

AX to Robots Migration Guide

Version 15.1

Published Monday, January 31, 2022



This page intentionally left blank

Table of contents

Overview of the migration process	5
Who is this guide for?	5
Performing the migration: our recommended approach	6
Prerequisites (before you start the migration)	7
Recreate your AX repository structure in Robots	11
Suggested guidelines	11
Comparing how AX and Robots structure content	12
Create robots to structure content and automated processes	14
A word about the AX Working directory and the Library	14
What to do next	15
Migrate the scripts	17
Backward compatibility of Robots	17
Export the scripts from AX	17
Do any necessary preparatory work	18
Commit (upload) the scripts to Robots	19
Replicating analytic chains in Robots	19
What to do next	22
Migrate the data	23
Recreating or migrating Analytics data tables	23
Recreating data tables	24
Migrating data tables	26
Create linked tables in Robots	32
Uploading or migrating non-Analytics related files	32
What to do next	34
Recreate automated AX processes in Robots	35
Create a robot task for each automated process	35
Test a task by running it ad hoc	36
Activate a script version to production	37

Create and schedule a robot task in production 38

What to do next39

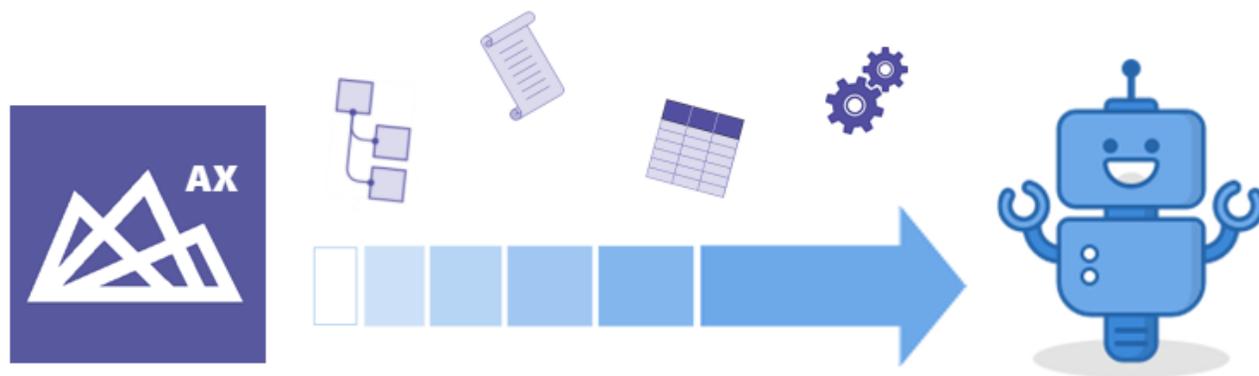
Post-migration activities 41

Additional information about migrating 43

The benefits of migrating 43

How much time and effort does the migