the **Actions** form you can add, modify, or delete a SendXML action. For more information about creating and configuring a SendXML action, see Create and configure a SendXML action.

Security note for inbound documents

Certain actions, such as creating exchange rates, cause data to be written directly to the Microsoft Dynamics AX database without manual end-user verification. When configuring endpoints and creating new actions, be careful to restrict access to trusted and reliable partners and applications.

Create and configure a SendXML action

The **Actions** form displays all of the actions that are available. ServiceOperation actions are defined in the AOT as part of a service. Therefore, they cannot be modified in the **Actions** form. However, the **Actions** form is used to add, modify, or delete SendXML actions.

In Application Integration Framework (AIF), there are two types of actions: ServiceOperation and SendXML. A ServiceOperation action defines an action that is available from a service. A SendXML action defines an action that is independent of any service or class and is used in X++ to send XML (outbound).

Add a SendXML action

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Actions**.
2. Press CTRL+N to add a new record and click the **General** tab.
3. In the **Action** field, type a unique action identifier.
4. In the **Name** field, type the action name.
5. Select the **Enabled** field to enable the action for use.
6. In the **External identifier** field, type an external identifier name. This is the externally-facing name of the action.
7. In the **Description** field, type a description. The action type defaults to **SendXml**. The action can now be used like other AIF actions and associated with an endpoint, configured with pipeline components, and so on.

Configure a SendXML action

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Actions**.
2. Click the **General** tab. The fields that you can edit are:
   - **Name**
   - **Enabled**
   - **External identifier**
   - **Description**

Delete a SendXML action

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Actions**.
2. Select an action and click ALT+F9.

Note:

You should ensure that an action is not used in an action policy for any endpoints before deleting the action.

Configure an adapter

Adapters are the software components that enable document exchange by communicating with specific transport mechanisms. Microsoft Dynamics AX ships with adapters for the following transports:

- File system (`AifFileSystemAdapter`)
- Microsoft Message Queuing (`AifMSMQAdapter`)
- BizTalk Server (`AifBizTalkAdapter`)

AIF uses these adapters to exchange data with external systems via the specific transport. Therefore, if you implement an exchange with an external system that is required to use BizTalk, you must set up the exchange to use the BizTalk adapter.

AIF also provides the ability for developers to create custom adapters for your own specific needs. After a custom adapter has been created (coded), you add it to the list of available adapters and enable it. Then you create and configure a new channel to use the new adapter. Note that an adapter can only be deleted if it has no corresponding channels.

Configure an adapter

Follow these steps to configure a transport adapter.

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Transport adapters**.
2. Select the adapter that you want to use from the **Adapter class** field. There may be a slight delay while Microsoft Dynamics AX scans for adapters.
3. Type a name for the adapter in the **Name** field.
4. Click **Active** to make the adapter available for use in a channel.
5. If the adapter has been implemented as unidirectional, on the **General** tab, select a direction (inbound or outbound) for the adapter.

   If the adapter is not set to unidirectional, the direction for the adapter appears as it has been implemented and you cannot change it.
6. The **Hosted** check box indicates whether the adapter is hosted.
   - A hosted adapter uses the gateway queue to send and receive message, for example the file system and MSMQ adapters.
   - An adapter that is isolated is external to the AIF gateway service and is implemented so that it controls the sending or receiving of messages. The BizTalk adapter does not use the gateway queue to send and receive messages.

   For more information about the AIF gateway service, see Start and stop the asynchronous AIF services.

Creating and configuring channels

Channels define the transport method that enables messages to move in and out of Application Integration Framework (AIF) to reach the endpoint. The channel defines the specifics for a particular transport adapter. For example, if you create a channel that uses the file system adapter, that channel specifies the directory in which the files will be sent or received. Before configuring a channel, at least one transport adapter must be available. For more information, see Configure an adapter.

A channel may support transfers that are inbound only, outbound only, or both. The configuration settings are different depending on the direction for the transfer in the channel and the type of adapter that the channel uses. An inbound channel may use a response channel to which verification or error responses are sent.

Three adapters are included with each Microsoft Dynamics AX installation to enable transfers by using:

- The file system
- Message Queuing (also known as MSMQ)
- BizTalk

These adapters are available for channel configuration without any customization of your installation.

For more information about how to create a channel, see Create a channel.

See Also

Configure document exchanges with adapters in AIF

Create a channel

Channels define the transport method and transport address that enable messages to move in and out of the framework to reach the endpoint. Before configuring a channel, you must have activated an adapter on the **Transport adapters** form. For more information, see Configure an adapter.

Create a channel to use the file system adapter

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Channels**.
2. Press CTRL+N to create a new channel.
3. Enter the identification information for the new channel, including a unique identifier in the **Channel ID** field and a friendly name in the **Name** field.

4. Select **File System Adapter** in the **Adapter** field.

   Note:

   You must first have activated the file system adapter on the **Transport adapters** form before it appears in the list of available adapters.

5. Click **Active** to activate the channel and allow it to participate in document exchanges.
6. Select from the available directions for the transfers to be performed in this channel (**Inbound**, **Outbound**, or **Both**). The list of available directions depends on the adapter.
7. If you have selected a direction of **Inbound**, select **Parallel processing** to enable inbound messages for this channel to be processed in parallel by multiple AOSs. This means that messages are processed without regard to the order in which they are received or produced. For more information, see AIF performance. When this field is cleared, messages for this channel will be processed sequentially.
8. In the **Address** field, select a directory (file folder) for the channel to use. You can select an existing directory or you can make a new directory by clicking **Make New Folder**.
9. On the **General** tab, set the **Maximum batch size** to the maximum number of messages to be processed at one time from the queues.

   Note:

   You can select **Unlimited** if you do not want to limit this number.

10. If the direction is set to **Inbound**, you can select a response channel. The response channel is used to respond back to the source endpoint with verification results or error responses to inbound transfers. For example, if an external system sends a message that contains the sales order service `read` action and a sales order entity key to an endpoint, a document containing the sales order is sent back to the external system through the response channel of the channel that received the request.

Create a channel to use the MSMQ adapter

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Channels**.
2. Press CTRL+N to create a new channel.
3. Enter the identification information for the new channel, including a unique identifier in the **Channel ID** field and a friendly name in the **Name** field.
4. Select **MSMQ Adapter** in the **Adapter** field.

   Note:

   You must first have activated the MSMQ adapter on the **Transport adapters** form before it appears in the list of available adapters.

5. Click **Active** to activate the channel and allow it to participate in document exchanges.
6. Select from the available directions for the transfers to be performed in this channel (**Inbound**, **Outbound**, or **Both**). The list of available directions depends on the adapter.

7. If you have selected a direction of **Inbound**, select **Parallel processing** to enable inbound messages for this channel to be processed in parallel by multiple AOSs. This means that messages are processed without regard to the order in which they are received or produced. For more information, see AIF performance. When this field is cleared, messages for this channel will be processed sequentially.
8. In the **Address** field, select an existing queue for the channel to use.

   Note:

   When creating the queue in Message Queuing, be sure to select the Transactional field; otherwise, the queue will not be available as a channel address.
9. On the **General** tab, set the **Maximum batch size** to the maximum number of messages to be processed at one time from the queues.

   Note:

   You can select **Unlimited** if you do not want to limit this number.
10. If the direction is set to **Inbound**, you can select a response channel. The response channel is used to respond back to the source endpoint with verification results or error responses to inbound transfers. For example, if an external system sends a message that contains the sales order service `read` action and a sales order entity key to an endpoint, a document containing the sales order is sent back to the external system through the response channel of the channel that received the request.

Create a channel to use the BizTalk adapter

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Channels**.
2. Press CTRL+N to create a new channel.
3. Enter the identification information for the new channel, including a unique identifier in the **Channel ID** field and a friendly name in the **Name** field.
4. Select **File System Adapter** in the **Adapter** field.

   Note:

   You must first have activated the file system adapter on the **Transport adapters** form before it appears in the list of available adapters.
5. Click **Active** to activate the channel and allow it to participate in document exchanges.
6. Select from the available directions for the transfers to be performed in this channel (**Inbound**, **Outbound**, or **Both**). The list of available directions depends on the adapter.
7. If you have selected a direction of **Inbound**, select **Parallel processing** to enable inbound messages for this channel to be processed in parallel by multiple AOSs. This means that messages are processed without regard to the order in which they are received or produced. For more information, see AIF performance. When this field is cleared, messages for this channel will be processed sequentially.

8. In the **Address** field, type the name of the BizTalk group to use.
9. On the **General** tab, set the **Maximum batch size** to the maximum number of messages to be processed at one time from the queues.

   Note:

   You can select **Unlimited** if you do not want to limit this number.
10. Click **Configure**.
11. Enter the names of the computers running BizTalk Server that belong to the specified BizTalk group. Press CTRL+N to add each server to the list.

    Note:

    The **Configure** button is available depending on the type of adapter that you are using in the channel.

See Also

Configure the file system for AIF

Configure Message Queuing for AIF

Configure BizTalk for AIF

Creating and configuring channels

Configure an adapter

Creating and configuring endpoints

Using Application Integration Framework (AIF), you enable document exchanges between endpoints and the local endpoint. The local endpoint refers to your Microsoft Dynamics AX installation. A local endpoint is the origination for sent messages and the destination for received messages.

An endpoint represents the external system that participates in data exchange. An endpoint is the destination for a sent document and the source of a received document.

For more information about creating and configuring endpoints see Create an endpoint and Configure an endpoint

Default endpoint

Microsoft Dynamics AX ships with an endpoint called the default endpoint. Therefore, it is not necessary to create an endpoint for data exchange. Creating and configuring a specific endpoint is optional, and if a message does not specify an endpoint, AIF implicitly uses the default endpoint.

If you disable the default endpoint in Microsoft Dynamics AX, then you must create a specific endpoint and all requests must reference that endpoint or the message will be rejected.

Specific endpoints

You may want to create endpoints for data exchange for a variety of reasons. Endpoints are useful in business-to-business scenarios where you want put restrictions on a data exchange. With endpoints you can:

- Set constraints on the endpoint. Constraints specify that messages can only be sent to or received from specific customers, vendors, or warehouses.
- Implement a higher level of security by assigning specific users or groups to an endpoint. This ensures that only certain users can send messages to AIF.
- Assign actions to an endpoint to restrict the types of data exchanges in which the endpoint can participate.
- Set data policies on an endpoint to restrict the data that can be retrieved or updated.
- Implement custom configuration requirements like XSLT transformations, value mapping, and custom pipeline components.
- Customize the message logging for each action associated with an endpoint.

Prerequisites

Before you can create an endpoint, the following must already exist and be configured:

- A local endpoint.
- A named service operation for the exchange, for example, `CustCustomerService.read`. For more information, see Creating and configuring actions.
- Microsoft Dynamics AX users that will be associated with the endpoint.
- An outbound channel must be already be defined if needed.

Endpoint information

When you configure an endpoint, you enter information about the following:

- Identifying information for the endpoint, including a unique identifier, a friendly name, the active status, error handling information, the associated channel, and the Microsoft Dynamics AX company identification.
- Constraints on the endpoint that restrict document exchange by defining valid Microsoft Dynamics AX customers, vendors, or warehouses.
- Microsoft Dynamics AX users and trusted intermediaries that are allowed to submit documents for the exchange.
- Endpoint action policies that relate actions to the endpoint.
- Pipeline components for any optional document transformations for the action related to the endpoint.

- Endpoint action data policies (also known as data policies) that define which fields in a document are allowed or required to participate in the exchange.
- Document configuration options, including value mapping. For more information, see About value mapping and "Set up external codes for AIF" in Applications and Business Processes.

Create an endpoint

Before you can create an endpoint, the following must already exist and be configured:

- A local endpoint.
- An enabled service operation for the exchange, for example, `CustCustomerService.read`. For more information, see Creating and configuring actions.
- Microsoft Dynamics AX users that will be associated with the endpoint.
- An outbound channel must be already be defined if needed.

Create the endpoint

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Press CTRL+N to create a new endpoint.
3. In the **Endpoint ID** field, type a unique identifier for the endpoint.
4. In the **Name** field, type a friendly name.
5. Select **Propagate errors** to return detailed error messages to the endpoint.

   Note:

   By default, Microsoft Dynamics AX logs detailed errors and sends a generic error back to the endpoint. Enabling this field will send the detailed error to the endpoint and could be a potential security risk. Only select this field if it is acceptable to send detailed error information to an endpoint.
6. Do not select **Intercompany organization** unless the endpoint is to be used in an intercompany transfer. For more information about these transfers, see "Manage intercompany sales orders" or "Manage intercompany purchase orders" in the Application and Business Processes Help.
7. If you select **Intercompany organization**, then you must select a company in the **Company** field.
8. In the **Outbound channel ID** field, select an outbound channel. You must select a channel if the endpoint will be used in an adapter-based exchange and data is to be sent outbound from Microsoft Dynamics AX to the endpoint. If the endpoint is participating only in Web services-based exchanges, the outbound channel is not necessary.
9. In the **Local endpoint ID** field, select the identifier for the local endpoint (your system) that participates in the exchange with the endpoint that you are configuring.
10. In the **Default encoding format** field, select the encoding format for this endpoint.

11. On the **Constraints** tab, enter the data constraints for the endpoint to restrict the data that can be processed by the endpoint. To allow data to be exchanged regardless of any associations, click **No constraints**. For more information, see Configure an endpoint.

    After selecting, the **No constraints** check box becomes unavailable. However, if you add constraints later, the check box clears itself.

12. On the **Overview** tab, select **Active** to enable the endpoint to participate in data exchanges.

13. On the **Users** tab, enter information to restrict users that are authorized to initiate transactions for the endpoint. In the **User type** field, select either **User** or **User group**.

    You can also designate trusted intermediaries on the **Users** tab. Trusted intermediaries are middleware applications that reside between external endpoints and Application Integration Framework (AIF), that is, they are Microsoft Dynamics AX users (or user groups) that are authorized to submit inbound requests on behalf of the endpoint. For more information about trusted intermediaries, see Security considerations for AIF.

    Notes:

    - When configuring users on an endpoint, keep in mind that these Microsoft Dynamics AX users may represent outside interests and must have permissions set appropriately. For more information about configuring Microsoft Dynamics AX users, see "Setting up and maintaining security" in the Microsoft Dynamics AX Installation Guide and the following topics in the System and Application Setup Help: "Manage security permissions for user groups and domain combinations," "Manage user groups," and "Manage users."
    - You must also set the appropriate security keys and record-level security for any users that are granted access to Microsoft Dynamics AX through AIF to help prevent unauthorized data access. For more information, see "Manage record level security" in the Application and Business Processes Help. Certain actions cause data to be written directly to the Microsoft Dynamics AX database without manual end-user verification (for example, creating exchange rates). When configuring endpoints and creating new actions, be especially careful to restrict access to trusted and reliable partners and applications.

14. Click **Action policies** to configure actions on the endpoint with the **Endpoint Action Policies** form. Examples of actions include the service operations `read` and `create`. For information on setting up action policies, see Configure endpoint action policies.

15. From the **Endpoint Action Policies** form you can select an action and click **Data Policies** to enter the data policy, that is, information about which fields are required and which are optional in the document to be exchanged. For details on setting up data policies, see Configure endpoint action data policies.

16. From the **Endpoint Action Policies** form, you can click **Configure** to perform document-specific configuration, including value mapping. Value mapping is the translation of field data values based on business rules, for example, translating internal item numbers to vendor-specific item numbers or industry standard numbers depending on the trading partner. For more information, see Configure endpoint action data policies.

Enabled fields

The fields that are enabled for an endpoint depend on the type of endpoint that you are configuring as shown in the following table.

Field	Default endpoint	Standard endpoint	Intercompany endpoint
Endpoint ID	No	Yes (when adding new endpoint)	Yes (when adding new endpoint)
Name	No	Yes	Yes
Active	Yes	Yes	Yes
Propagate errors	Yes	Yes	Yes
Intercompany organization	No	Yes	Yes
Company	No	Yes	Yes
Outbound channel ID	No	Yes	No
Local endpoint ID	No	Yes	No
Default encoding format	No	Yes	No

Troubleshooting Trusted Intermediary

If you receive an error on an inbound message and it was submitted by a trusted intermediary, it may be due to the fact that the submitting user of the message and the user specified in the message `SourceEndpointUser` element are not in the same Microsoft Dynamics AX domain.

The trusted intermediary is a valid Microsoft Dynamics AX user that is allowed to submit AIF messages. Typically, the trusted intermediary does not have access to the AIF services. The source endpoint user is a valid Microsoft Dynamics AX user that has access to the AIF services. If the user submitting the message (the trusted intermediary) is in a different domain than the source endpoint user, you may receive an error.

To resolve this problem, give the source endpoint user open domain access permissions. To locate the security key for these permissions, use the following steps.

1. Open **Administration** > **Setup** > **User groups**.
2. Click **Permissions**.

3. On the click **Permissions** tab, expand the **Administration** node and select **Open domain access**.

   Note:

   This refers to Microsoft Dynamics AX domains and not Active Directory domains. Microsoft Dynamics AX domains are defined in **Administration** > **Setup** > **Domains**.

For more information, see "Manage security permissions for user groups and domain combinations" in and "Manage domains" in the System and Application Setup Help.

See Also

Configure endpoint action policies

Configure endpoint action data policies

Creating and configuring a pipeline

Configure an endpoint

When you configure an endpoint, you enter information in the **Endpoints** form about the following:

- Identifying information for the endpoint, including a unique identifier, a friendly name, the active status, error handling information, the intercompany status, local endpoint, Microsoft Dynamics AX company identification for an intercompany exchange, outbound channel information, and encoding format for the transfer.
- Constraints on the endpoint that restrict document exchange by defining valid Microsoft Dynamics AX customers, vendors, or warehouses.
- Microsoft Dynamics AX users and trusted intermediaries that are allowed to submit documents for the exchange.
- Endpoint action policies that relate actions to the endpoint.
- Pipeline components for an action related to the endpoint.
- Endpoint action data policies that define which fields in a document are allowed or required for the exchange.

Available tabs

In the **Endpoints** form, user interface tabs are available depending on what type of endpoint you select in the **Overview** tab.

Endpoint type	Tabs enabled
Default endpoint	• **Overview** • **General**
Specific endpoint	• **Overview** • **General** • **Constraints** • **Users**
Specific intercompany endpoint	• **Overview** • **General** • **Constraints**

Default endpoint

Microsoft Dynamics AX ships with a default endpoint that can be used to enable data exchanges as soon as services are generated. The default endpoint has minimal configuration options, and you cannot add any constraints or users to the default endpoint. By default, all actions are enabled for the default endpoint but you must still enable the service actions on the **AIF Services** form.

You can change the following options on the default endpoint:

- In the **Endpoints** form, you can update the **Active** field and the **Propagate errors** field. For more information about these fields, see Create an endpoint.
- In the **Endpoint Action Policies** form, you can enable and configure actions.
- In the **Endpoint action data policies** form, you can define which fields are allowed or required in the data exchange.
- In the **Pipeline components** form, you can define data transformations for inbound or outbound exchanges.
- In the **Parameter Schemas** form, you can view the schema of the action parameters and the return value and optionally save them to an .xsd file.
- In the **Value Mapping** form, you can configure value mapping.

Configure endpoint identification

1. Click **Basic** > **Setup** > **Application Integration Framework**.> **Endpoints**.
2. Press CTRL+N to create a new endpoint, or select an existing endpoint from the list to be modified.
3. Enter the information for the endpoint on the **General** tab, including the unique endpoint identification information, the friendly name for the endpoint, the local endpoint ID, and the default encoding format.
4. Select **Propagate errors** to return detailed error messages to the endpoint.

   Note:

   By default, Microsoft Dynamics AX logs detailed errors and sends a generic error back to the endpoint. Enabling this field will send the detailed error to the endpoint. Only select this field if it is acceptable to send detailed error information to an endpoint.
5. For intercompany transfers, select **Intercompany organization** if the endpoint is a company within your Microsoft Dynamics AX installation, and select a company in the **Company** field.
6. Do not select **Intercompany organization** unless the endpoint is to be used in an intercompany transfer. For more information about these transfers, see "Manage intercompany sales orders" or "Manage intercompany purchase orders" in the Application and Business Processes Help.
7. Selecting an outbound channel is not required for all exchanges. However, if your Microsoft Dynamics AX installation is sending messages to this endpoint (for example, if you are configuring an outbound-only exchange), you must select a channel in the **Outbound channel ID** field.
8. In the **Local endpoint ID** field, select the local endpoint that participates in exchanges with this endpoint.

   Note:

   There may be more than one local endpoint configured for your Microsoft Dynamics AX installation. If that is the case, then make sure to select the correct local endpoint to participate in the exchange with the endpoint that you are configuring.
9. View the pre-populated setting for encoding format in the **Default encoding format** field. This setting defaults to the value set on the **Integration Framework global settings** form.

Configure constraints on an endpoint and activate the endpoint

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Press CTRL+N to create a new endpoint, or select an existing endpoint from the list to be modified.

   Note:

   You cannot add constraints to the default endpoint so the **Constraints** tab will not be enabled if the default endpoint is selected.
3. On the **Constraints** tab, select **No constraints** to clear the form and allow data to be exchanged regardless of any association. The **No constraints** check box becomes unavailable. However, if you add constraints later, the check box clears itself.
4. Press CTRL+N to create a new constraint.
5. Choose the **Constraint type** (**Vendor**, **Customer**, or **Warehouse**).
6. Select a **Constraint ID** from the list. The **Name** field is completed when you select the **Constraint ID**.
7. On the **Overview** tab, select **Active** to activate the endpoint.

   Messages flow through the framework from the local endpoint to and from any active endpoints.

Configure users and trusted intermediaries for an endpoint

You must enter information for at least one endpoint user or trusted intermediary who is authorized to initiate transactions for the endpoint on the **Users** tab.

Notes:

- When configuring users on an endpoint, remember that these Microsoft Dynamics AX users may represent outside interests and must have permissions set appropriately. For more information about configuring Microsoft Dynamics AX users, see "Setting up and maintaining security" in the Microsoft Dynamics AX Installation Guide and the following topics in the System and Application Setup Help: "Manage security permissions for user groups and domain combinations," "Manage user groups," and "Manage users."
- You must also set the appropriate security keys and record-level security for any users that are granted access to Microsoft Dynamics AX through Application Integration Framework (AIF), to help prevent unauthorized data access. For more information, see "Manage record level security" in the Application and Business Processes Help.

- Certain actions cause data to be written directly to the Microsoft Dynamics AX database without manual end-user verification (for example, creating exchange rates). When configuring endpoints and creating new actions, be especially careful to restrict access to trusted and reliable partners or applications.

Configure an endpoint user

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Press CTRL+N to create a new endpoint, or select an existing endpoint from the list to be modified.

   Note:

   You cannot add users or user groups to the default endpoint so the **Users** tab will not be enabled if the default endpoint is selected.
3. Click the **Users** tab.
4. Under **Endpoint users**, in the **User type** field, select either **User** or **User group**.
5. In the **Application user or group** field, select a valid Microsoft Dynamics AX user or user group name. The **Name** field is filled in automatically.

   For more information about Microsoft Dynamics AX users and user groups, see "Setting up and maintaining security" in the System and Application Setup Help.

Configure a trusted intermediary

Trusted intermediaries are middleware applications that reside between external endpoints and AIF. That is, they are Microsoft Dynamics AX users (or user groups) that are authorized to submit inbound requests on behalf of the endpoint. A trusted intermediary prevents an unauthorized user from accessing AIF and is typically used in a business-to-business data exchange scenario. For more information about trusted intermediaries, see Security considerations for AIF.

For more information about Microsoft Dynamics AX users and user groups, see "Setting up and maintaining security" in the System and Application Setup Help.

1. Under **Trusted intermediaries**, select **Use trusted intermediary** to enable a trusted intermediary for exchanges with this endpoint.

   Note:

   If the **Use trusted intermediary** box is checked, there must be at least one entry in the **Trusted intermediaries** grid.
2. Under **Trusted intermediaries**, in the **User type** field, also select either **User** or **User group**.
3. In the **Application user or group** field, select a valid Microsoft Dynamics AX user or user group name. The **Name** field is filled in automatically.

Configure action policies, data policies, and pipeline components for an endpoint

1. After you have completed the preceding steps, click **Action policies** to configure actions on the endpoint.

   For more information about action policies, see Configure endpoint action policies.
2. Press CTRL+S to save the action policy.
3. From the **Endpoint Action Policies** form, select an action, press CTRL+S, and click **Data Policies** to enter information about which fields are required and which are optional in the document to be exchanged.
4. Click **Configure** to perform document-specific configuration, including value mapping, for the document exchange.

   Value mapping is translation of field data values based on business rules, for example, translating internal item numbers to vendor-specific item numbers or industry standard numbers depending on the trading partner. For more information, see Configure endpoint action data policies.
5. Click **Inbound Pipeline** or **Outbound Pipeline** to configure the pipeline components for any custom transformations performed on the document. For more information, see Creating and configuring a pipeline.
6. Click **Parameter schema** to see a list of parameters and the return value for the selected action. On the **Parameter Schemas** form, click **View schema** to view the XML schema for the selected parameter or return value. On the **Schema** form, you can click **Save as** to save the schema as an .xsd file.

Give the endpoint user access to Business Connector when using Web services

When you configure an endpoint for Web services, you must configure an endpoint user and/or trusted intermediary, as outlined above. Next, you must give that Microsoft Dynamics AX user or user group access to the Business Connector.

1. Click **Administration** > **Setup** > **User groups** and select the user group for the endpoint, or the user group that contains the user for the endpoint.
2. Click **Permissions**.
3. On the **Permissions** tab, select **Business Connector** and then select **Full control**.
4. Click **Cascade**.

Disable an endpoint

To disable an endpoint, follow these steps.

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Select an endpoint in the grid and clear the **Active** field.

Important:

For adapter-based exchanges, outbound messages are processed completely even when the endpoint is disabled during processing and a response may still be sent. To ensure that no data is sent from AIF when disabling an endpoint, first disable the batch processing jobs. Be sure that there are no outbound messages in the queue, disable the endpoint, and then restart the batch processing jobs.

See Also

Create and configure local endpoints

Creating and configuring actions

Configure endpoint action policies

Configure endpoint action data policies

Creating and configuring a pipeline

Configure global settings for document exchange

Configure endpoint action policies

To enable exchange of documents for endpoints, you must select which actions an endpoint may perform. This process is called configuring endpoint action policies.

Typically, you associate service actions with an endpoint. These actions are defined as ServiceOperation actions in the **Actions** form. For example, the `CustCustomerService.create` action creates a new customer in Microsoft Dynamics AX. ServiceOperation actions are registered automatically so you do not need to create or enable them.

You may also associate a SendXML action with an endpoint. A SendXML action defines an action that is independent of any service or class and is used in X++ to send XML (outbound).

For more information about actions, see Creating and configuring actions.

Associate an action with an endpoint

The default endpoint automatically has all service actions enabled. However, you must ensure that the service associated with the actions that you want to use is enabled. For more information, see Configure services.

Follow these steps to configure actions on an endpoint other than the default endpoint.

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Select the endpoint to configure and click **Action policies**. The **Overview** tab lists the available actions that are currently associated with the endpoint.

   Note:

   By default, all actions for all services as defined in the AOT are associated with the default endpoint. You do not need to enable them in the **Actions** form.
3. Press CTRL+N to enter a new action policy.
4. Click the **General** tab. In the **Action ID** field, select the action that you want to associate with the endpoint.

   The name of the action in the **Action ID** field and the name of the service class in the **Class name** field cannot be changed.
5. Select the **Is default policy** check box to use the default data policy. The default data policy specifies that all fields defined in the message schema will be used in the exchange. If you clear this field, the **Data Policies** button is enabled, and you can modify the data policy for the action.

   Note:

   This field is only editable for the default endpoint. By default, the **Is default policy** check box is selected for all actions associated with the default endpoint. If you want to modify the data policy for an action associated with the default endpoint, you must clear this field.
6. You can change the status of an action associated with the endpoint to **Enabled**, **Disabled**, or **Hold**.
   - Select **Enabled** to make the action active for this endpoint.
   - Selecting **Disabled** has the same effect as if the action was not configured on the endpoint.
   - Select **Hold** to prevent outbound documents from being passed to the adapter and inbound documents from being passed to the document class.

     If the action status is **Hold**, the document is held in the queue and may be examined and resubmitted. For more information, see Edit and resubmit messages in the queues. For a synchronous exchange such as a Web service, an error message is generated for the hold condition.

   In the **External identifier override** field, type an action identifier to override the **External identifier** field on the **Actions** form. Any messages referencing this endpoint and action must use the external identifier in the <Action> tag.
7. Select the **Automatically respond to errors** check box to send any errors that are encountered back to the caller.
8. Press CTRL+S to save.

Select a logging mode for an action on an endpoint

The logging mode defines how messages are logged on a per action per endpoint basis.

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Select an endpoint and click **Action policies**.
3. Select an action on the **Overview** tab.
4. On the **General** tab, select a logging mode for the endpoint action.
   - **Log Original** captures only the information for the initial document exchange.
   - **Log All** captures information about every transfer including all the different versions of a document, for example, the submitted XML, the XML generated after each pipeline transformation is applied, and so on.
   - **Log None** stores no data for this action and endpoint.

To view the document history by message or by document, click **Basic** > **Periodic** > **Application Integration Framework** > **Document history**.

Configure document-specific options including value mapping

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Select an endpoint and click **Action policies**.
3. Select an action on the **Overview** tab and click **Configure** to perform document-specific configuration, including value mapping.

   Value mapping is the translation of field data values based on business rules, for example, translating internal item numbers to vendor-specific item numbers or industry standard numbers depending on the trading partner. For more information, see About value mapping and "Set up external codes for AIF" in the Application and Business Processes Help.

Configure data policies for an action on an endpoint

For more information about data policies, see Configure endpoint action data policies.

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Select an endpoint and click **Action policies**.
3. Select an action on the **Overview** tab, and click **Data Policies** to configure required and allowed fields for the document transfer.

   Note:

   If the **Is default policy** check box is selected, the endpoint will use the default policy of all fields for that action. If you want to set specific data policies for the endpoint, you must clear this field first.

Configure endpoint action data policies

When you set up a document exchange in Application Integration Framework (AIF), you decide, on a field-by-field basis, which data fields are transferred. This is known as the data policy. The data policy is defined for each action on each endpoint. You configure the data policy on the **Data Policies** form.

Default data policy

In Microsoft Dynamics AX, an action associated with the default endpoint can use a default data policy. This is specified by selecting the **Is default policy** field in the **Data Policies** form.

For a default endpoint action, using the default data policy means that the message uses the full document schema. Required fields as defined in the schema must be in the message. The message can contain any number of optional fields.

If you want to use a specific data policy for an action associated with the default endpoint, you must clear the **Is default policy** field and modify the action data policy as you would for any other endpoint.

Mandatory/required fields

There are two types of qualifiers for data fields on the **Data Policies** form: required and enabled. These have different meanings and effects depending on the direction of the transfer.

If a data field is allowed to be included in an inbound exchange, it is said to be enabled. For inbound documents, only fields that are enabled are allowed to be submitted by the endpoint. If a document is received that includes fields that are not enabled, the document is rejected and an exception is logged.

There are two terms that are used when discussing whether a field is required in a document: mandatory fields and required elements.

Term	Location	Description
Mandatory field	Database	Database field that has the `Mandatory` property set to `Yes`.
Required element	XML document	Element required to be present in the XML document to satisfy the schema. Required elements often correspond to mandatory fields in the database. A database field that is mandatory but that can be defaulted does not have to be required in the XML document.

Note:

For inbound documents, mandatory fields (that is, fields required by the Microsoft Dynamics AX database) should be set to **Enabled** and **Required** on the **Data Policies** form if they cannot be set by default in the database. For outbound documents, the fields to be sent must be set to **Enabled**.

Required fields and document direction

For inbound documents only (processed by a `create` action, for example), fields may be designated as **Required** if the document class defines them as mandatory (they are required for the database record to be inserted or updated and they cannot be defaulted). Additionally, the XML document may specify required elements depending on the business logic in the document class. You can also specify additional required elements by selecting **Required** for the field on the **Endpoint action data policies** form. However, you cannot use the data policy to make an element optional if it is required by the document class.

For an inbound document, fields that are enabled but not required are optional to the exchange. Fields in an inbound exchange that are required are automatically designated as enabled - if the document does not contain these fields, the document is rejected.

The concept of required fields does not apply to outbound transfers. For outbound documents, only fields that are enabled are included in the exchange.

Note:

When you clear the **Enabled** check box for a field used for calculating the value of another field, you may also need to clear the **Enabled** check box for the calculated field, so that unauthorized users may not be able to deduce the value of the original field that is not enabled. For more information about the calculated fields available in each document, see "Standard Axd Documents" in the Microsoft Dynamics AX SDK Help.

Configure data fields for an inbound document

You must first configure an endpoint and enable the actions for the exchange.

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Select an endpoint, click **Action policies**, and select an action.
3. Click **Data Policies** and then click **Data Policies** to configure required and allowed fields for the document transfer.

   Note:

   If the default endpoint is selected, all actions use the default data policy by default and the **Is default policy** field is selected for all actions. To configure an action data policy for the default endpoint, you must clear the **Is default policy** field to enable the **Data Policies** button.

   For fields that are required to be present in the XML document according to the document class, select the **Enabled** and **Required** check boxes.
4. For other fields in the document, you can select **Required** if the field is required for the document exchange (**Enabled** is automatically set).

   Note:

   If you find that your needs for the document transfers change, you can clear the **Required** check box.
5. Click **Set** to clear or select all fields at one time.Configure data fields for an outbound document exchange

You must first configure an endpoint and enable the actions for the exchange.

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Select an endpoint, click **Action policies**, and select an action.
3. Click **Data Policies** and then click **Data Policies**
4. Select **Enabled** for each field to be included in the document transfer.

   Note:

   For an inbound document, fields that are enabled but not required are optional in the exchange. The concept of required fields does not apply to outbound transfers. Only fields for which you have selected the **Enabled** check box are sent in the transfer.
5. Click **Set** to clear or select all fields.

See Also

Configure endpoint action policies

Configure an endpoint

Creating and configuring a pipeline

A pipeline consists of a set of components that transform XML documents as they flow in or out of Microsoft Dynamics AX through Application Integration Framework (AIF). In addition to the pipeline components that ship with Microsoft Dynamics AX, the architecture of the pipeline allows developers to create and configure custom components to transform documents.

A separate pipeline consisting of one or more pipeline components may be specified for every endpoint action policy, which enables custom transformations on a per-endpoint basis. You can specify separate components for inbound documents and outbound documents.

Two pipeline components are installed with Microsoft Dynamics AX:

- A component for value substitution
- A component that enables Extensible Stylesheet Language Transformations (XSLT) document transformations

The `AifValueSubstitutor` pipeline component allows you to substitute one character string for another character string in a given field. This enables you to change field values (an item code, for example) in an inbound or outbound message to match the requirements of the system receiving the data. To apply XSLT document transformations, you must first import an XSLT style sheet into Microsoft Dynamics AX and then specify the `AifXMLTransform` pipeline component for the desired endpoint action policy.

Prerequisites

You can automatically configure pipeline components for actions that are associated with the default endpoint. Otherwise, before configuring any pipeline components, you must have the following:

- A local endpoint. For more information, see Create and configure local endpoints.
- An endpoint with an action policy and a data policy. For more information, see Configure an endpoint, Configure endpoint action policies, and Configure endpoint action data policies.

Create a pipeline

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints** and select an endpoint.
2. Click **Action policies** and select an action.
3. Click **Inbound Pipeline** or **Outbound Pipeline** depending on whether you want the transformations to occur on inbound documents or outbound documents.
4. Press CTRL+N to create a new pipeline component entry.
5. Select a component in the **Class name** field. There may be a delay while the system scans the AOT for pipeline components.
6. In the **Description** field, type a description of the pipeline.

7. Press CTRL+S to save and enable the **Configure** button.
8. Each pipeline component has different configuration requirements, so you see a different form when you click **Configure** for any pipeline component. For more information, see Configure a pipeline.

Configure a pipeline

Configuring a pipeline for an action on an endpoint involves specifying the pipeline components for a transformation of the document, in execution order, on the **Pipeline components** form. You can define pipeline components for inbound or outbound actions separately. For more information about creating a pipeline, see Creating and configuring a pipeline.

Two pipeline components are included with Microsoft Dynamics AX. You can configure these pipeline components to perform value substitution and XSLT transformations. Other custom pipeline components may be developed for your system by your team or outside consultants or partners. Configuration of any custom pipeline component depends entirely on the implementation of that component.

The two pipeline components available with your Microsoft Dynamics AX installation are:

- `AifValueSubstitutor` - For simple string mapping of field values.
- `AifXMLTransform` - For XSLT transforms of XML documents.

You can include as many pipeline components as you need to transform the document to meet the needs of the exchange.

Each pipeline component has different configuration requirements, so you see a different form when you click **Configure** for any pipeline component.

Configure value substitution

Before you can configure a value substitution pipeline, you must define lookup values. For more information, see About value lookups.

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints** and select an endpoint.
2. Click **Action policies** and select an action.
3. Click **Inbound Pipeline** or **Outbound Pipeline**.
4. In the **Class name** field, select `AifValueSubstitutor`. If there is no pipeline, press CTRL+N to add a new pipeline.
5. Press CTRL+S to save.
6. Click **Configure**.
7. On the **Pipeline Value Substitution Parameter Selection** form, select the parameter in the **Parameter name** field.
8. Click **Configure value substitution** to display the **Pipeline value substitution** form.

9. In the **Lookup table ID** field, select the lookup table identification for the value lookup table (that you entered on the **Value lookup** form) for the fields requiring value substitution. For more information about configuring value lookups, see About value lookups.

   Notes:

   - Values for **Lookup table ID** are filtered by type. If no values are displayed for **Lookup table ID**, you may need to return to the **Value lookup** form and enter a value for **Type** on the **General** tab.
   - On the **Pipeline value substitution** form, the following read-only fields appear:
   - **Element name** - The name of the data field.
   - **XPath** - Specifies where the data field fits into the schema hierarchy.
   - **Type** - The Microsoft Dynamics AX data type.

Configure an XSLT transform

To configure a transformation pipeline, you must first import the XSLT into the repository. Then you must create the transformation pipeline based on the XSLT. For more information about security best practices when implementing transformations, see Security considerations for AIF.

Add an XSLT style sheet to the XSLT repository

1. Click **Basic** > **Setup** > **Application Integration Framework** > **XSLT repository**.
2. Click CTRL+N to create a new record.
3. In the **XSLT ID** field, enter a unique identifier for the XSLT transform.
4. In the **Name** field, enter a text description for the transform.
5. Click **Import** and specify the file name of the XSLT style sheet for the transform.
6. Click **View** to view the XML for the style sheet, and then click **Save as** to export the XML to an .xsl file.

Create the pipeline

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints** and select an endpoint.
2. Click **Action policies** and select an action.
3. Click **Inbound Pipeline** or **Outbound Pipeline**.
4. In the **Class name** field, select `AifXMLTransform`. If there is no pipeline, press CTRL+N to add a new pipeline.
5. Click CTRL+S to save.
6. Click **Configure**.

7. On the **Pipeline XSLT transform** form, select **Apply transform to parameter** if the transform applies to an action parameter. If this field is not selected, the transformation will apply to the entire document.
8. If you select **Apply transform to parameter**, then you must select a parameter from the **Parameter name** field.
9. In the **XSLT ID** field, select the identification of the XSLT transform that you entered on the **XSLT repository** form.
10. If you want any Microsoft Visual Studio scripts in the XSLT file to be executed, select **Scripting enabled**.
11. Press CTRL+S to save.

Note:

When an XSLT pipeline transformation run, errors are logged only if the component throws an exception. If you use an incorrect XSLT, an exception will not be generated. An XSLT only transforms matching nodes; if there are no matching modes, then no transformation is applied and no error is generated.

See Also

Creating and configuring a pipeline

About value lookups

About value lookups

With value lookups, you can substitute one character string for another in any field of a document. You can implement value lookups using the `AifValueSubstitutor` pipeline component. For more information, see Creating and configuring a pipeline and Configure a pipeline.

You can also create value lookups for any pipeline component, if you know the data type for the field. After creating a value lookup using the **Configure value lookup** form, you relate that value lookup to the pipeline component by entering the value lookup identification when you configure the pipeline component.

Configure value lookups for a pipeline component

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Value lookup**.
2. Press CTRL+N to create a new line.
3. On the **Overview** tab, enter:
   - A new identification in the **Lookup table ID** field.
   - A name for the table in the **Name** field.

4. On the **General** tab, select the Microsoft Dynamics AX data type from the list of available data types. For the data types that reference a table in Microsoft Dynamics AX, the internal values are populated from that table.
5. Enter the internal values and the external values for the string substitution in the lower pane of the form.

Use a value lookup table with the AifValueSubstitutor pipeline component

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints** and select an endpoint.
2. Click **Action policies** and select an action.
3. Click **Inbound Pipeline** or **Outbound Pipeline**.
4. In the **Class name** field, select `AifValueSubstitutor`. If there is no pipeline, press CTRL+N to add a new pipeline and press CTRL+S to save.
5. Click **Configure**.
6. On the **Pipeline Value Substitution Parameter Selection** form, select the parameter in the **Parameter name** field.

   The **Pipeline value substitution** form is populated with the data fields that are **Enabled** on the **Data Policies** form for the document.
7. Enter the **Lookup table ID** for each data field to be substituted. This is the identifier you entered on the **Value lookup** form.
8. In the **XPath** field, you can view the location in the XML schema hierarchy where the element resides.
9. In the **Type** field, you can view the Microsoft Dynamics AX data type.

See Also

Creating and configuring a pipeline

Configure a pipeline

About value mapping

Value mapping is the translation of field data values that is based on business rules; for example, translating internal item numbers to vendor-specific item numbers or industry-standard numbers, depending on the trading partner.

Value mapping can be performed on inbound and outbound XML documents and is configured on each document endpoint. Value mapping creates a translation index between the specific field in Microsoft Dynamics AX and an external field in the document. This index enables more flexibility when you are handling various internal, vendor-based, or industry-based codes.

Value mapping example

One vendor requires all item numbers on the outbound purchase requisition from Microsoft Dynamics AX to be designated by using the vendor's item number system, whereas another vendor (which is a separate endpoint) does not have this requirement, and a third vendor (also a separate endpoint) requires a common industry item number system on the purchase requisition lines.

To handle these varying requirements, each purchase requisition document can be configured differently for each endpoint, and the item number value mapping settings can be configured to reflect the vendor requirements.

When the outbound purchase requisition is generated, the active endpoint configuration translates the internal item number codes to the codes that are specified in the **Value Mapping** form. This translation ensures that each vendor receives the purchase requisition in the format that their system requires.

You can map the values of the fields shown in the following table

Type	Field
Trading partners	• Vendor Account number • Customer Account Number
Addresses	• Country code • County code • State code • Zip/postal code
Items	• Item number • Units • Warehouse numbers
Other data	• Currency code • Delivery Methods • Terms of delivery • Misc. charges

Value mapping for document transformation

Value mapping is configured in the **Value Mapping** form. For more information about the value mapping form, see topic "Value mapping (form)" in the Application and Business Processes Help.

This section describes forms that are used to set up value mapping for endpoints and external codes for different fields used in the documents.

External Codes

You set up, define, and maintain external codes in the **External codes** form. These external codes are for different fields used to send and receive specific documents electronically through Application Integration Framework (AIF).If it is necessary, set up external codes for:

- Trading partners (vendor and customer account numbers)
- Addresses (countries/regions, counties, states, and Postal/ZIP Codes)
- Inventory (item numbers, bar codes, item units, and warehouses)
- Currency codes
- Delivery methods
- Delivery terms
- Miscellaneous charges
- Dimensions (department, purpose, cost center)

Endpoint Value Mapping

Map the values that are used for the active action policy and for the particular endpoint such as item number, customer account number, vendor account number, and terms of delivery in the **Value Mapping** form. You can map values for the following elements:

- Trading partners (vendor and customer account numbers)
- Addresses (countries/regions, counties, states, and Postal/ZIP Codes)
- Inventory (item numbers, bar codes, item units, and warehouses)
- Currency codes
- Delivery methods
- Delivery terms
- Miscellaneous charges
- Dimensions (department, purpose, cost center)

Map values

You can map the following internal values from Microsoft Dynamics AX to external values in inbound or outbound XML documents by using the **Value Mapping** form.

For more information, see "Set up external codes for AIF" and "Currency code document value" in the Application and Business Processes Help.

Map vendor or customer numbers field values

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Select the endpoint for which you want to map field values, and then click **Action policies** to open the **Endpoint Action Policies** form.

3. Select the service action for which you want to map field values, and then click **Configure** to open the **Value Mapping** form.
4. On the **Partners** tab, select the trading partner (vendor or customer) for which you want to map field values.
5. In the **Document value** field, select the type of field value mapping:
   - **Not specified**
   - **Our**
   - **External code**
6. If you selected **External code** in step 5, select the external code in the **Customer code** or **Vendor code** field.

Map address field values

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Select the endpoint for which you want to map address field values, and then click **Action policies** to open the **Endpoint Action Policies** form.
3. Select the action for which you want to map address field values, and then click **Configure** to open the **Value Mapping** form.
4. On the **Addresses** tab, select the address type (countries, counties, states or ZIP/Postal Codes) for which you want to map field values.
5. In the **Document value** field, select the type of field value mapping:
   - **Not specified**
   - **Our**
   - **External code**
6. If you selected **External code** in step 5, specify the external code in the corresponding field.

Map terms of delivery, delivery methods, and misc. charges field values

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Select the endpoint for which you want to map field values, and then click **Action policies** to open the **Endpoint Action Policies** form.
3. Select the action for which you want to map field values, and then click **Configure** to open the **Value Mapping** form.
4. On the **Other base data** tab, select the field value (terms of delivery, delivery methods, or misc. charges) for which you want to map field values.

5. In the **Document value** field, select the type of field value mapping:
    - **Not specified**
    - **Our**
    - **External code**
6. If you selected **External code** in step 5, specify the external code in the corresponding field.

Map currency code field values

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Select the endpoint for which you want to map currency field values, and then click **Action policies** to open the **Endpoint Action Policies** form.
3. Select the action for which you want to map currency field values, and then click **Configure** to open the **Value Mapping** form.
4. On the **Other base data** tab, in the **Handling currency codes** section, in the **Document value** field, select the type of currency field value mapping:
    - **Not specified**
    - **Our**
    - **External code**
    - **ISO currency code**
5. If you selected **External code** in step 4, specify the external code in the **Currency code** field.

Map units and warehouse numbers field values

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Select the endpoint for which you want to map field values, and then click **Action policies** to open the **Endpoint Action Policies** form.
3. Select the action for which you want to map field values, and then click **Configure** to open the **Value Mapping** form.
4. On the **Items** tab, select the field value (units or warehouse number) for which you want to map field values.
5. In the **Document value** field, select the type of field value mapping:
    - **Not specified**
    - **Our**
    - **External code**
6. If you selected **External code** in step 5, specify the external code in the corresponding field.

Map item numbers field values

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Select the endpoint for which you want to map field values, and then click **Action policies** to open the **Endpoint Action Policies** form.
3. Select the action for which you want to map item number field values, and then click **Configure** to open the **Value Mapping** form.
4. On the **Items** tab, in the **Handling item numbers** section, in the **Document value** field, select the type of field value mapping:
   - **Not specified**
   - **Our**
   - **External code**
   - **External item number**
   - **Bar code**
   - **Company item**
5. If you selected **External code** in step 4, specify the external code in the **Item number code** field.

   If you selected **Bar code** in step 4, select the bar code type that the company uses in the **Bar code setup** field, and enter the bar code type that your trading partner uses in the **Value** field.

Configure data validation and defaulting

In Application Integration Framework (AIF), validation of data in an inbound XML document is usually performed by the `Ax`<*Table*> classes to ensure that referential integrity, number sequence, and business logic restrictions are enforced and to prevent incorrect data from being inserted into the application. If you disable data validation, the data from the inbound XML document is inserted into the application regardless of the data quality.

Defaulting of fields is performed by the `Ax`<*Table*> classes to set predefined values in the application data tables if the inbound document does not contain these values. Otherwise, the inbound document fails.

If you disable field defaults, data from the inbound XML document is inserted into the application regardless of the presence of required field values. This can result in some fields not containing values, the document failing, and an error being logged if any fields marked in the table as mandatory are empty.

Note:

Defaulting and validation are enabled by default.

How to disable data validation for inbound documents

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Select the endpoint for which you want to disable data validation, and then click **Action policies** to open the **Endpoint Action Policies** form.
3. Select the action for which you want to disable validation, and then click **Configure** to open the **Value Mapping** form.
4. In the **Setup** tab, clear the **Validate input** field.
5. To enable data validation, select the **Validate input** field again.

How to set disable defaulting for inbound documents

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Select the endpoint for which you want to disable defaulting, and then click **Action policies** to open the **Endpoint Action Policies** form.
3. Select the action for which you want to disable defaulting, and then click **Configure** to open the **Value Mapping** form.
4. In the **Setup** tab, clear the **Use defaulting** field.
5. To enable data validation, select the **Use defaulting** field again.

Configure document parameters

This section lists the setup parameters for some commonly-used documents to be sent or received by Application Integration Framework (AIF).

Document notes

Define the name of the note document type in the **Document management parameters** form. For more information, see "Document management parameters (form)" in the Application and Business Processes Help.

Inbound Sales Order document

- Define where inbound sales order documents are to be received in the **Accounts receivable parameters** form (**AIF** tab > (**Order type** field). For more information, see "Accounts receivable parameters (form)" in the Application and Business Processes Help.
- The inbound sales order is received in the **Sales order** form. For more information, see "Sales orders (form)" in the Application and Business Processes Help.
- The inbound sales order is received in the **Sales orders** form. For more information, see "Sales orders (form)" in the Application and Business Processes Help.

Inbound Purchase Invoice document

- Define the default register to receive the purchase invoice document in the **Accounts payable parameters** form (**AIF** tab > (**Journal name** field). For more information, see "Accounts payable parameters (form)" in the Application and Business Processes Help.
- Set up the manner in which duplicate invoices are processed in the **Accounts payable parameters** form (**Updates** tab > (**Check the invoice number used** field). For more information, see "Accounts payable parameters (form)" in the Application and Business Processes Help.

Inbound Packing Slip document

Define the default settings for the inbound packing slip document in the **Accounts receivable parameters** form (**AIF** tab > **Packing slip** field). For more information, see "Accounts receivable parameters (form)" in the Application and Business Processes Help.

Inbound Inventory Counting document

Set the default inventory counting journal for the inventory counting document in the **Inventory parameters** form (**AIF** tab > (**Counting** field). For more information, see "Inventory parameters (form)" in the Application and Business Processes Help.

Inbound Transfer Journal document

Set the default counting transfer journal for the inventory transfer document in the **Inventory parameters** form (**AIF** tab > (**Transfer** field). For more information, see "Inventory parameters (form)" in the Application and Business Processes Help.

Inbound Inventory Profit/Loss document

Set the default counting-profit-and-loss journal for the inventory profit-and-loss document in the **Inventory parameters** form (**AIF** tab > (**Profit/Loss** field). For more information, see "Inventory parameters (form)" in the Application and Business Processes Help.

Inbound Price/Discount agreement document (trade agreements)

Set the default counting price discount journal for the price discount document in the **Accounts payable parameters** form (**AIF** tab > **Price/Discount agreement** field). For more information, see "Accounts payable parameters (form)" in the Application and Business Processes Help.

Limit outbound documents

You can configure an endpoint to limit the number of documents or entity keys that are returned from a request in Application Integration Framework (AIF). Limiting the number of records returned reduces the size of the XML document that the Application Object Server (AOS) processes and improves navigation of the XML document or entity keys.

A scenario in which limiting documents may be useful is when the `read` and `find` actions are called. This is because those actions will typically be returning full documents and the number of records returned is unknown at request time leading to potentially large data sets being returned.

Limit the number of returned documents

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Select the endpoint for which you want to define return document limits, and then click **Action policies** to open the **Endpoint Action Policies** form.
3. Select the action for which you want to define return document limits, and then click **Configure** to open the **Value Mapping** form.
4. In the **Limit number of documents** field on the **Setup** tab, select **Yes** to limit the number of documents that are returned by a query. By default, the value in the (**Limitation type** field is **Default**, and the maximum number of documents returned is set to 1000.
5. To change the number of returned documents, select **Specified** in the (**Limitation type** field and enter the maximum number of documents returned by a request in the (**Max. number of documents** field.

Note:

If you send a request to AIF and anticipate the return of many large documents, you may want to first send a request using the `findKeys` action to return only all the entity keys (IDs) that match the criteria. After you receive a message with the entity keys, you can then manage processing of the data based on how many records the query returns.

Web services-based exchanges in AIF

Application Integration Framework (AIF) includes support for Windows Communication Foundation (WCF) Web services that allow Microsoft Dynamics AX to exchange data with external systems. Web services are programmatic interfaces that are made available to facilitate integration with external systems. In AIF, data exchanges using Web services are synchronous; that is, they do not require AIF queues and batch processing services to transfer information.

Web services do not require an adapter, nor do they require a channel to be configured. However, you must configure a Web site on the **Web sites** form, and you must configure the service and its operations so it can be consumed.

Microsoft Dynamics AX ships with a default endpoint but you can also set up your own endpoints. You can optionally configure a local endpoint, an endpoint, endpoint users, and endpoint operations. Additionally, you can configure pipeline components (including value lookups) and endpoint action data policies. For more information about configuring a data exchange with Web services, see Configure document exchanges with Web services in AIF.

When you use AIF to expose Web services to external trading partners and systems, you may allow them to create, read, update, delete, or query for records in the Microsoft Dynamics AX database.

Note:

It is unsafe to deploy AIF using Web services outside the intranet without installing additional middleware to ensure proper security. As installed with Microsoft Dynamics AX, AIF Web services are intended for intranet deployment only.

See Also

Configure document exchanges with Web services in AIF

Configure Web sites for document exchange

Security considerations for AIF

Security considerations for AIF Web services

Configure document exchanges with Web services in AIF

In Application Integration Framework (AIF), the following steps are used to set up and manage a document exchange using Web services:

- Install the prerequisites.
- Configure the exchange.
- Configure other AIF settings, if required.
- Maintain the integration.

Prerequisites

If you are setting up a data exchange that uses Web services, you must first perform these steps:

1. Install the Web services on the application integration server. This is the gateway computer that sends and receives AIF messages and communicates with one or more AOSs. For more information, see "Install AIF Web Services" in the Microsoft Dynamics AX Installation Guide.
2. Use the **Web sites** form to configure the AIF Web site that hosts the Web services. For more information, see Configure Web sites for document exchange

Configure the exchange

To quickly configure a document exchange, follow the steps in the minimal configuration. If you want to customize or modify the document exchange, follow the steps in the extended configuration.

Minimal configuration

To set up a document exchange with the minimal configuration, perform the following steps:

1. Generate the Web services. In this step, you publish the Web services for external consumption. See Configure services. This step is **required**.

   This step is the only task that you need to do to enable the AIF Web services. AIF ships with a default endpoint that is used to send or receive data so no endpoint is necessary in the message header. By default, all service operations (for services that have been generated) are enabled for the default endpoint.

   Note:

   If the default endpoint is used and records are created in Microsoft Dynamics AX, those records will be associated with the default company of the user that submitted the request.

2. Grant permissions to the Web service. See Grant permissions to a service. This step is **required**.

   After you have completed these steps, you can exchange messages with the Web services that you have enabled.

3. Create the XML message. The message you send to AIF through the Web service will vary in format depending on which service operation you are calling. If you want to create a sales order, then the XML message will contain the `create` action and the data required to create a sales order in Microsoft Dynamics AX. If you want to request a sales order, then the XML message will contain the `read` action and the ID of the sales order that you want to retrieve. For more information about messages, see "AIF Messages" in the Microsoft Dynamics AX SDK Help.

4. Get the Web service URL and call the service. Follow these steps to get the service URL:
   a. Click **Start** > **Administrative tools** > **Internet Information Services (IIS) Manager**.
   b. Expand the **Default Web Site** node or the Web site node that contains the AIF Web services virtual directory.
   c. Navigate to the virtual directory **MicrosoftDynamicsAXAif50**. The services appear in the pane on the right side of the screen and have a file name extension of .svc.
   d. Right-click the .svc file and select **Browse**. An Internet Explorer browser window appears with the service name at the top. The URL to which Internet Explorer points is the URL of the service, for example,

      `http://localhost/MicrosoftDynamicsAXAif50/salesorderservice.svc` (you will need the fully-qualified URL without the `localhost` path in order to successfully call the service).

You must create a program to call the appropriate Web service and pass the message to the service.

Extended configuration

The extended configuration provides more flexibility and options for configuring the AIF components that make up a document exchange. This configuration allows you to create your own endpoints, configure action policies for those endpoints, and specify endpoint action data policies. To set up a document exchange with the extended configuration, perform the following steps:

1. Generate the Web services. This step publishes the Web services for external consumption. See Configure services. This step is **required**.
2. Grant permissions to the Web service. See Grant permissions to a service. This step is **required**.
3. Create a local endpoint. See Create and configure local endpoints. This step is **required** if you are going to create an endpoint in the following step.
4. Create and configure an endpoint for the external system that will consume the Web services. This is only necessary if you do not use the default endpoint that ships in AIF. See Creating and configuring endpoints. This step is **required** if you are not going to use the default endpoint.
5. Configure the endpoint action policy to associate the desired actions with the endpoint. See Configure endpoint action policies. This step is **required** if you are not going to use the default endpoint.
6. Configure data policies. See Configure endpoint action data policies. This step is **required** if you are not going to use the default endpoint.
7. Create the XML message. The message you send to AIF through the Web service will vary in format depending on which action you are calling. If you want to create a sales order, then the XML message will contain the `create` action and the data required to create a sales order in Microsoft Dynamics AX. If you want to request a sales order, then the XML message will contain the `read` action and the ID of the sales order that you want to retrieve. For more information about messages, see "AIF Messages" in the Microsoft Dynamics AX SDK Help.

8. Get the Web service URL and call service. Follow these steps to get the service URL:
    a. Click **Start** > **Administrative tools** > **Internet Information Services (IIS) Manager**.
    b. Expand the **Default Web Site** node or the Web site node that contains the AIF Web services virtual directory.
    c. Navigate to the virtual directory **MicrosoftDynamicsAXAif50**. The services appear in the pane on the right side of the screen and have a file name extension of .svc.
    d. Right-click the .svc file and select **Browse**. An Internet Explorer browser window appears with the service name at the top. The URL to which Internet Explorer points is the URL of the service, for example,
    `http://localhost/MicrosoftDynamicsAXAif50/salesorderservice.svc`.

You must create a program to call the appropriate Web service and pass the message to the service.

Other AIF settings

The following forms can be used when configuring a data exchange regardless of whether you are using the minimal configuration or the extended configuration:

- Use the **Global settings** form to configure global defaults for configuring adapters, actions, resource locks for batch processing, and schema validation, as well as the default encoding format for documents. For more information, see Configure global settings for document exchange.
- Use the **Pipeline components** form to configure optional document transformations, including XSLT style sheet mapping or value substitutions. For more information, see Creating and configuring a pipeline.
- Use the **Value Mapping** form to set up optional predefined value mapping that is available for certain documents. For more information, see About value mapping.

Maintain integration with external software systems

After you have configured a document exchange using AIF, you will need to maintain that integration and possibly troubleshoot any errors that occur. This may include the following tasks:

- Checking error logs and message queues to monitor traffic.
- Stopping and restarting the framework when necessary.
- Reconfiguring the channel and endpoint if conditions change.
- Viewing the document history and deleting old messages.

For more information, see Manage document exchanges in AIF.

See Also

Web services-based exchanges in AIF

Configure Web sites for document exchange

After you have installed the Application Integration Framework (AIF) Web services on the application integration gateway server, you must create a Web site for use by AIF. This Web site points to the shared content directory that was created when you installed the Web services. This directory is where all the generated service artifacts will be copied for external use when you configure Web services.

For more information about installing an application integration server, see "Install AIF Web services" in the Microsoft Dynamics AX Installation Guide.

Add a Web site for use by AIF

After installing AIF Web services, you must add a Web site. Adding a Web site specifies to the Application Object Server (AOS) the content directory where the Web services are stored. This directory links Microsoft Dynamics AX with the Web services configuration that is created when you install AIF Web services.

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Web sites**.
2. Press CTRL+N to create a new Web site or select one of the Web sites from the list.
3. Enter a descriptive name for this Web site in the **Name** field.

   The name can contain special characters and blanks and can be up to 50 characters long.
4. In the **Virtual directory share path** field, enter the path to the content directory that contains the Web service components that are generated.

   Note:

   By default, the content directory is located in the <*Microsoft Dynamics AX installation path*>*AifWebServices* (C:\Program Files\Microsoft Dynamics AX\50\AifWebServices, for example). The network share name for this directory is **MicrosoftDynamicsAXAif50**.
5. To search for the content directory path, on the **General** tab, click **Browse**.
6. In the **Description** field, enter a description of the Web site.

Validate the Web site

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Web sites**.
2. On the **Overview** tab, select a Web site from the list.
3. To check that each selected content directory path exists and that the current AOS user account has read, write, and delete permissions on the directory, click **Validate**. The validation results appear in an **Infolog** window.

See Also

Configure document exchanges with Web services in AIF

Configure global settings for document exchange

In the **Integration Framework global settings** form, you can set global defaults for AIF components including:

- Actions
- Resource locks
- Schema validation
- Default encoding format for documents
- Response cache messages

Set the maximum resource locking interval

The default value for the resource locking interval is 30 minutes. The locking interval only needs to be modified if you have multiple Application Object Server (AOS) instances running. The maximum resource locking interval determines how often the various services lock resources when processing messages. This setting is used by the gateway service to lock channels for inbound and outbound processing, by the inbound processing service to lock channels for inbound processing, and by the outbound processing service to lock endpoints for outbound processing.

An example of resource locking is if you have two AOS instances, AOS1 and AOS2. When AOS1 retrieves messages from the channel, it locks the channel so that AOS2 cannot access those messages and send them out twice. If AOS1 fails, that channel remains locked for the time specified in the global settings. If AOS1 remains unavailable, when the lock expires after 30 minutes, AOS2 can access those messages and deliver them. It is recommended that you monitor the volume of exchanges for your particular system and then set the maximum locking interval.

- If the locking interval is too short, the locks expire faster. When a lock expires and another AOS is available, the same channel or endpoint is processed again. This can lead to wasted resources because the channel or endpoint is processed continuously when no messages are present.
- If the locking interval is too long, it takes the system longer to recover in the event of a system failure.

You should change the maximum locking interval after monitoring the volume of exchanges for your particular system.

For adapter-based exchanges, AIF implements a scheme for locking endpoints and channels to guarantee that messages are processed in a particular order. After one of the AIF services (`AifInboundProcessingService`, `AifOutboundProcessingService`, `GatewaySendService`, and `GatewayReceiveService`) begins processing messages related to a channel or an endpoint, a lock is set for that resource. If the service stops processing the resource before all messages have been transferred, the lock on the resource expires after the maximum locking interval, and another service can begin processing that resource. For more information about the AIF batch services, see Start and stop the asynchronous AIF services.

The default value for the maximum resource locking interval is 30 minutes. If your installation only runs one instance of the AIF services, and those services are interrupted while processing channels or endpoints, the maximum time before the services can begin processing again is 30 minutes.

However, if you are exchanging documents using more than one channel or more than one endpoint and you use multiple AOS instances installed on multiple computers, you may need to set the maximum locking interval for channels and endpoints.

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Global settings**.
2. Enter the time in minutes in the **Maximum resource locking interval (minutes)** field.
3. Press CTRL+S to save.

Set the default encoding format

The default encoding format specifies the encoding of the documents exchanged with an endpoint. When a new endpoint is created, the encoding format for that endpoint defaults to the specified value in the global settings.

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Global settings**.
2. In the **Default encoding format** field, select a supported encoding format from the list to be used as the default on the **Endpoints** form.
3. Press CTRL+S to save.

Set up validation for outbound documents

If you select this option, the XML format for all outbound messages will be validated against the document class schema. For more information about document schemas, see "Document Schema Overview" in the Microsoft Dynamics AX SDK Help.

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Global settings**.
2. Select the **Validate outbound schema** field.
3. Press CTRL+S to save.

Note:

Choosing to validate the schema for every outbound document may negatively impact performance.

Set the response cache lifetime

The response cache lifetime specifies the time in hours that message responses are cached. This setting is only used when you set the `EnableIdempotence` property for a particular service operation to true in the AOT. If a service operation has idempotency enabled, the response message for any message that calls that operation is cached. If a duplicate message is received, AIF generates an error and sends the original response back as part of the generated error.

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Global settings**.

2. In the **Response cache lifetime (hours)** field, enter the number of hours that response messages must be cached.
3. Press CTRL+S to save.

Configure services

Application Integration Framework (AIF) includes a number of services for integrating your system in common business processes. In order to enable an exchange using a service, you must first enable the service. For external systems to consume the service, you must generate it and make it available to external callers.

Enable and generate the service

Use the following steps to enable and generate a service.

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Services**. The first time the form opens, you must click **Refresh** to load the services. There may be a delay while the services load.
2. Select the **Enabled** check box for each service that you want to use in an exchange.
3. Click **Generate** to generate the service artifacts for each enabled service. The generation process copies the service and all the generated files to <*Microsoft Dynamics AX installation path*>*Application**Appl**DynamicsAx**ServiceGeneration* folder. If there are Web sites configured in the **Web sites** form, the service is copied to the content directory specified for each Web site (by default this directory is <*Microsoft Dynamics AX installation path*>*AifWebservices*). This enables external callers to consume the service and is only necessary for document exchanges using Web services.

The service artifacts that are generated include the following:

- The .svc file
- The schemas for the service operation parameters and return values
- The web.config file

Test the service

After you have generated a service, you can test that it is functioning in Internet Information Services (IIS) by browsing to the service.

1. Click **Start** > **Administrative tools** > **Internet Information Services (IIS) Manager**.
2. Expand the **Default Web Site** node or the Web site node that contains the AIF Web services virtual directory.
3. Navigate to the virtual directory **MicrosoftDynamicsAXAif50**. The services appear in the pane on the right side of the screen and have a file name extension of .svc.
4. Right-click the .svc file and select **Browse**. An Internet Explorer browser window appears with the service name at the top. If the Web service is correctly configured in IIS, you will see a link to the Web service WSDL. If the Web service is not correctly configured in IIS, you will see a service error in the browser.

If you receive HTTP Error 404.3

If you have installed AIF Web services, done the necessary configuration to expose a service as a Web Service, and you receive an HTTP Error 404.3 when you try to call the service, you may need to register Windows Communication Foundation (WCF) with IIS.

If IIS is installed after the .NET Framework is installed, you must register WCF with IIS and ASP.NET on the computer where Web services are installed by following these steps.

Operating system	Steps
Windows Server 2003 Windows Server 2008	Use the `ServiceModelReg` tool by entering the following command at a command prompt: `ServiceModelReg.exe /i /x`. This command line tool is located in the WCF directory, for example, `C:\Windows\Microsoft.NET\Framework\v3.0\Windows Communication Foundation`.

Refresh the services

If you make any changes to a service, or if you add a custom service in the AOT, you must refresh the services for those changes to be reflected in AIF. To refresh the services, click **Refresh**.

The refresh process scans the AOT and refreshes the list of services reflecting any changes to services in the AOT **Services** node. There may be a delay while the services are refreshed.

Configure the Web service

The AIF services support WCF and WCF-based Web services, so after you have generated a service, you can configure the WCF properties. To modify the WCF configuration, click **Configure**.

If the Windows SDK is installed, the WCF Service Configuration Editor (`SvcConfigEditor.exe`) opens and enables you to modify the configuration. If the SDK is not installed, the configuration opens in Notepad.

View service operations

Service operations define what actions you can perform on a service. Service operations are implemented as methods on the service.

To view the service operations available for an AIF service, click **Service operations**. In the **Service Operations** form, you can view the following items:

- The service operations (methods)
- The parameters and return value for each operation
- The schema for each parameter and return value

Service naming conventions

The **AIF Services** form contains all the services that are included with Microsoft Dynamics AX. The following table illustrates the naming conventions used for the service name and the external name (the service name exposed outside of Microsoft Dynamics AX).

Tab	Naming convention	Example
Service name	*<Prefix>* + *<Document Name>* + "Service"	`CustCustomerService`
External name	*<Document Name>* + "Service"	`CustomerService`

See Also

Configure Web sites for document exchange

Grant permissions to a service

After you have enabled and generated services in Application Integration Framework (AIF), you must assign users access to them through security key permissions. The services that ship with Microsoft Dynamics AX are already assigned to a security key and each service has its own security key. The service security key has a parent key that is based on the functional area that the service is part of.

For example, the purchase requisition service (`PurchPurchReqService`) is secured by the `PurchPurchReqService` key. The parent key is the `VendServices` key. The full security key path is `Accounts Payable\Services\PurchPurchReqService`.

View the security keys

The service security keys are part of the standard security in Microsoft Dynamics AX. To view the security keys for AIF services, follow these steps:

1. Click **Administration** > **Setup** > **Security** > **User group permissions** and select the **Permissions** tab.
2. Expand a node in the tree, for example, **Accounts payable**.
3. Navigate to the **Services** node and expand it. You'll see the AIF service keys listed underneath, for example, **PurchPurchReqService**

Find the security key to which a service is assigned

The service security key is stored as a property in the AOT. To find out which security key a service is associated with, follow these steps:

1. Open the **AOT** and navigate to the **Services** node.
2. Expand the **Services** node and navigate to the service.
3. Right-click the service and select **Properties**. The **SecurityKey** property contains the name of the security key with which the service is associated.

To give users access to services, follow the standard method of assigning user permissions to security keys. For more information, see "Manage security permissions for user groups and domain combinations" in the System Setup Help.

See Also

Configure document exchanges with Web services in AIF

Configure services

Create and configure local endpoints

The local endpoint refers to your Microsoft Dynamics AX installation. A local endpoint is the origination for sent messages and the destination for received messages. There may be more than one local endpoint, depending on how many companies are configured for your Microsoft Dynamics AX installation.

The name of the local endpoint identifies you to external systems. Therefore, we recommend that the name for the local endpoint be representative of the Microsoft Dynamics AX company name that participates in the exchange.

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Local endpoints**.
2. Press CTRL+N to create a new local endpoint record.
3. Select a company from the **Company** field.
4. Enter a name for the endpoint in the **Local endpoint** field.

   The name for the local endpoint should be representative of the Microsoft Dynamics AX company name that participates in the exchange.
5. Press CTRL+S to save the data.

See Also

Configure document exchanges with adapters in AIF

Configure document exchanges with Web services in AIF

Creating and configuring actions

In Application Integration Framework (AIF), there are two types of actions:

- ServiceOperation
- SendXML

A ServiceOperation action defines an action that is available from a service. A SendXML action defines an action that is independent of any service or class and is used in X++ to send XML (outbound).

The **Actions** form displays all of the actions that are available.

ServiceOperation Actions

The services that ship with Microsoft Dynamics AX come with one or more of the actions in the following table. These actions are ServiceOperation actions and are read-only in the **Actions** form.

Service operation	Description
`create`	Accepts XML data, writes a record to the database, and returns the ID of the new record.
`delete`	Accepts ID values and deletes the associated records.
`find`	Accepts query criteria and returns the matching data.
`findKeys`	Accepts query criteria and returns IDs of the matching data.
`read`	Accepts a list of IDs and returns the associated data.
`update`	Accepts a list of IDs and data and updates the associated records in the database.

Each service does not implement all of the available actions. Rather, each service implements actions based on the requirements of the business process that the service supports. Follow these steps for more information about the service operations that are supported by a service:

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Services**
2. Select a service and click > **Service Operations**. The **Service Operations** form lists all of the available operations.
3. To view the parameters for a service operation, select the operation and click **Parameter schema**.

SendXML action type

You can use the SendXML action type when you want to create an action that is not a service operation. In the **Actions** form you can add, modify, or delete a SendXML action. For more information about creating and configuring a SendXML action, see Create and configure a SendXML action.

Security note for inbound documents

Certain actions, such as creating exchange rates, cause data to be written directly to the Microsoft Dynamics AX database without manual end-user verification. When configuring endpoints and creating new actions, be careful to restrict access to trusted and reliable partners and applications.

Create and configure a SendXML action

The **Actions** form displays all of the actions that are available. ServiceOperation actions are defined in the AOT as part of a service. Therefore, they cannot be modified in the **Actions** form. However, the **Actions** form is used to add, modify, or delete SendXML actions.

In Application Integration Framework (AIF), there are two types of actions: ServiceOperation and SendXML. A ServiceOperation action defines an action that is available from a service. A SendXML action defines an action that is independent of any service or class and is used in X++ to send XML (outbound).

Add a SendXML action

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Actions**.
2. Press CTRL+N to add a new record and click the **General** tab.
3. In the **Action** field, type a unique action identifier.
4. In the **Name** field, type the action name.
5. Select the **Enabled** field to enable the action for use.
6. In the **External identifier** field, type an external identifier name. This is the externally-facing name of the action.
7. In the **Description** field, type a description. The action type defaults to **SendXml**. The action can now be used like other AIF actions and associated with an endpoint, configured with pipeline components, and so on.

Configure a SendXML action

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Actions**.
2. Click the **General** tab. The fields that you can edit are:
   - **Name**
   - **Enabled**
   - **External identifier**
   - **Description**

Delete a SendXML action

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Actions**.
2. Select an action and click ALT+F9.

Note:

You should ensure that an action is not used in an action policy for any endpoints before deleting the action.

See Also

Creating and configuring actions

Creating and configuring endpoints

Using Application Integration Framework (AIF), you enable document exchanges between endpoints and the local endpoint. The local endpoint refers to your Microsoft Dynamics AX installation. A local endpoint is the origination for sent messages and the destination for received messages.

An endpoint represents the external system that participates in data exchange. An endpoint is the destination for a sent document and the source of a received document.

For more information about creating and configuring endpoints see Create an endpoint and Configure an endpoint

Default endpoint

Microsoft Dynamics AX ships with an endpoint called the default endpoint. Therefore, it is not necessary to create an endpoint for data exchange. Creating and configuring a specific endpoint is optional, and if a message does not specify an endpoint, AIF implicitly uses the default endpoint.

If you disable the default endpoint in Microsoft Dynamics AX, then you must create a specific endpoint and all requests must reference that endpoint or the message will be rejected.

Specific endpoints

You may want to create endpoints for data exchange for a variety of reasons. Endpoints are useful in business-to-business scenarios where you want put restrictions on a data exchange. With endpoints you can:

- Set constraints on the endpoint. Constraints specify that messages can only be sent to or received from specific customers, vendors, or warehouses.
- Implement a higher level of security by assigning specific users or groups to an endpoint. This ensures that only certain users can send messages to AIF.
- Assign actions to an endpoint to restrict the types of data exchanges in which the endpoint can participate.
- Set data policies on an endpoint to restrict the data that can be retrieved or updated.
- Implement custom configuration requirements like XSLT transformations, value mapping, and custom pipeline components.
- Customize the message logging for each action associated with an endpoint.

Prerequisites

Before you can create an endpoint, the following must already exist and be configured:

- A local endpoint.
- A named service operation for the exchange, for example, `CustCustomerService.read`. For more information, see Creating and configuring actions.

- Microsoft Dynamics AX users that will be associated with the endpoint.
- An outbound channel must be already be defined if needed.

Endpoint information

When you configure an endpoint, you enter information about the following:

- Identifying information for the endpoint, including a unique identifier, a friendly name, the active status, error handling information, the associated channel, and the Microsoft Dynamics AX company identification.
- Constraints on the endpoint that restrict document exchange by defining valid Microsoft Dynamics AX customers, vendors, or warehouses.
- Microsoft Dynamics AX users and trusted intermediaries that are allowed to submit documents for the exchange.
- Endpoint action policies that relate actions to the endpoint.
- Pipeline components for any optional document transformations for the action related to the endpoint.
- Endpoint action data policies (also known as data policies) that define which fields in a document are allowed or required to participate in the exchange.
- Document configuration options, including value mapping. For more information, see About value mapping and "Set up external codes for AIF" in Applications and Business Processes.

See Also

Create and configure local endpoints

Creating and configuring actions

Create an endpoint

Configure an endpoint

Configure endpoint action policies

Configure endpoint action data policies

Configure an endpoint

When you configure an endpoint, you enter information in the **Endpoints** form about the following:

- Identifying information for the endpoint, including a unique identifier, a friendly name, the active status, error handling information, the intercompany status, local endpoint, Microsoft Dynamics AX company identification for an intercompany exchange, outbound channel information, and encoding format for the transfer.
- Constraints on the endpoint that restrict document exchange by defining valid Microsoft Dynamics AX customers, vendors, or warehouses.

- Microsoft Dynamics AX users and trusted intermediaries that are allowed to submit documents for the exchange.
- Endpoint action policies that relate actions to the endpoint.
- Pipeline components for an action related to the endpoint.
- Endpoint action data policies that define which fields in a document are allowed or required for the exchange.

Available tabs

In the **Endpoints** form, user interface tabs are available depending on what type of endpoint you select in the **Overview** tab.

Endpoint type	Tabs enabled
Default endpoint	• **Overview** • **General**
Specific endpoint	• **Overview** • **General** • **Constraints** • **Users**
Specific intercompany endpoint	• **Overview** • **General** • **Constraints**

Default endpoint

Microsoft Dynamics AX ships with a default endpoint that can be used to enable data exchanges as soon as services are generated. The default endpoint has minimal configuration options, and you cannot add any constraints or users to the default endpoint. By default, all actions are enabled for the default endpoint but you must still enable the service actions on the **AIF Services** form.

You can change the following options on the default endpoint:

- In the **Endpoints** form, you can update the **Active** field and the **Propagate errors** field. For more information about these fields, see Create an endpoint.
- In the **Endpoint Action Policies** form, you can enable and configure actions.
- In the **Endpoint action data policies** form, you can define which fields are allowed or required in the data exchange.
- In the **Pipeline components** form, you can define data transformations for inbound or outbound exchanges.

- In the **Parameter Schemas** form, you can view the schema of the action parameters and the return value and optionally save them to an .xsd file.
- In the **Value Mapping** form, you can configure value mapping.

Configure endpoint identification

1. Click **Basic** > **Setup** > **Application Integration Framework**.> **Endpoints**.
2. Press CTRL+N to create a new endpoint, or select an existing endpoint from the list to be modified.
3. Enter the information for the endpoint on the **General** tab, including the unique endpoint identification information, the friendly name for the endpoint, the local endpoint ID, and the default encoding format.
4. Select **Propagate errors** to return detailed error messages to the endpoint.

   Note:

   By default, Microsoft Dynamics AX logs detailed errors and sends a generic error back to the endpoint. Enabling this field will send the detailed error to the endpoint. Only select this field if it is acceptable to send detailed error information to an endpoint.
5. For intercompany transfers, select **Intercompany organization** if the endpoint is a company within your Microsoft Dynamics AX installation, and select a company in the **Company** field.
6. Do not select **Intercompany organization** unless the endpoint is to be used in an intercompany transfer. For more information about these transfers, see "Manage intercompany sales orders" or "Manage intercompany purchase orders" in the Application and Business Processes Help.
7. Selecting an outbound channel is not required for all exchanges. However, if your Microsoft Dynamics AX installation is sending messages to this endpoint (for example, if you are configuring an outbound-only exchange), you must select a channel in the **Outbound channel ID** field.
8. In the **Local endpoint ID** field, select the local endpoint that participates in exchanges with this endpoint.

   Note:

   There may be more than one local endpoint configured for your Microsoft Dynamics AX installation. If that is the case, then make sure to select the correct local endpoint to participate in the exchange with the endpoint that you are configuring.
9. View the pre-populated setting for encoding format in the **Default encoding format** field. This setting defaults to the value set on the **Integration Framework global settings** form.

Configure constraints on an endpoint and activate the endpoint

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Press CTRL+N to create a new endpoint, or select an existing endpoint from the list to be modified.

   Note:

   You cannot add constraints to the default endpoint so the **Constraints** tab will not be enabled if the default endpoint is selected.
3. On the **Constraints** tab, select **No constraints** to clear the form and allow data to be exchanged regardless of any association. The **No constraints** check box becomes unavailable. However, if you add constraints later, the check box clears itself.
4. Press CTRL+N to create a new constraint.
5. Choose the **Constraint type** (**Vendor**, **Customer**, or **Warehouse**).
6. Select a **Constraint ID** from the list. The **Name** field is completed when you select the **Constraint ID**.
7. On the **Overview** tab, select **Active** to activate the endpoint.

   Messages flow through the framework from the local endpoint to and from any active endpoints.

Configure users and trusted intermediaries for an endpoint

You must enter information for at least one endpoint user or trusted intermediary who is authorized to initiate transactions for the endpoint on the **Users** tab.

Notes:

- When configuring users on an endpoint, remember that these Microsoft Dynamics AX users may represent outside interests and must have permissions set appropriately. For more information about configuring Microsoft Dynamics AX users, see "Setting up and maintaining security" in the Microsoft Dynamics AX Installation Guide and the following topics in the System and Application Setup Help: "Manage security permissions for user groups and domain combinations," "Manage user groups," and "Manage users."
- You must also set the appropriate security keys and record-level security for any users that are granted access to Microsoft Dynamics AX through Application Integration Framework (AIF), to help prevent unauthorized data access. For more information, see "Manage record level security" in the Application and Business Processes Help.
- Certain actions cause data to be written directly to the Microsoft Dynamics AX database without manual end-user verification (for example, creating exchange rates). When configuring endpoints and creating new actions, be especially careful to restrict access to trusted and reliable partners or applications.

Configure an endpoint user

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Press CTRL+N to create a new endpoint, or select an existing endpoint from the list to be modified.

   Note:

   You cannot add users or user groups to the default endpoint so the **Users** tab will not be enabled if the default endpoint is selected.
3. Click the **Users** tab.
4. Under **Endpoint users**, in the **User type** field, select either **User** or **User group**.
5. In the **Application user or group** field, select a valid Microsoft Dynamics AX user or user group name. The **Name** field is filled in automatically.

   For more information about Microsoft Dynamics AX users and user groups, see "Setting up and maintaining security" in the System and Application Setup Help.

Configure a trusted intermediary

Trusted intermediaries are middleware applications that reside between external endpoints and AIF. That is, they are Microsoft Dynamics AX users (or user groups) that are authorized to submit inbound requests on behalf of the endpoint. A trusted intermediary prevents an unauthorized user from accessing AIF and is typically used in a business-to-business data exchange scenario. For more information about trusted intermediaries, see Security considerations for AIF.

For more information about Microsoft Dynamics AX users and user groups, see "Setting up and maintaining security" in the System and Application Setup Help.

1. Under **Trusted intermediaries**, select **Use trusted intermediary** to enable a trusted intermediary for exchanges with this endpoint.

   Note:

   If the **Use trusted intermediary** box is checked, there must be at least one entry in the **Trusted intermediaries** grid.
2. Under **Trusted intermediaries**, in the **User type** field, also select either **User** or **User group**.
3. In the **Application user or group** field, select a valid Microsoft Dynamics AX user or user group name. The **Name** field is filled in automatically.

Configure action policies, data policies, and pipeline components for an endpoint

1. After you have completed the preceding steps, click **Action policies** to configure actions on the endpoint.

   For more information about action policies, see Configure endpoint action policies.
2. Press CTRL+S to save the action policy.

3. From the **Endpoint Action Policies** form, select an action, press CTRL+S, and click **Data Policies** to enter information about which fields are required and which are optional in the document to be exchanged.
4. Click **Configure** to perform document-specific configuration, including value mapping, for the document exchange.

   Value mapping is translation of field data values based on business rules, for example, translating internal item numbers to vendor-specific item numbers or industry standard numbers depending on the trading partner. For more information, see Configure endpoint action data policies.
5. Click **Inbound Pipeline** or **Outbound Pipeline** to configure the pipeline components for any custom transformations performed on the document. For more information, see Creating and configuring a pipeline.
6. Click **Parameter schema** to see a list of parameters and the return value for the selected action. On the **Parameter Schemas** form, click **View schema** to view the XML schema for the selected parameter or return value. On the **Schema** form, you can click **Save as** to save the schema as an .xsd file.

Give the endpoint user access to Business Connector when using Web services

When you configure an endpoint for Web services, you must configure an endpoint user and/or trusted intermediary, as outlined above. Next, you must give that Microsoft Dynamics AX user or user group access to the Business Connector.

1. Click **Administration** > **Setup** > **User groups** and select the user group for the endpoint, or the user group that contains the user for the endpoint.
2. Click **Permissions**.
3. On the **Permissions** tab, select **Business Connector** and then select **Full control**.
4. Click **Cascade**.

Disable an endpoint

To disable an endpoint, follow these steps.

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Select an endpoint in the grid and clear the **Active** field.

Important:

For adapter-based exchanges, outbound messages are processed completely even when the endpoint is disabled during processing and a response may still be sent. To ensure that no data is sent from AIF when disabling an endpoint, first disable the batch processing jobs. Be sure that there are no outbound messages in the queue, disable the endpoint, and then restart the batch processing jobs.

See Also

Create and configure local endpoints

Creating and configuring actions

Configure endpoint action policies

Configure endpoint action data policies

Creating and configuring a pipeline

Configure global settings for document exchange

Create an endpoint

Before you can create an endpoint, the following must already exist and be configured:

- A local endpoint.
- An enabled service operation for the exchange, for example, `CustCustomerService.read`. For more information, see Creating and configuring actions.
- Microsoft Dynamics AX users that will be associated with the endpoint.
- An outbound channel must be already be defined if needed.

Create the endpoint

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Press CTRL+N to create a new endpoint.
3. In the **Endpoint ID** field, type a unique identifier for the endpoint.
4. In the **Name** field, type a friendly name.

5. Select **Propagate errors** to return detailed error messages to the endpoint.

   Note:

   By default, Microsoft Dynamics AX logs detailed errors and sends a generic error back to the endpoint. Enabling this field will send the detailed error to the endpoint and could be a potential security risk. Only select this field if it is acceptable to send detailed error information to an endpoint.

6. Do not select **Intercompany organization** unless the endpoint is to be used in an intercompany transfer. For more information about these transfers, see "Manage intercompany sales orders" or "Manage intercompany purchase orders" in the Application and Business Processes Help.
7. If you select **Intercompany organization**, then you must select a company in the **Company** field.
8. In the **Outbound channel ID** field, select an outbound channel. You must select a channel if the endpoint will be used in an adapter-based exchange and data is to be sent outbound from Microsoft Dynamics AX to the endpoint. If the endpoint is participating only in Web services-based exchanges, the outbound channel is not necessary.
9. In the **Local endpoint ID** field, select the identifier for the local endpoint (your system) that participates in the exchange with the endpoint that you are configuring.
10. In the **Default encoding format** field, select the encoding format for this endpoint.
11. On the **Constraints** tab, enter the data constraints for the endpoint to restrict the data that can be processed by the endpoint. To allow data to be exchanged regardless of any associations, click **No constraints**. For more information, see Configure an endpoint.

    After selecting, the **No constraints** check box becomes unavailable. However, if you add constraints later, the check box clears itself.

12. On the **Overview** tab, select **Active** to enable the endpoint to participate in data exchanges.
13. On the **Users** tab, enter information to restrict users that are authorized to initiate transactions for the endpoint. In the **User type** field, select either **User** or **User group**.

    You can also designate trusted intermediaries on the **Users** tab. Trusted intermediaries are middleware applications that reside between external endpoints and Application Integration Framework (AIF), that is, they are Microsoft Dynamics AX users (or user groups) that are authorized to submit inbound requests on behalf of the endpoint. For more information about trusted intermediaries, see Security considerations for AIF.

    Notes:

    - When configuring users on an endpoint, keep in mind that these Microsoft Dynamics AX users may represent outside interests and must have permissions set appropriately. For more information about configuring Microsoft Dynamics AX users, see "Setting up and maintaining security" in the Microsoft Dynamics AX Installation Guide and the following topics in the System and Application Setup Help: "Manage security permissions for user groups and domain combinations," "Manage user groups," and "Manage users."

- You must also set the appropriate security keys and record-level security for any users that are granted access to Microsoft Dynamics AX through AIF to help prevent unauthorized data access. For more information, see "Manage record level security" in the Application and Business Processes Help. Certain actions cause data to be written directly to the Microsoft Dynamics AX database without manual end-user verification (for example, creating exchange rates). When configuring endpoints and creating new actions, be especially careful to restrict access to trusted and reliable partners and applications.

14. Click **Action policies** to configure actions on the endpoint with the **Endpoint Action Policies** form. Examples of actions include the service operations `read` and `create`. For information on setting up action policies, see Configure endpoint action policies.
15. From the **Endpoint Action Policies** form you can select an action and click **Data Policies** to enter the data policy, that is, information about which fields are required and which are optional in the document to be exchanged. For details on setting up data policies, see Configure endpoint action data policies.
16. From the **Endpoint Action Policies** form, you can click **Configure** to perform document-specific configuration, including value mapping. Value mapping is the translation of field data values based on business rules, for example, translating internal item numbers to vendor-specific item numbers or industry standard numbers depending on the trading partner. For more information, see Configure endpoint action data policies.

Enabled fields

The fields that are enabled for an endpoint depend on the type of endpoint that you are configuring as shown in the following table.

Field	Default endpoint	Standard endpoint	Intercompany endpoint
Endpoint ID	No	Yes (when adding new endpoint)	Yes (when adding new endpoint)
Name	No	Yes	Yes
Active	Yes	Yes	Yes
Propagate errors	Yes	Yes	Yes
Intercompany organization	No	Yes	Yes
Company	No	Yes	Yes
Outbound channel ID	No	Yes	No
Local endpoint ID	No	Yes	No
Default encoding format	No	Yes	No

Troubleshooting Trusted Intermediary

If you receive an error on an inbound message and it was submitted by a trusted intermediary, it may be due to the fact that the submitting user of the message and the user specified in the message `SourceEndpointUser` element are not in the same Microsoft Dynamics AX domain.

The trusted intermediary is a valid Microsoft Dynamics AX user that is allowed to submit AIF messages. Typically, the trusted intermediary does not have access to the AIF services. The source endpoint user is a valid Microsoft Dynamics AX user that has access to the AIF services. If the user submitting the message (the trusted intermediary) is in a different domain than the source endpoint user, you may receive an error.

To resolve this problem, give the source endpoint user open domain access permissions. To locate the security key for these permissions, use the following steps.

1. Open **Administration** > **Setup** > **User groups**.
2. Click **Permissions**.
3. On the click **Permissions** tab, expand the **Administration** node and select **Open domain access**.

   Note:

   This refers to Microsoft Dynamics AX domains and not Active Directory domains. Microsoft Dynamics AX domains are defined in **Administration** > **Setup** > **Domains**.

For more information, see "Manage security permissions for user groups and domain combinations" in and "Manage domains" in the System and Application Setup Help.

See Also

Configure an endpoint

Configure endpoint action policies

Configure endpoint action data policies

Creating and configuring a pipeline

Configure endpoint action policies

To enable exchange of documents for endpoints, you must select which actions an endpoint may perform. This process is called configuring endpoint action policies.

Typically, you associate service actions with an endpoint. These actions are defined as ServiceOperation actions in the **Actions** form. For example, the `CustCustomerService.create` action creates a new customer in Microsoft Dynamics AX. ServiceOperation actions are registered automatically so you do not need to create or enable them.

You may also associate a SendXML action with an endpoint. A SendXML action defines an action that is independent of any service or class and is used in X++ to send XML (outbound).

For more information about actions, see Creating and configuring actions.

Associate an action with an endpoint

The default endpoint automatically has all service actions enabled. However, you must ensure that the service associated with the actions that you want to use is enabled. For more information, see Configure services.

Follow these steps to configure actions on an endpoint other than the default endpoint.

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Select the endpoint to configure and click **Action policies**. The **Overview** tab lists the available actions that are currently associated with the endpoint.

   Note:

   By default, all actions for all services as defined in the AOT are associated with the default endpoint. You do not need to enable them in the **Actions** form.
3. Press CTRL+N to enter a new action policy.
4. Click the **General** tab. In the **Action ID** field, select the action that you want to associate with the endpoint.

   The name of the action in the **Action ID** field and the name of the service class in the **Class name** field cannot be changed.
5. Select the **Is default policy** check box to use the default data policy. The default data policy specifies that all fields defined in the message schema will be used in the exchange. If you clear this field, the **Data Policies** button is enabled, and you can modify the data policy for the action.

   Note:

   This field is only editable for the default endpoint. By default, the **Is default policy** check box is selected for all actions associated with the default endpoint. If you want to modify the data policy for an action associated with the default endpoint, you must clear this field.
6. You can change the status of an action associated with the endpoint to **Enabled**, **Disabled**, or **Hold**.
   - Select **Enabled** to make the action active for this endpoint.
   - Selecting **Disabled** has the same effect as if the action was not configured on the endpoint.

- Select **Hold** to prevent outbound documents from being passed to the adapter and inbound documents from being passed to the document class.

  If the action status is **Hold**, the document is held in the queue and may be examined and resubmitted. For more information, see Edit and resubmit messages in the queues. For a synchronous exchange such as a Web service, an error message is generated for the hold condition.

In the **External identifier override** field, type an action identifier to override the **External identifier** field on the **Actions** form. Any messages referencing this endpoint and action must use the external identifier in the <Action> tag.

7. Select the **Automatically respond to errors** check box to send any errors that are encountered back to the caller.
8. Press CTRL+S to save.

Select a logging mode for an action on an endpoint

The logging mode defines how messages are logged on a per action per endpoint basis.

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Select an endpoint and click **Action policies**.
3. Select an action on the **Overview** tab.
4. On the **General** tab, select a logging mode for the endpoint action.
   - **Log Original** captures only the information for the initial document exchange.
   - **Log All** captures information about every transfer including all the different versions of a document, for example, the submitted XML, the XML generated after each pipeline transformation is applied, and so on.
   - **Log None** stores no data for this action and endpoint.

To view the document history by message or by document, click **Basic** > **Periodic** > **Application Integration Framework** > **Document history**.

Configure document-specific options including value mapping

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Select an endpoint and click **Action policies**.
3. Select an action on the **Overview** tab and click **Configure** to perform document-specific configuration, including value mapping.

   Value mapping is the translation of field data values based on business rules, for example, translating internal item numbers to vendor-specific item numbers or industry standard numbers depending on the trading partner. For more information, see About value mapping and "Set up external codes for AIF" in the Application and Business Processes Help.

Configure data policies for an action on an endpoint

For more information about data policies, see Configure endpoint action data policies.

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Select an endpoint and click **Action policies**.
3. Select an action on the **Overview** tab, and click **Data Policies** to configure required and allowed fields for the document transfer.

   Note:

   If the **Is default policy** check box is selected, the endpoint will use the default policy of all fields for that action. If you want to set specific data policies for the endpoint, you must clear this field first.

See Also

Configure endpoint action data policies

Configure a pipeline

Configure an endpoint

View document history

Manage document exchanges in AIF

Configure endpoint action data policies

When you set up a document exchange in Application Integration Framework (AIF), you decide, on a field-by-field basis, which data fields are transferred. This is known as the data policy. The data policy is defined for each action on each endpoint. You configure the data policy on the **Data Policies** form.

Default data policy

In Microsoft Dynamics AX, an action associated with the default endpoint can use a default data policy. This is specified by selecting the **Is default policy** field in the **Data Policies** form.

For a default endpoint action, using the default data policy means that the message uses the full document schema. Required fields as defined in the schema must be in the message. The message can contain any number of optional fields.

If you want to use a specific data policy for an action associated with the default endpoint, you must clear the **Is default policy** field and modify the action data policy as you would for any other endpoint.

Mandatory/required fields

There are two types of qualifiers for data fields on the **Data Policies** form: required and enabled. These have different meanings and effects depending on the direction of the transfer.

If a data field is allowed to be included in an inbound exchange, it is said to be enabled. For inbound documents, only fields that are enabled are allowed to be submitted by the endpoint. If a document is received that includes fields that are not enabled, the document is rejected and an exception is logged.

There are two terms that are used when discussing whether a field is required in a document: mandatory fields and required elements.

Term	Location	Description
Mandatory field	Database	Database field that has the `Mandatory` property set to `Yes`.
Required element	XML document	Element required to be present in the XML document to satisfy the schema. Required elements often correspond to mandatory fields in the database. A database field that is mandatory but that can be defaulted does not have to be required in the XML document.

Note:

For inbound documents, mandatory fields (that is, fields required by the Microsoft Dynamics AX database) should be set to **Enabled** and **Required** on the **Data Policies** form if they cannot be set by default in the database. For outbound documents, the fields to be sent must be set to **Enabled**.

Required fields and document direction

For inbound documents only (processed by a `create` action, for example), fields may be designated as **Required** if the document class defines them as mandatory (they are required for the database record to be inserted or updated and they cannot be defaulted). Additionally, the XML document may specify required elements depending on the business logic in the document class. You can also specify additional required elements by selecting **Required** for the field on the **Endpoint action data policies** form. However, you cannot use the data policy to make an element optional if it is required by the document class.

For an inbound document, fields that are enabled but not required are optional to the exchange. Fields in an inbound exchange that are required are automatically designated as enabled - if the document does not contain these fields, the document is rejected.

The concept of required fields does not apply to outbound transfers. For outbound documents, only fields that are enabled are included in the exchange.

Note:

When you clear the **Enabled** check box for a field used for calculating the value of another field, you may also need to clear the **Enabled** check box for the calculated field, so that unauthorized users may not be able to deduce the value of the original field that is not enabled. For more information about the calculated fields available in each document, see "Standard Axd Documents" in the Microsoft Dynamics AX SDK Help.

Configure data fields for an inbound document

You must first configure an endpoint and enable the actions for the exchange.

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Select an endpoint, click **Action policies**, and select an action.
3. Click **Data Policies** and then click **Data Policies** to configure required and allowed fields for the document transfer.

   Note:

   If the default endpoint is selected, all actions use the default data policy by default and the **Is default policy** field is selected for all actions. To configure an action data policy for the default endpoint, you must clear the **Is default policy** field to enable the **Data Policies** button.

   For fields that are required to be present in the XML document according to the document class, select the **Enabled** and **Required** check boxes.
4. For other fields in the document, you can select **Required** if the field is required for the document exchange (**Enabled** is automatically set).

   Note:

   If you find that your needs for the document transfers change, you can clear the **Required** check box.
5. Click **Set** to clear or select all fields at one time.

Configure data fields for an outbound document exchange

You must first configure an endpoint and enable the actions for the exchange.

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Select an endpoint, click **Action policies**, and select an action.
3. Click **Data Policies** and then click **Data Policies**

4. Select **Enabled** for each field to be included in the document transfer.

   Note:

   For an inbound document, fields that are enabled but not required are optional in the exchange. The concept of required fields does not apply to outbound transfers. Only fields for which you have selected the **Enabled** check box are sent in the transfer.

5. Click **Set** to clear or select all fields.

See Also

Configure endpoint action policies

Configure an endpoint

Creating and configuring a pipeline

A pipeline consists of a set of components that transform XML documents as they flow in or out of Microsoft Dynamics AX through Application Integration Framework (AIF). In addition to the pipeline components that ship with Microsoft Dynamics AX, the architecture of the pipeline allows developers to create and configure custom components to transform documents.

A separate pipeline consisting of one or more pipeline components may be specified for every endpoint action policy, which enables custom transformations on a per-endpoint basis. You can specify separate components for inbound documents and outbound documents.

Two pipeline components are installed with Microsoft Dynamics AX:

- A component for value substitution
- A component that enables Extensible Stylesheet Language Transformations (XSLT) document transformations

The `AifValueSubstitutor` pipeline component allows you to substitute one character string for another character string in a given field. This enables you to change field values (an item code, for example) in an inbound or outbound message to match the requirements of the system receiving the data. To apply XSLT document transformations, you must first import an XSLT style sheet into Microsoft Dynamics AX and then specify the `AifXMLTransform` pipeline component for the desired endpoint action policy.

Prerequisites

You can automatically configure pipeline components for actions that are associated with the default endpoint. Otherwise, before configuring any pipeline components, you must have the following:

- A local endpoint. For more information, see Create and configure local endpoints.
- An endpoint with an action policy and a data policy. For more information, see Configure an endpoint, Configure endpoint action policies, and Configure endpoint action data policies.

Create a pipeline

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints** and select an endpoint.
2. Click **Action policies** and select an action.
3. Click **Inbound Pipeline** or **Outbound Pipeline** depending on whether you want the transformations to occur on inbound documents or outbound documents.
4. Press CTRL+N to create a new pipeline component entry.
5. Select a component in the **Class name** field. There may be a delay while the system scans the AOT for pipeline components.
6. In the **Description** field, type a description of the pipeline.
7. Press CTRL+S to save and enable the **Configure** button.
8. Each pipeline component has different configuration requirements, so you see a different form when you click **Configure** for any pipeline component. For more information, see Configure a pipeline.

Configure a pipeline

Configuring a pipeline for an action on an endpoint involves specifying the pipeline components for a transformation of the document, in execution order, on the **Pipeline components** form. You can define pipeline components for inbound or outbound actions separately. For more information about creating a pipeline, see Creating and configuring a pipeline.

Two pipeline components are included with Microsoft Dynamics AX. You can configure these pipeline components to perform value substitution and XSLT transformations. Other custom pipeline components may be developed for your system by your team or outside consultants or partners. Configuration of any custom pipeline component depends entirely on the implementation of that component.

The two pipeline components available with your Microsoft Dynamics AX installation are:

- `AifValueSubstitutor` - For simple string mapping of field values.
- `AifXMLTransform` - For XSLT transforms of XML documents.

You can include as many pipeline components as you need to transform the document to meet the needs of the exchange.

Each pipeline component has different configuration requirements, so you see a different form when you click **Configure** for any pipeline component.

Configure value substitution

Before you can configure a value substitution pipeline, you must define lookup values. For more information, see About value lookups.

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints** and select an endpoint.
2. Click **Action policies** and select an action.

3. Click **Inbound Pipeline** or **Outbound Pipeline**.
4. In the **Class name** field, select `AifValueSubstitutor`. If there is no pipeline, press CTRL+N to add a new pipeline.
5. Press CTRL+S to save.
6. Click **Configure**.
7. On the **Pipeline Value Substitution Parameter Selection** form, select the parameter in the **Parameter name** field.
8. Click **Configure value substitution** to display the **Pipeline value substitution** form.
9. In the **Lookup table ID** field, select the lookup table identification for the value lookup table (that you entered on the **Value lookup** form) for the fields requiring value substitution. For more information about configuring value lookups, see About value lookups.

   Notes:

   - Values for **Lookup table ID** are filtered by type. If no values are displayed for **Lookup table ID**, you may need to return to the **Value lookup** form and enter a value for **Type** on the **General** tab.
   - On the **Pipeline value substitution** form, the following read-only fields appear:
   - **Element name** - The name of the data field.
   - **XPath** - Specifies where the data field fits into the schema hierarchy.
   - **Type** - The Microsoft Dynamics AX data type.

Configure an XSLT transform

To configure a transformation pipeline, you must first import the XSLT into the repository. Then you must create the transformation pipeline based on the XSLT. For more information about security best practices when implementing transformations, see Security considerations for AIF.

Add an XSLT style sheet to the XSLT repository

1. Click **Basic** > **Setup** > **Application Integration Framework** > **XSLT repository**.
2. Click CTRL+N to create a new record.
3. In the **XSLT ID** field, enter a unique identifier for the XSLT transform.
4. In the **Name** field, enter a text description for the transform.
5. Click **Import** and specify the file name of the XSLT style sheet for the transform.
6. Click **View** to view the XML for the style sheet, and then click **Save as** to export the XML to an .xsl file.

Create the pipeline

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints** and select an endpoint.
2. Click **Action policies** and select an action.

3. Click **Inbound Pipeline** or **Outbound Pipeline**.
4. In the **Class name** field, select `AifXMLTransform`. If there is no pipeline, press CTRL+N to add a new pipeline.
5. Click CTRL+S to save.
6. Click **Configure**.
7. On the **Pipeline XSLT transform** form, select **Apply transform to parameter** if the transform applies to an action parameter. If this field is not selected, the transformation will apply to the entire document.
8. If you select **Apply transform to parameter**, then you must select a parameter from the **Parameter name** field.
9. In the **XSLT ID** field, select the identification of the XSLT transform that you entered on the **XSLT repository** form.
10. If you want any Microsoft Visual Studio scripts in the XSLT file to be executed, select **Scripting enabled**.
11. Press CTRL+S to save.

Note:

When an XSLT pipeline transformation run, errors are logged only if the component throws an exception. If you use an incorrect XSLT, an exception will not be generated. An XSLT only transforms matching nodes; if there are no matching modes, then no transformation is applied and no error is generated.

See Also

Creating and configuring a pipeline

About value lookups

About value lookups

With value lookups, you can substitute one character string for another in any field of a document. You can implement value lookups using the `AifValueSubstitutor` pipeline component. For more information, see Creating and configuring a pipeline and Configure a pipeline.

You can also create value lookups for any pipeline component, if you know the data type for the field. After creating a value lookup using the **Configure value lookup** form, you relate that value lookup to the pipeline component by entering the value lookup identification when you configure the pipeline component.

Configure value lookups for a pipeline component

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Value lookup**.
2. Press CTRL+N to create a new line.

3. On the **Overview** tab, enter:
   - A new identification in the **Lookup table ID** field.
   - A name for the table in the **Name** field.
4. On the **General** tab, select the Microsoft Dynamics AX data type from the list of available data types. For the data types that reference a table in Microsoft Dynamics AX, the internal values are populated from that table.
5. Enter the internal values and the external values for the string substitution in the lower pane of the form.

Use a value lookup table with the AifValueSubstitutor pipeline component

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints** and select an endpoint.
2. Click **Action policies** and select an action.
3. Click **Inbound Pipeline** or **Outbound Pipeline**.
4. In the **Class name** field, select `AifValueSubstitutor`. If there is no pipeline, press CTRL+N to add a new pipeline and press CTRL+S to save.
5. Click **Configure**.
6. On the **Pipeline Value Substitution Parameter Selection** form, select the parameter in the **Parameter name** field.

   The **Pipeline value substitution** form is populated with the data fields that are **Enabled** on the **Data Policies** form for the document.
7. Enter the **Lookup table ID** for each data field to be substituted. This is the identifier you entered on the **Value lookup** form.
8. In the **XPath** field, you can view the location in the XML schema hierarchy where the element resides.
9. In the **Type** field, you can view the Microsoft Dynamics AX data type.

See Also

Creating and configuring a pipeline

Configure a pipeline

About value mapping

Value mapping is the translation of field data values that is based on business rules; for example, translating internal item numbers to vendor-specific item numbers or industry-standard numbers, depending on the trading partner.

Value mapping can be performed on inbound and outbound XML documents and is configured on each document endpoint. Value mapping creates a translation index between the specific field in Microsoft Dynamics AX and an external field in the document. This index enables more flexibility when you are handling various internal, vendor-based, or industry-based codes.

Value mapping example

One vendor requires all item numbers on the outbound purchase requisition from Microsoft Dynamics AX to be designated by using the vendor's item number system, whereas another vendor (which is a separate endpoint) does not have this requirement, and a third vendor (also a separate endpoint) requires a common industry item number system on the purchase requisition lines.

To handle these varying requirements, each purchase requisition document can be configured differently for each endpoint, and the item number value mapping settings can be configured to reflect the vendor requirements.

When the outbound purchase requisition is generated, the active endpoint configuration translates the internal item number codes to the codes that are specified in the **Value Mapping** form. This translation ensures that each vendor receives the purchase requisition in the format that their system requires.

You can map the values of the fields shown in the following table

Type	Field
Trading partners	• Vendor Account number • Customer Account Number
Addresses	• Country code • County code • State code • Zip/postal code
Items	• Item number • Units • Warehouse numbers
Other data	• Currency code • Delivery Methods • Terms of delivery • Misc. charges

Value mapping for document transformation

Value mapping is configured in the **Value Mapping** form. For more information about the value mapping form, see topic "Value mapping (form)" in the Application and Business Processes Help.

This section describes forms that are used to set up value mapping for endpoints and external codes for different fields used in the documents.

External Codes

You set up, define, and maintain external codes in the **External codes** form. These external codes are for different fields used to send and receive specific documents electronically through Application Integration Framework (AIF).

If it is necessary, set up external codes for:

- Trading partners (vendor and customer account numbers)
- Addresses (countries/regions, counties, states, and Postal/ZIP Codes)
- Inventory (item numbers, bar codes, item units, and warehouses)
- Currency codes
- Delivery methods
- Delivery terms
- Miscellaneous charges
- Dimensions (department, purpose, cost center)

Endpoint Value Mapping

Map the values that are used for the active action policy and for the particular endpoint such as item number, customer account number, vendor account number, and terms of delivery in the **Value Mapping** form. You can map values for the following elements:

- Trading partners (vendor and customer account numbers)
- Addresses (countries/regions, counties, states, and Postal/ZIP Codes)
- Inventory (item numbers, bar codes, item units, and warehouses)
- Currency codes
- Delivery methods
- Delivery terms
- Miscellaneous charges
- Dimensions (department, purpose, cost center)

Map values

You can map the following internal values from Microsoft Dynamics AX to external values in inbound or outbound XML documents by using the **Value Mapping** form.

For more information, see "Set up external codes for AIF" and "Currency code document value" in the Application and Business Processes Help.

Map vendor or customer numbers field values

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Select the endpoint for which you want to map field values, and then click **Action policies** to open the **Endpoint Action Policies** form.

3. Select the service action for which you want to map field values, and then click **Configure** to open the **Value Mapping** form.
4. On the **Partners** tab, select the trading partner (vendor or customer) for which you want to map field values.
5. In the **Document value** field, select the type of field value mapping:
   - **Not specified**
   - **Our**
   - **External code**
6. If you selected **External code** in step 5, select the external code in the **Customer code** or **Vendor code** field.

Map address field values

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Select the endpoint for which you want to map address field values, and then click **Action policies** to open the **Endpoint Action Policies** form.
3. Select the action for which you want to map address field values, and then click **Configure** to open the **Value Mapping** form.
4. On the **Addresses** tab, select the address type (countries, counties, states or ZIP/Postal Codes) for which you want to map field values.
5. In the **Document value** field, select the type of field value mapping:
   - **Not specified**
   - **Our**
   - **External code**
6. If you selected **External code** in step 5, specify the external code in the corresponding field.

Map terms of delivery, delivery methods, and misc. charges field values

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Select the endpoint for which you want to map field values, and then click **Action policies** to open the **Endpoint Action Policies** form.
3. Select the action for which you want to map field values, and then click **Configure** to open the **Value Mapping** form.
4. On the **Other base data** tab, select the field value (terms of delivery, delivery methods, or misc. charges) for which you want to map field values.
5. In the **Document value** field, select the type of field value mapping:
   - **Not specified**
   - **Our**
   - **External code**
6. If you selected **External code** in step 5, specify the external code in the corresponding field.

Map currency code field values

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Select the endpoint for which you want to map currency field values, and then click **Action policies** to open the **Endpoint Action Policies** form.
3. Select the action for which you want to map currency field values, and then click **Configure** to open the **Value Mapping** form.
4. On the **Other base data** tab, in the **Handling currency codes** section, in the **Document value** field, select the type of currency field value mapping:
   - **Not specified**
   - **Our**
   - **External code**
   - **ISO currency code**
5. If you selected **External code** in step 4, specify the external code in the **Currency code** field.

Map units and warehouse numbers field values

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Select the endpoint for which you want to map field values, and then click **Action policies** to open the **Endpoint Action Policies** form.
3. Select the action for which you want to map field values, and then click **Configure** to open the **Value Mapping** form.
4. On the **Items** tab, select the field value (units or warehouse number) for which you want to map field values.
5. In the **Document value** field, select the type of field value mapping:
   - **Not specified**
   - **Our**
   - **External code**
6. If you selected **External code** in step 5, specify the external code in the corresponding field.

Map item numbers field values

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Select the endpoint for which you want to map field values, and then click **Action policies** to open the **Endpoint Action Policies** form.
3. Select the action for which you want to map item number field values, and then click **Configure** to open the **Value Mapping** form.

4. On the **Items** tab, in the **Handling item numbers** section, in the **Document value** field, select the type of field value mapping:
   - **Not specified**
   - **Our**
   - **External code**
   - **External item number**
   - **Bar code**
   - **Company item**
5. If you selected **External code** in step 4, specify the external code in the **Item number code** field.

   If you selected **Bar code** in step 4, select the bar code type that the company uses in the **Bar code setup** field, and enter the bar code type that your trading partner uses in the **Value** field.

Configure data validation and defaulting

In Application Integration Framework (AIF), validation of data in an inbound XML document is usually performed by the `Ax`<*Table*> classes to ensure that referential integrity, number sequence, and business logic restrictions are enforced and to prevent incorrect data from being inserted into the application. If you disable data validation, the data from the inbound XML document is inserted into the application regardless of the data quality.

Defaulting of fields is performed by the `Ax`<*Table*> classes to set predefined values in the application data tables if the inbound document does not contain these values. Otherwise, the inbound document fails.

If you disable field defaults, data from the inbound XML document is inserted into the application regardless of the presence of required field values. This can result in some fields not containing values, the document failing, and an error being logged if any fields marked in the table as mandatory are empty.

Note:

Defaulting and validation are enabled by default.

How to disable data validation for inbound documents

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Select the endpoint for which you want to disable data validation, and then click **Action policies** to open the **Endpoint Action Policies** form.
3. Select the action for which you want to disable validation, and then click **Configure** to open the **Value Mapping** form.
4. In the **Setup** tab, clear the **Validate input** field.
5. To enable data validation, select the **Validate input** field again.

How to set disable defaulting for inbound documents

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Select the endpoint for which you want to disable defaulting, and then click **Action policies** to open the **Endpoint Action Policies** form.
3. Select the action for which you want to disable defaulting, and then click **Configure** to open the **Value Mapping** form.
4. In the **Setup** tab, clear the **Use defaulting** field.
5. To enable data validation, select the **Use defaulting** field again.

Configure document parameters

This section lists the setup parameters for some commonly-used documents to be sent or received by Application Integration Framework (AIF).

Document notes

Define the name of the note document type in the **Document management parameters** form. For more information, see "Document management parameters (form)" in the Application and Business Processes Help.

Inbound Sales Order document

- Define where inbound sales order documents are to be received in the **Accounts receivable parameters** form (**AIF** tab > (**Order type** field). For more information, see "Accounts receivable parameters (form)" in the Application and Business Processes Help.
- The inbound sales order is received in the **Sales order** form. For more information, see "Sales orders (form)" in the Application and Business Processes Help.
- The inbound sales order is received in the **Sales orders** form. For more information, see "Sales orders (form)" in the Application and Business Processes Help.

Inbound Purchase Invoice document

- Define the default register to receive the purchase invoice document in the **Accounts payable parameters** form (**AIF** tab > (**Journal name** field). For more information, see "Accounts payable parameters (form)" in the Application and Business Processes Help.
- Set up the manner in which duplicate invoices are processed in the **Accounts payable parameters** form (**Updates** tab > (**Check the invoice number used** field). For more information, see "Accounts payable parameters (form)" in the Application and Business Processes Help.

Inbound Packing Slip document

Define the default settings for the inbound packing slip document in the **Accounts receivable parameters** form (**AIF** tab > **Packing slip** field). For more information, see "Accounts receivable parameters (form)" in the Application and Business Processes Help.

Inbound Inventory Counting document

Set the default inventory counting journal for the inventory counting document in the **Inventory parameters** form (**AIF** tab > (**Counting** field). For more information, see "Inventory parameters (form)" in the Application and Business Processes Help.

Inbound Transfer Journal document

Set the default counting transfer journal for the inventory transfer document in the **Inventory parameters** form (**AIF** tab > (**Transfer** field). For more information, see "Inventory parameters (form)" in the Application and Business Processes Help.

Inbound Inventory Profit/Loss document

Set the default counting-profit-and-loss journal for the inventory profit-and-loss document in the **Inventory parameters** form (**AIF** tab > (**Profit/Loss** field). For more information, see "Inventory parameters (form)" in the Application and Business Processes Help.

Inbound Price/Discount agreement document (trade agreements)

Set the default counting price discount journal for the price discount document in the **Accounts payable parameters** form (**AIF** tab > **Price/Discount agreement** field). For more information, see "Accounts payable parameters (form)" in the Application and Business Processes Help.

Limit outbound documents

You can configure an endpoint to limit the number of documents or entity keys that are returned from a request in Application Integration Framework (AIF). Limiting the number of records returned reduces the size of the XML document that the Application Object Server (AOS) processes and improves navigation of the XML document or entity keys.

A scenario in which limiting documents may be useful is when the `read` and `find` actions are called. This is because those actions will typically be returning full documents and the number of records returned is unknown at request time leading to potentially large data sets being returned.

Limit the number of returned documents

1. Click **Basic** > **Setup** > **Application Integration Framework** > **Endpoints**.
2. Select the endpoint for which you want to define return document limits, and then click **Action policies** to open the **Endpoint Action Policies** form.

3. Select the action for which you want to define return document limits, and then click **Configure** to open the **Value Mapping** form.
4. In the **Limit number of documents** field on the **Setup** tab, select **Yes** to limit the number of documents that are returned by a query. By default, the value in the (**Limitation type** field is **Default**, and the maximum number of documents returned is set to 1000.
5. To change the number of returned documents, select **Specified** in the (**Limitation type** field and enter the maximum number of documents returned by a request in the (**Max. number of documents** field.

Note:

If you send a request to AIF and anticipate the return of many large documents, you may want to first send a request using the `findKeys` action to return only all the entity keys (IDs) that match the criteria. After you receive a message with the entity keys, you can then manage processing of the data based on how many records the query returns.

Manage document exchanges in AIF

After you set up data exchanges with external systems using Application Integration Framework (AIF), you will need to maintain this integration. Microsoft Dynamics AX provides the ability to manage your document exchanges throughout their life cycle and troubleshoot any issues with data transfer. Maintaining and managing document exchanges involves:

- Monitoring traffic and viewing document history as documents pass through the framework.
- Clearing and reviewing messages in the queues for adapter-based exchanges.
- Viewing the exception logs when problems arise.
- Editing and resubmitting messages that contain formatting errors.
- Stopping and starting the batch services when necessary.

Document history

For both adapter-based and Web services-based exchanges, information about messages and document history is organized by action for each endpoint. You set the parameters for logging this information when you configure endpoint action policies.

You view the logged information on the **Document history** form by clicking **Basic** > **Periodic** > **Application Integration Framework** > **Document history**.

For more information, see View document history, Viewing the document log and "AIF Document history (form)" in the Application and Business Processes Help.

Queue manager

For adapter-based exchanges, you use the Microsoft Dynamics AX batch functionality to start and stop the operation of the four services that move messages in to and out of the document

processing queues. After documents are exchanged, you can monitor the activity of documents in the queues in the Queue manager. You can also edit and resubmit documents that have errors with the **Queue manager** form.

To open the Queue manager, click **Basic** > **Periodic** > **Application Integration Framework** > **Queue manager**. For more information, see Edit and resubmit messages in the queues.

Exceptions

You can view information about AIF error messages when they occur by clicking **Basic** > **Periodic** > **Application Integration Framework** > **Exceptions**. The **Exceptions** form contains information about the module and subsystem where the error occurred, a description of the error, when the error was logged, the user associated with the error, and the form or business logic where the error occurred.

For more information, see Viewing the exceptions log.

Starting and stopping the batch services

For adapter-based exchanges, you use the Microsoft Dynamics AX batch functionality to start and stop the operation of the four AIF services: **AIFOutboundProcessingService**, **AIFInboundProcessingService**, **GatewaySendService**, and **GatewayReceiveService**. These services move messages in to and out of the document processing queues. You may need to start and stop these services to troubleshoot any issues or to change the batch job settings such as recurrence.

For more information, see Start and stop the asynchronous AIF services.

Start and stop the asynchronous AIF services

For Application Integration Framework (AIF) to begin sending and receiving documents for adapter-based exchanges, the four services that move documents through the queues must be running as batch jobs within Microsoft Dynamics AX:

- **AIFGatewayReceiveService** - Communicates with the adapters, receiving messages from their external locations (channels) and bringing them into AIF by putting them in the AIF gateway queue.
- **AIFInboundProcessingService** - Takes inbound messages received by AIF through the **AIFGatewayReceiveService** and then sends them to the correct Microsoft Dynamics AX document class (`Ax`<*Document*> class).
- **AIFOutboundProcessingService** - Combines an outbound document from Microsoft Dynamics AX with the document metadata such as endpoint and channel information and creates a fully-formed AIF message. The service then sends the message to the AIF gateway queue where it is picked up by the **AIFGatewaySendService**.
- **AIFGatewaySendService** - Communicates with the adapters and writes the messages to the correct external locations.

Until these batch jobs are started, no documents can be processed in adapter-based exchanges. Inbound documents do not enter your Microsoft Dynamics AX system and outbound documents accumulate in the AIF outbound processing queue.

Each of these AIF services is implemented as a task in a Microsoft Dynamics AX batch job. For more information about batch jobs, see "Processing batch jobs" in the Application and Business Processes Help.

You can create a single batch job to support both inbound and outbound processing tasks, or you can create multiple batch jobs to run on one or more computers, depending on your processing needs.

To start the services and allow AIF to begin processing documents from the queues, perform the following steps:

1. Create a batch job.
2. Add a task for each of the AIF queue processing services to the batch job.
3. Start the batch job by changing the status to **Waiting**.

Create a batch job

Follow these steps to create a batch job. For more information, see "Batch job (form)" in the Application and Business Processes Help.

1. Click **Basic** > **Inquiries** > **Batch job**.
2. On the **Batch job** form, press CTRL+N to add a new job.
3. In the **Job description** field, enter a text description for the batch job.
4. In the **Scheduled start date/time** field, click the calendar icon to select a date and time for the batch job to start processing.
5. In the **Save job to history** field on the **General** tab, select how the batch job processing should be saved to the batch history:
   - **Always** - Saves a record to the batch history each time this batch job runs.

     Note:

     If you have this set to always save batch job processing history and you have a high recurrence frequency set for the batch job, it could lead to a high volume of batch history records.
   - **Errors only** - Saves a record to the batch history only if an error occurs.
   - **Never** - Does not save a record to the batch history regardless of whether the batch job runs successfully or encounters an error.
6. Press CTRL+S to save.
7. Click **Recurrence** to set how frequently the batch job should run.

Note You cannot change batch information while the job is running or waiting to be run. You must change the batch status to **Withhold** before changing batch information.

Add tasks to the batch job

After you create the batch job, you must add a task to the job for each of the AIF queue services. The services should run in a specific order depending on whether the document exchange is inbound or outbound.

- Inbound - For inbound exchanges, the **AIFGatewayReceiveService** should run first followed by the **AIFInboundProcessingService**. This is because the **AIFGatewayReceiveService** retrieves the inbound message from the external location (channel) and puts it in the queue and the **AIFInboundProcessingService** processes the message. The **AIFGatewayReceiveService** and **AIFInboundProcessingService** services are responsible for bringing messages into AIF.
- Outbound - For outbound exchanges, the **AIFOutboundProcessingService** should run first followed by the **AIFGatewaySendService**. This is because the **AIFOutboundProcessingService** processes the message and the **AIFGatewaySendService** sends it to the proper location (channel). The **AIFOutboundProcessingService** and **AIFGatewaySendService** services are responsible for sending messages out of AIF.

For more information, see "Batch tasks (form)" in the Application and Business Processes Help.

To add tasks for the AIF services to the batch job, follow these steps:

1. Click **Basic** > **Inquiries** > **Batch job**.
2. Select a job and click **View tasks**. This opens the **Batch tasks** form.
3. Press CTRL+N to add a new task.
4. In the **Task description** field, enter a text description of the task.
5. In the **Company accounts** field, select the company for which the task will run.
6. In the **Class name** field, select **AifGatewayReceiveService**. This specifies that this task will run the receive service.
7. Press CTRL+S to save.
8. Repeat steps 3 through 7 and select **AifInboundProcessingService** in the **Class name** field.
9. Repeat steps 3 through 7 and select **AifOutboundProcessingService** in the **Class name** field.
10. Repeat steps 3 through 7 and select **AifGatewaySendService** in the **Class name** field.
11. Add the appropriate batch constraints to ensure that the services are processed in the correct order.

Start the batch job

When you create a batch job, the status is **Withhold** by default. After you have added the tasks to the job, you must change the job status to **Waiting** to start the processing of the batch job tasks.

1. Click **Basic** > **Inquiries** > **Batch job**.
2. Select the batch job.
3. Click **Functions** > **Change status**.
4. In the **Select new status** form, select **Waiting**. The AIF services will now start processing.

The batch job status will change to **Executing** when the batch job is running. To refresh the batch job form, press F5.

Stop the batch job

To stop processing of the batch job tasks, follow these steps.

1. Click **Basic** > **Inquiries** > **Batch job**.
2. Select the batch job.
3. Click **Functions** > **Change status**.
4. In the **Select new status** form, select **Withhold**. The AIF services will now stop processing.

Note:

You cannot change the status of the batch job while the batch job status is **Executing**. You must wait until the status changes to **Waiting** or **Ended**. Press F5 to refresh the batch job form and check the status.

View document history

Documents that are exchanged using Application Integration Framework (AIF) reflect data in the Microsoft Dynamics AX database, such as a sales order. To exchange documents and data with external systems, AIF creates a message which contains header information with the document data.

You can view document and message information on the **Document history** form by clicking **Basic** > **Periodic** > **Application Integration Framework** > **Document history**.

- If you select **Message** in the **Display by** field, you see information about the related action, the source and destination endpoints, and the date and time that the log entry was created.
- If you select **Document** in the **Display by** field, in addition to the information above, you also see information about the form name and the entity of the underlying document.

Information about messages and document history are organized by action for each endpoint. Set the parameters for logging when you configure endpoint action policies. For more information, see Configure endpoint action policies.

View general document history

1. Click **Basic** > **Periodic** > **Application Integration Framework** > **Document history**.
2. In the **Display by** field, you can filter the display by selecting either **Message** or **Document**.
   - When you filter by **Message**, you see the following information about the message: the action, the source and destination endpoints, and the date and time of the message transfer.
   - When you filter by **Document**, in addition to the items above, you also see the form name and entity key for the document.
3. Click the **General** tab to view the **Message ID** field.
4. Click the **Details** tab to view the following information:
   - The message direction (inbound or outbound).
   - The pipeline identification (if any).
   - User information for the endpoint (the Microsoft Dynamics AX user associated with the endpoint) and the submitting user. This is the user associated with the process that submitted the message (either the Microsoft Dynamics AX user that submitted the message for the source endpoint or a trusted intermediary).

     For more information, see Configure an endpoint.
   - For outbound documents sent in response to read requests, the **Request message ID** field shows the message ID for the original request.
   - Processing details, which include the **Channel**, the **Adapter**, and the **Transport address** used in the exchange.
5. Click **Correlation** to view the record in the database that corresponds to the message.
6. Click **Document logs** to view the raw XML for each version of the document as it is transformed by each of the components in the pipeline.
7. Click **Clear document XML** to clear all or some of the XML for any of the versions of the document that currently exist in the system.

View the data in a document for a message

1. Click **Basic** > **Periodic** > **Application Integration Framework** > **Document history**.
2. In the **Display by** field, filter by **Message**.
3. Select a message and click **Correlation**.
4. On the **Document correlation** form, view the form name, table name, and entity key for the database record contained in the message that you selected. You can also view a record for each of the numbered versions of the document and the related processing steps.
5. Click **View** to see the data displayed in the default form for that document.

Delete a message

When you delete a message on the **Document history** form, you delete it from the `AifMessageLog` table.

1. Click **Basic** > **Periodic** > **Application Integration Framework** > **Document history**.
2. In the **Display by** field, filter by **Message**.
3. Press ALT+F9 to delete the record from the document history.

Delete the XML for a document

1. Click **Basic** > **Periodic** > **Application Integration Framework** > **Document history**.
2. In the **Display by** field, filter by **Document** and select a document.
3. Click **Clear document XML**.
4. To clear all the XML document images in the system, click **Clear all versions**.
5. To clear all versions of the XML document except the version closest to the local endpoint, click **Clear interim versions**.
    - For outbound documents, this clears all versions except the highest-numbered version.
    - For inbound documents, this clears all versions except the first one.

Edit and resubmit messages in the queues

The **Queue manager** form displays information about messages in the Application Integration Framework (AIF) queues, including the following values for the status of the message:

- **Ready**
- **In Process** (for inbound messages only)
- **Hold**
- **Error**
- **In Transport Process** (for outbound messages only)
- **Malformed XML**

If the status for a message is **Ready**, you can change it to **Hold** and vice versa. You can delete or edit a particular message if its status is set to **Error** or **Hold**.

For more information, see Start and stop the asynchronous AIF services and "AIF Queue manager (form)" in the Application and Business Processes Help.

View message status and details

To view the status and other details about a message, follow these steps.

1. Click **Basic** > **Periodic** > **Application Integration Framework** > **Queue manager** to view a list of current messages in the AIF queues.
2. On the **Overview** tab, view the channel, direction, status, action, source endpoint, destination endpoint, and any associated error message.
3. On the **Details** tab, view information about the submitting user, the Microsoft Dynamics AX user identification for the endpoint, and the date and time the message was created.
4. Click **Refresh** to update the display.
5. Click **Document log** to view information about the document contained in the message, depending on the log options set for the endpoint on the **Endpoint Action Policies** form:
   - If the **Logging mode** field is set to **Log All**, then you can click **Document log** to view the document at each stage of its transformation.
   - If the **Logging mode** field is set to **Log Original**, then you can click **Document log** to view the original document.
   - If the **Logging mode** is set to **Log None**, then no document log is available.

Delete a message

If a message remains unprocessed in the AIF queues, you can delete it in Queue manager.

1. Click **Basic** > **Periodic** > **Application Integration Framework** > **Queue manager** to view a list of current messages in the queues.
2. If the value in the **Status** field is **Error**, **Malformed XML**, or **Hold**, then you can delete it.
3. Press ALT+F9 to delete the message.

Edit and resubmit a message

If a message enters the queue but cannot be processed because of an error, it may be possible to edit and resubmit the message. To resubmit a message, follow these steps.

1. If the message status is **Error** or **Hold**, click **View message** to see the message and optionally write the message to an XML file after entering the file name and path.
2. Edit the file that you just saved in any XML editor to correct the field or fields in error.
3. Click **Import message** to import the file.
4. To signal the queue to begin processing the message, change the status of the message from **Error** or **Hold** to **Ready**.

Viewing the document log

You can store copies of the XML code for documents that are exchanged with Application Integration Framework (AIF) and view them in the document log. When you configure the action policy for an endpoint, you set the logging mode to one of these values for each action on the endpoint:

- **Log Original** - Only the original document is stored in the log.
- **Log All** - A version of the document is stored after each pipeline transformation is applied in addition to the original document.
- **Log None** - No document XML is stored.

For more information about setting these options, see Configure endpoint action policies.

To view the document log, follow these steps.

1. Click **Basic** > **Periodic** > **Application Integration Framework** > **Document history**.
2. In the **Display by** field, select **Document**.
3. On the **Overview** tab, select a document and click **Document logs**.
    - In the **Document log** form, you can view a record of the numbered versions of the document and the related processing steps, as well as the date and time for each log entry.
    - To view the XML associated with one of the versions, select the record and click **View XML**.

Viewing the exceptions log

The exceptions log contains a record of all the errors that occur during document exchanges in Application Integration Framework (AIF). On the **Exceptions** form, you can view information about the module and subsystem where the error occurred, a description of the error, when the error was logged, the user associated with the error, and the form or business logic where the error occurred.

View the exception log

1. Click **Basic** > **Periodic** > **Application Integration Framework** > **Exceptions**.
2. On the **Overview** tab, select an exception record.
3. Click **View** to see the form or business logic related to the exception, if it is available.
4. Click **Exception help** to see more information about the exception, if more information is available.

Clear the exception log

1. Click **Basic** > **Periodic** > **Application Integration Framework** > **Exceptions**.
2. On the **Overview** tab, select an exception record.
3. Click the delete icon on the toolbar or press ALT+F9 to delete the exception record from the system.

Note:

You can use the SHIFT and CTRL keys to select multiple records in the exception log grid.

Maintenance and data recovery

A Microsoft Dynamics AX system requires ongoing maintenance, which includes the following processes:

- Back up and recover databases
- Back up and recover application files
- Monitor specific events, either in Microsoft Dynamics AX or in the database

You will need a maintenance strategy for all the environments that you run: production, development, and test.

This section provides information about planning backup and data recovery strategies for the Microsoft Dynamics AX system.

Backups and data recovery

This section provides basic information about backup and recovery strategies for Microsoft Dynamics AX. The following topics are included:

- Plan database backups
- Plan application file backups
- Plan for disaster recovery

For information about backing up and restoring registry settings, see Save or export a configuration (Server).

Plan database backups

Include all the databases in your Microsoft Dynamics AX system in your backup and recovery strategy. These databases can include the following:

- A Microsoft Dynamics AX database, either SQL Server or Oracle
- A SharePoint database to support Enterprise Portal
- A SQL Server 2005 Reporting Services database to support ad hoc reporting
- A SQL Server Analysis Services database to support OLAP reporting

Creating backups will help you recover a damaged database. Backups of a database are also useful for routine purposes, such as copying a database from one server to another, setting up database mirroring, and archiving for governmental purposes.

Backing up and restoring data should be customized for a particular environment and must work with the available resources. A well-designed backup and recovery strategy maximizes data availability and minimizes data loss, considering your particular business requirements.

The backup part of the strategy defines the type and frequency of backups, the nature and speed of the hardware that is required for them, how backups are tested, and where and how backup media is stored (including security considerations).

The recovery part of the strategy defines how databases should be restored to meet your goals for availability of the database and for minimizing data loss, and who should recover the data. We recommend that you document your backup and recovery procedures and keep a copy of the documentation in your operations manual.

Designing an effective backup and recovery strategy requires careful planning, implementation, and testing. Consider a variety of factors, including:

- The production goals of your organization for the databases—especially the requirements for availability and protection of data from loss.
- Constraints on resources such as hardware, personnel, space for storing backup media, and the physical security of the stored media.
- The nature of each of your databases:
    - How frequently does the data in each database change?
    - Are some tables modified more frequently than others?
    - What are your critical database production periods? What are the usage patterns during these periods?
    - When does the database experience heavy use, resulting in frequent inserts and updates? You might want to schedule differential or log backups during periods of the heaviest use and schedule full backups during off-peak hours.

Refer to the database documentation for detailed information about how to select and implement a backup and recovery strategy.

See Also

Microsoft SQL Server documentation

Microsoft Windows SharePoint Services documentation

Plan application file backups

The Microsoft Dynamics AX business logic is stored in application files. To make sure that all modifications and customizations are saved if a computer fails, regularly back up the application files directory using Microsoft Windows backup or another backup system.

By default, application files are located in the **Program Files\Microsoft Dynamics AX\50\Application\Appl\<InstanceName>** directory. Examples of the files in this directory include indexes, headers, and labels for each layer.

Refer to the operating system documentation for detailed information about how to select and implement a file backup and recovery strategy.

Plan for disaster recovery

To make sure that all systems and data can be quickly restored to normal operation if a failure occurs, you must create a disaster recovery plan. When you create this plan, consider scenarios for different types of disasters that might affect the system, including natural disasters, such as a fire, and technical disasters, such as a two-disk failure in a RAID-5 array. When you create a disaster recovery plan, identify and prepare for all the steps that you must follow to respond to each type of disaster. Testing the recovery steps for each scenario is important. We recommend that you verify your disaster recovery plan through the simulation of a data loss event.

Consider disaster recovery planning in the context of your particular environmental and business needs. For example, suppose a fire destroys your 24-hour data center. Are you certain you can recover? How long will it take you to recover and have the system available? How much data loss can users tolerate?

Ideally, your disaster recovery plan includes a time estimate for recovery and expectations for the extent of recovery. For example, you might determine that after the acquisition of specified hardware, recovery will be completed in 48 hours, and data will be restored only through the end of the previous week.

A disaster recovery plan can be structured in many different ways and can contain many types of information. Disaster recovery plan types include the following:

- A plan to acquire hardware.
- A communication plan.
- A list of people to be contacted if there is a disaster.
- Ownership for each phase of the recovery plan.
- A checklist of required tasks for each recovery scenario. To help you review how disaster recovery progressed, it helps to initial each task as it is completed, and indicate the time of completion on the checklist.

Ensuring disaster readiness

To make sure that you can recover the system after a disaster, we recommend that you periodically perform the following activities:

- Test your backup and recovery procedures thoroughly before a real failure occurs. Testing helps make sure that you have the required backups to recover from various failures, that your procedures are clearly defined and documented, and that they can be executed smoothly and quickly by any qualified operator.
- Perform regular database, transaction log, and file system backups to minimize the lost data. We recommend that you back up both system and user databases.
- Maintain system logs in a secure manner. Keep records of all service packs installed on Microsoft Windows, the database, and Microsoft Dynamics AX.

- On another server or set of servers, assess the steps that you have to take to recover from a disaster. Amend the steps as necessary to suit the local server environment, and then test the amended steps.
- Make sure that you understand and document the database and file permissions required to recover the database and application folder and return the server to a working production state.
- Plan for the loss of each Microsoft Dynamics AX server, including the AOS server, database server, application file server, and Enterprise Portal server. You should also understand the implications of the loss of the domain controller to the Microsoft Dynamics AX implementation.
- Review related documentation, such as the Windows SharePoint Services Administration Guide, so that you can recover the other databases used with Microsoft Dynamics AX.

System maintenance tasks

This section describes maintenance tasks that you should perform periodically, such as cleaning up user logs. This section contains the following topics:

Prepare Microsoft Dynamics AX for maintenance

Use the following procedure to take Microsoft Dynamics AX offline for users but keep it available to the administrator to run maintenance tasks.

1. Open the **Online users** form (**Administration** > **Online users**).
2. On the **Server Instances** tab, select each AOS instance except the one that will be used to perform maintenance, and then click **Reject new clients**.

   In-progress client sessions will not be disconnected automatically, but no new connections will be accepted. If you do not want to wait until current client sessions are logged off, you can forcibly disconnect users by selecting them and by clicking the **End sessions** button.
3. Use the **Services** control panel (**Start** > **Administrative Tools** > **Services**) to stop all AOS instances except the one that will be used to perform maintenance.
4. Open the **Server configuration** form (**Administration** > **Setup** > **Server configuration**).
5. On the **Overview** tab, select the AOS instance that is still running.
6. In the **Max concurrent sessions** field, change the number to 1. When this field is set to 1, any users trying to connect are denied access while the administrator is connected.

7. Press CTRL+S to save changes. You can now perform maintenance tasks on Microsoft Dynamics AX.
8. After maintenance has been completed, restart all AOS instances and allow client connections.

Clean up user logs

Use the **User log cleanup** form to delete user log information that is no longer needed.

1. Click **Administration** > **Inquiries** > **User log** > **Clean up** button.
2. Enter a value in the **History limit (days)** field to define a limit for the deletion.

   Only log information older than the given number of days is deleted.
3. Click **Select** to open the inquiry form.
4. Select cleanup criteria.

   Select a user or a range of users, and, optionally, additional user information, such as date and time.
5. Click **OK** to return to the **User log cleanup** form.
6. Click **OK** to perform the cleanup once, or click the **Batch** tab to define parameters to clean up the user log at regular intervals.

Note:

The cleanup process permanently deletes data.

Optimizing performance

This section provides information about monitoring and tuning servers to improve Microsoft Dynamics AX performance. This section contains the following topics:

Manage load balancing

Use load balancing to distribute the user load among multiple Application Object Server (AOS) instances. AOS load balancing in Microsoft Dynamics AX has the following benefits.

- Less downtime: If one AOS instance fails, new sessions can be automatically routed to different AOS instances.
- Easier maintenance: Take an AOS instance offline for maintenance while other AOS instances handle the load.
- Greater scalability: AOS instances can be added as needed.

This section contains following topics:

Create a load balancing cluster

To distribute the user and transaction load among multiple Application Object Server (AOS) instances, add the instances to a load balancing group, or cluster.

You can create multiple clusters so that users performing similar functions are always connected to the same set of servers. For example, you might set up the following clusters:

- Cluster 1: Contains servers A and B. Used for Enterprise Portal users.
- Cluster 2: Contains servers C and D. Used for sales order entry.
- Cluster 3: Contains servers E and F. Used for batch processing.

Choose a load balancing configuration

There are two types of load balancing clusters for Microsoft Dynamics AX. You can create a cluster with one AOS instance that acts solely as a load balancer. You can also create a cluster with no specified load balancer.

Set up a cluster with a load balancer

If you set up a cluster with a load balancer, the load balancer AOS instance is dedicated to distributing the user load. The load balancer AOS instance does not accept client connections. A cluster that contains a load balancer AOS instance must also contain at least one AOS instance that is not a load balancer.

In this configuration, you must set client configurations to connect to the load balancer AOS instance. You can then add and remove other AOS instances from the cluster without needing to update client configurations.

When a client starts, it connects to the load balancer AOS instance. The load balancer AOS instance returns a list of active AOS servers in the cluster, sorted based on workload. The client tries connecting to the first AOS instance in the list. If that connection fails, the client attempts to connect to the second AOS instance in the list, and so on.

Set up a cluster without a load balancer

If you set up a cluster without a load balancer, each AOS instance acts as both a load balancer and an active AOS instance that accepts client connections.

In this configuration, client configurations must contain connection information for multiple AOS instances. If you need to remove an AOS instance from the cluster, you must update the client configurations that refer to that instance.

When a client starts, it sends a request to the first server that is listed in the client configuration. That server returns the list of the active AOS servers in the cluster, sorted based on workload. The client tries connecting to the first AOS instance in the list. If that connection fails, the client attempts to connect to the second AOS instance in the list, and so on.

Install additional AOS instances

Before you set up clusters, you must install additional instances of the AOS. Point all AOS instances to the same application file location and the same database. For more information about installing an AOS instance, see the Microsoft Dynamics AX Installation Guide.

New AOS instances are automatically added to the default cluster, which does not participate in load balancing. After installing a new AOS instance for load balancing, you must add it to a cluster other than the default.

Create a cluster

1. Open the **Cluster configuration** form (**Administration** > **Setup** > **Cluster configuration**).
2. Press CTRL+N to create a new cluster.
3. Enter a name and description for the cluster.
4. Press CTRL+S to save changes.

Add an AOS instance to a cluster

1. In the **Cluster configuration** form, click **Map AOS instances to clusters**.
2. Select an AOS instance.
3. If the selected AOS instance should act as a load balancer, select the **Load Balancer** option.

   Notes:

   - You can select the **Load Balancer** option for an AOS instance when the AOS instance is not running, if the instance has been started at least once.
   - If an AOS instance is used as a load balancer, it cannot be used as a batch server.
4. In the **Cluster name** field, select the appropriate cluster for the selected AOS instance.
5. Press CTRL+S to save changes.

Change client configurations

If the cluster uses a load balancer AOS instance, set client configurations to connect to the load balancer AOS instance. If the cluster does not use a load balancer AOS instance, add all AOS instances in the cluster to the client configurations.

Use the Microsoft Dynamics AX Configuration utility to change client configurations. For more information, see Managing configurations (Client).

Remove an AOS from load balancing

Before you remove an AOS instance from a load-balancing cluster, you must stop clients from connecting to that AOS.

Change client configurations

If the cluster uses a load balancer AOS and you are not removing the load balancer, you do not need to change client configurations to connect to a different server.

If the cluster does not use a load balancer AOS, or if you are removing the load balancer, you must change client configurations to connect to a different server.

Use the Microsoft Dynamics AX Configuration utility to change client configurations. For more information, see Managing configurations (Client).

Set the AOS instance to reject new clients

1. Open the **Online users** form (**Administration** > **Common Forms** > **Online users**).
2. On the **Server Instances** tab, select the AOS instance that you want to remove.
3. Click **Reject new clients**.

   Client sessions already in progress will not be disconnected, but no new connections will be accepted. When the number of clients connected to the AOS reaches 0, it is safe to remove the AOS from the cluster.

Remove the AOS instance from the cluster

1. Open the **Cluster configuration** form (**Administration** > **Setup** > **Cluster configuration**).
2. Click the tab **Map AOS instances to clusters**.
3. Select the AOS instance that you want to remove.
4. Click the tab **Cluster management**, and in the **Cluster name** field, choose the default cluster (**Non Load Balanced AOS Instances**). AOS instances in the default cluster do not participate in load balancing.
5. Press CTRL+S to save changes.

Uninstall the AOS instance

To completely remove the AOS instance from the environment, you must uninstall it. For more information about uninstalling an AOS instance, see Uninstall Microsoft Dynamics AX.

Set up Performance Monitor counters

You can use the Performance Monitor counters that are included with Microsoft Dynamics AX to help you monitor the usage of system resources. With counters you can collect and view real-time performance data about server resources such as processor and memory use, and about Microsoft Dynamics AX and Microsoft SQL Server resources such as locks and transactions.

The following table describes the counters for the Microsoft Dynamics AX Object Server performance object.

Counter	Description
Active Sessions	The number of currently active server sessions.
Number of Bytes Received by Server	The number of bytes received by the Application Object Server (AOS) instance since it started.
Number of Bytes Sent by Server	The number of bytes sent by the AOS instance since it started.

Counter	Description
Number of Client Requests	The number of client-to-server requests since the AOS instance started.
Number of Client Requests per Second	The number of client-to-server requests processed per second by the AOS instance.
Number of Server Requests	The number of server-to-client requests processed since the AOS instance started.
Total Sessions	The total number of active sessions since the AOS instance started.

The following table describes the counters available for the Microsoft Dynamics AX: Enterprise Portal performance object. All Enterprise Portal counters are .NET Business Connector counters. If you call the .NET Business Connector through another application, the same counters can be used.

Counter	Description
Number of Sessions	The number of currently active .NET Business Connector sessions.
Web Part Execution Time	The time in seconds taken to execute and render a Web Part.
Fatal Session Exceptions	The number of fatal .NET Business Connector session exceptions. For Enterprise Portal, this means that the page was not rendered. A Windows SharePoint Services error page was displayed to the user.
Nonfatal Session Exceptions	The number of nonfatal .NET Business Connector session exceptions. For Enterprise Portal, this means that the page was rendered, but some Web Parts on the page were not rendered.
Xpp Session Exceptions	The number of X++ .NET session exceptions.
Sessions Allocated	The total number of .NET Business Connector sessions allocated since AOS startup.
Sessions Disposed	The total number of .NET Business Connector sessions disposed of since AOS startup.
Session Allocation Rate	The number of .NET Business Connector sessions allocated per second.

You may also want to monitor counters for the AOS process (Ax32Serv), such as CPU usage, memory usage, handle counts, and thread counts.

Add counters

1. Open the **Performance** window (**Start** > **Administrative Tools** > **Performance**).
2. Click **Add** or press CTRL+I.
3. In the **Add Counters** dialog box, verify that the correct server name appears.
4. Select **Select Counters from Computer**.
5. In the **Performance Object** list, select an object to add counters for, such as **Microsoft Dynamics AX Object Server**.
6. Select all counters for the object, or select individual counters.
7. Click **Add**, and then click **Close**.

Set up an alert

1. On the navigation tree of the **Performance** window, expand **Performance Logs and Alerts**.
2. Right-click **Alerts**, and then click **New Alert Settings**.
3. In the **New Alert Settings** dialog box, type a name for the new alert, and then click **OK**.
4. On the **General** tab of the dialog box for the new alert, add a comment, and then click **Add** to add a counter to the alert.

   All alerts must have at least one counter.
5. In the **Add Counters** dialog box, select a Microsoft Dynamics AX object from the **Performance Object** list, and then select a counter from the **Select counters from** list.
6. To add the counter to the alert, click **Add**. You can continue to add counters, or you can click **Close** to return to the dialog box for the new alert.
7. In the new alert dialog box, select either **Over** or **Under** in the **Alert when the value is** list, and then enter a threshold value in **Limit**.
8. Click **Apply**.

   The alert is generated when the value for the counter is more than or less than the threshold value (depending on whether you selected **Over** or **Under**).
9. In the **Sample data every** boxes, set the sampling frequency.
10. On the **Action** tab, set actions to occur every time the alert is triggered.
11. On the **Schedule** tab, set the start and stop schedule for the alert scan.

Tracing

Microsoft Dynamics AX includes a tracing tool to help you create a performance baseline for your Microsoft Dynamics AX servers. A performance baseline can help you understand when server performance is slow or when you might need to make changes in your configuration to manage server capacity.

The topics in this section describe how to set tracing options in Microsoft Dynamics AX and how to read trace files.

Set tracing options

Microsoft Dynamics AX provides multiple locations to set tracing options for server and client activity. You can set traces:

- In the Microsoft Dynamics AX Server Configuration Utility on the computer running the Application Object Server (AOS) instance.
- In the Microsoft Dynamics AX Configuration Utility on a client or for an instance of Business Connector that is running non-interactively.
- Within Microsoft Dynamics AX, in the **Tools** > **Options...** dialog box, on the **Development** and **SQL** tabs.

  Note:

  This option is not available unless you also select **Allow client tracing on Application Object Server instance** in the Server Configuration Utility.

In general, we recommend that you use these tools to help you trace in the following scenarios:

Scenario	Tool
Monitoring general performance	Trace from the Server Configuration Utility on a computer running an AOS instance.
Standard troubleshooting	Trace from the Server Configuration Utility on a computer running an AOS instance.
Debugging code	Trace from the Configuration Utility on the client - or - For line-by-line tracing only, use the options in the **Tools** > **Options** dialog box. This may option may degrade system performance.
Create an application profile	Trace from the Configuration Utility on the client.
Deep troubleshooting	Trace from the Server Configuration Utility on the AOS.

Considerations for tracing from the configuration utilities

When you are setting up tracing from the Configuration Utility or Server Configuration Utility, be aware of the following:

Note:

Tracing is processing-intensive and space-intensive - We recommend that you do not turn tracing on for more than one client at a time.

- On Windows Vista and Windows Server 2008 operating systems, only Administrators and SYSTEM accounts can use tracing. To run tracing on a client, you must choose the option run as Administrator. To run tracing on the server, the AOS account must be a SYSTEM account.
- The log directory cannot be changed - The log is always installed to *installationdirectory\log*. Restrict access to the directory to administrators and the AOS account (the domain account or Network Service account associated with the AOS service).
- Trace files are stored in binary format, and can be read with the Microsoft Dynamics AX Tracing Utility. Set tracing options (Server)

1. Open the Server Configuration utility (**Start** > **Administrative Tools** > **Microsoft Dynamics AX Server Configuration Utility**).
2. Verify that the currently selected Application Object Server (AOS) instance and configuration are the ones you want to modify.

3. On the **Tracing** tab, evaluate the type of tracing you need to do, and choose settings.

To do this	Select these options
Monitor general performance on a production server	**RPC round trips to server**
Perform standard troubleshooting	**RPC round trips to server** **X++ method calls, Number of nested calls: 4** **SQL statements** **Allow client tracing on Application Object Server instance**
Debug code	**RPC round trips to server** **X++ methods** **SQL statements** **Row fetch summary (count and time)** **Allow client tracing on Application Object Server instance**
Deep troubleshooting	All options. Performance may be degraded while all tracing options are on.

4. On the **Tracing** tab, click **Start trace**. If the AOS Windows service is running, the trace starts within 15 seconds. If the service is stopped, the trace starts the next time the service is started.

Set tracing options (Client)

1. Open the Configuration utility (**Start** > **Control Panel** > **Administrative Tools** > **Microsoft Dynamics AX Configuration Utility**).
2. Verify that the currently selected configuration is the one you want to modify.
3. On the **Tracing** tab, evaluate the type of tracing you need to do, and choose settings.

To do this	Select these options
Monitor general performance on a production server	**RPC round trips to server**
Perform standard troubleshooting	**RPC round trips to server** **X++ method calls, Number of nested calls: 4** **SQL statements**
Debug code	**RPC round trips to server** **X++ methods** **SQL statements** **Row fetch summary (count and time)** **Enable method tracing to client desktop**
Deep troubleshooting	All options. Performance may be degraded while all tracing options are on.

4. To start tracing once you have set the options you want, close the **Configuration Utility**, and restart your Microsoft Dynamics AX client.

View a trace file

Files from traces are saved to the following locations:

Type of trace	Location
AOS trace files	*AOS computer Log\<servername>_<timestamp>.trc*
AOS settings and SQL settings triggered from client (**Allow client tracing on Application Object Server instance** is selected)	*AOS computer Log\<Username>_<ClientIP>_<sessionID>_<server>.trc*
Client method trace triggered from client (**Enable method tracing to client desktop** is selected)	*Client computer log\<Username>_<ClientIP>_<sessionID>_<client>.trc*

A new file is created each time tracing is started, or when a new day starts.

Note:

If you are running frequent traces, be sure to remove or archive unneeded trace files often.

Troubleshooting tracing

This section provides information on troubleshooting issues encountered while tracing.

In the Configuration Utility it appears that a trace is running, but when I look in Windows, the trace is not running

When a trace file reaches its size limit, it is stopped by Windows. The Configuration Utility interface does not synchronize with Windows until you click **Stop trace**.

When I run more than one client tracing session at a time, my system slows down

Tracing is processing-intensive and space-intensive - we recommend that you do not turn tracing on for more than one client at a time.

Reading trace files

Trace files store Microsoft Dynamics AX information gathered by turning on tracing in the configuration utilities.

- Client log and trace files are stored at: Program Files\Microsoft Dynamics AX\50\Client\Log
- Server log and trace files are stored at: Program Files\Microsoft Dynamics AX\50\Server\company_name\Log

When you trace method calls, the values returned are multiples of 2.

To determine the actual call depth, divide the value by 2.

Note:

The call depth is reset to 0 each time a call crosses a tier (calls from the client to the server, or from the server to the client).

See Also

Set tracing options

Setting processor affinity

If you use a multi-processor server for Microsoft Dynamics AX, you can specify which processor hosts the AOS service. This is called processor affinity, and it can help improve server performance.

1. Open the **Microsoft Dynamics AX Server Configuration** utility (**Start** > **Administrative Tools** > Microsoft Dynamics AX 2009**Server Configuration**).
2. Click **Manage** > **Create configuration**.
3. Enter a name, select where to copy the configuration from, and click **OK**.
4. Click the **Performance** tab.
5. In the **Processor Affinity** section, select **Custom**, and then select which processor should host the AOS service.
6. Click **OK**.

Tune database settings

You may want to tune the database settings for Microsoft Dynamics AX to improve performance. Before changing settings, trace the usage of your Microsoft Dynamics AX database to ensure that you have clear understanding of performance under the current settings. To trace Microsoft Dynamics AX database performance, use:

- Tracing from the Microsoft Dynamics AX Server Configuration Utility. For more information, see Set tracing options.
- Windows Performance Monitor, using Microsoft Dynamics AX Object counters.

Test all tuning changes before implementing them in a production environment. In a test or development environment, make a single change and then test your system's performance before making another change.

Tune connections

The following table lists common connection issues, and some adjustments to try in the Server Configuration Utility.

Symptom	Adjustments to try
Results for common queries are returned slowly.	Increase the **Maximum buffer size** value.
Results for ad hoc queries are returned slowly.	Check to see that the appropriate indexes are in place. For the most recent guidance about indexing, check Microsoft Dynamics AX Online.
Transactions are failing frequently.	Decrease the **Transaction retry interval** value.
Data grids for commonly used tables draw slowly.	Increase the **Array fetch ahead** value.

Tune queries

If queries in the system are running slowly, you may want to change settings for literals, string functions, or hints. Microsoft Dynamics AX no longer uses literals by default in form and report queries, or in complex-join queries.

Adjust the use of literals

Microsoft Dynamics AX may pass either parameters (placeholders) or literals (actual values) in queries.

- Parameters allow Microsoft Dynamics AX and the database server to reuse the query when search values change. They are preferred for high-frequency queries.
- Literals allow the database server to optimize the query for a specific piece of information. This provides an optimal query for that piece of information, but the database server must perform the optimization for every query executed. Literals may be used for long running queries such as complex joins.

A developer can override the default use of literals by specifying parameters in their code, or an administrator can override the use of literals in the Server Configuration Utility.

Symptom	Adjustments to try	Anticipated effect
Long-running queries run slowly.	Review the query plan statements sent to SQL Server and consider taking corrective action. Using literals may be one solution. Select **Use literals in join queries from forms and reports**. Select **Use literals in complex joins from X++.**	Long-running queries pass literals to the database. Processing time for long-running queries should go down.

Adjust the use of autogenerated string functions

Microsoft Dynamics AX embeds some string functions in `SELECT` statements automatically. String functions are included to support:

- Treating uppercase and lowercase versions of the same text as the same text (single case) for Oracle installations.
- Left justification or right justification.

When a string function is included in a query, the optimizer may have to choose a less-than-optimal access plan, such as a table scan, for retrieving data. If customers do not require the use of mixed case outside Microsoft Dynamics AX and do not use left justification or right justification, these functions are not required and should be turned off. To improve performance, we recommend that all values be stored left-aligned by default.

Adjust the use of hints

In Microsoft Dynamics AX, you can allow developers to override the index selected by the query optimizer. In most situations, allowing the query optimizer to select an index for a query results in improved performance.

If queries include `INDEX` hints and are running more slowly than expected, clear the **Allow INDEX hints in queries** option.

Changes in the use of hints

If you have upgraded to Microsoft Dynamics AX, the queries in your system may contain outdated Microsoft SQL Server hints. Configuration commands are no longer available to globally enable or disable many of the hints from previous versions. If hints are explicitly specified in an X++ statement, they are added to the SQL Server query that is generated. Otherwise, they are not added.

The following changes have also been made:

- The `OPTION (FAST)`, `LOOP`, and `FORCE ORDER` hints are not applied by default, but are applied if explicitly specified in X++.
- A `FIRSTONLY` hint in X++ is translated into the addition of a `TOP 1` statement to the SQL Server query.
- `FASTFORWARD` cursors are used for all user queries unless a cursor has been marked as `FOR UPDATE`.
- `FOR UPDATE`, `NOLOCK`, and `READPAST`, hints are added to statements depending on the type of the cursor that an X++ query has produced. No interface is available to modify these hints.

Change the concurrency mode

Concurrency settings enable you to reduce locking conflicts on your system. For more information, see the following topics in the Microsoft Dynamics AX 2009 Developer Documentation.

- Performance optimizations: Database design and operations
- Transaction integrity
- Exception handling
- Select statement syntax
- Table properties

Manage database logs

Database logs contain sensitive data. By default, any user who has database access can query a database log by using Business Connector, X++, or alerts, or by using direct database access. To protect data, restrict permissions on the sysdatabaselog table. For more information, see "Manage table and field access" in the **System and Application Setup** Help, available from the Microsoft Dynamics AX Help menu.

Consider carefully which tables you select for database logging. For example, logging changes in transaction tables where there are often many changes has a negative impact on overall system performance. To limit log entries and to improve performance, select specific fields to log instead of entire tables. For individual fields, only updates can be logged.

Modifying or uninstalling Microsoft Dynamics AX

This section contains information about using the Setup wizard to add or remove individual Microsoft Dynamics AX components. It also includes information about using Windows to remove Microsoft Dynamics AX.

This section contains the following topics:

Add or remove individual Microsoft Dynamics AX components

This section describes how to use the Setup wizard to add or remove individual Microsoft Dynamics AX components.

Important:

Some components cannot be removed using Setup. Databases, log files, and application files must be removed manually. For more information, see Uninstall Microsoft Dynamics AX.

Add individual Microsoft Dynamics AX components

1. Start Microsoft Dynamics AX Setup.
2. Step through the initial wizard pages.
3. On the **Modify Microsoft Dynamics AX installation** page, click **Add or modify components**. Then click **Next**.
4. On the **Add or modify components** page, select the components you want to add and then click **Next**.

   If a component is not available, it means that an instance is already installed, and only one instance of that component can be installed on a computer.

   Note:

   To add or remove Help languages, select the **Client** component.

5. Step through the wizard pages and enter the required information for the components you are installing.

   For more information about installing a specific component, see the Microsoft Dynamics AX Installation Guide.

6. On the **Ready to install** page, click **Install**.

Remove individual Microsoft Dynamics AX components

1. Start Microsoft Dynamics AX Setup.
2. Step through the initial wizard pages.
3. On the **Modify Microsoft Dynamics AX installation** page, click **Remove components**. Then click **Next**.
4. On the **Remove components** page, select the components you want to remove and then click **Next**.

   If a component is not available, no instance of that component is installed on the computer, or that component cannot be removed by using Setup. The following components can be removed if they have been installed on the local computer:

   - Application Object Server (AOS) instances

     Note:

     You can also uninstall AOS instances by using the Add or Remove Programs control panel. Before you remove an AOS instance, use the **Microsoft Dynamics AX Configuration** utility to point all clients to a valid AOS instance, or update the shared configuration file. When you remove an AOS instance, it is not automatically removed from the list of batch and load balancing servers. After you uninstall an AOS instance, you must manually delete it using the **Server configuration** form or the **Cluster configuration** form.

   - Client
   - Role Centers and the Enterprise Portal framework

     Uninstalling Role Centers and the Enterprise Portal framework removes the files installed by Setup, but does not remove Web sites. Delete Web sites using SharePoint Central Administration.

   - Workflow
   - Reporting extensions

     Uninstalling the reporting extensions removes the files installed by Setup, but does not remove Reporting Services objects such as data sources, reports, and report models. To remove these objects, use the Reporting Services administration tools.

- Analysis extensions

  Uninstalling the analysis extensions does not delete SQL Server Analysis Services objects, such as databases, cubes, and models. To remove these objects, use the Analysis Services administration tools.
- Debugger
- Enterprise Portal developer tools
- Reporting tools
- .NET Business Connector
- AIF Web services
- BizTalk adapter
- Synchronization proxy for Microsoft Project
- Synchronization service for Microsoft Project

5. Step through the wizard pages and enter the required information for the components you are removing.
6. On the **Ready to uninstall** page, click **Remove**.

Uninstall Microsoft Dynamics AX

This topic explains how to uninstall Microsoft Dynamics AX by using **Add or Remove Programs** on the Control Panel.

Databases, log files, and application files must be removed manually. Information about how to manually remove components is also included in this topic.

Uninstall components by using Add or Remove Programs

Use this procedure to remove Microsoft Dynamics AX components.

1. Open **Add or Remove Programs**. (**Start** > **All Programs** > **Control Panel** > **Add or Remove Programs**).

2. Select the component that you want to remove, and then click **Remove**. The components that are listed in the following table can be removed.

Option	Removes
Microsoft Dynamics AX Components	Selecting this option removes the following components: • Client • Role Centers and the Enterprise Portal framework • Workflow • Reporting extensions • Debugger • Enterprise Portal developer tools • Reporting tools • .NET Business Connector • AIF Web services • BizTalk adapter • Synchronization proxy for Microsoft Office Project • Synchronization service for Microsoft Office Project This option removes all components that are installed on the local computer. You cannot select to remove individual components.
Microsoft Dynamics AX Client Help Files	Select this option to remove all Help files in all installed languages. You should not remove Help files unless the client is also being removed.
Microsoft Dynamics AX Object Server (*instance name*)	Select this option to remove a specific Application Object Server (AOS) instance. Before you remove an AOS instance, use the Microsoft Dynamics AX Configuration utility to point all clients to a valid AOS instance, or update the shared configuration file. When you remove an AOS instance, it is not automatically removed from the list of batch and load balancing servers. After you uninstall an AOS instance, you must manually delete it by using the **Server configuration** form or the **Cluster configuration** form.

3. A message box asks you to confirm that you want to uninstall. To proceed, click **Yes**.

Remove remaining components manually

If you choose to remove an entire installation, some components remain after **Add or Remove Programs** is finished. The following table provides more information about removing components manually.

To remove this	Do this
Application files	Delete the application file directory from the location that you installed it to.
Database and log files	Use database server administration tools to delete the database and log files.
Role Centers and the Enterprise Portal framework	Delete Web sites by using SharePoint Central Administration.
Reporting extensions	• Delete SQL Server Reporting Services objects, such as data sources, reports, and report models by using Reporting Services. • Delete the contents of the Microsoft Dynamics AX report folder.
Analysis extensions	Delete SQL Server Analysis Services objects, such as databases, cubes, and models by using Analysis Services.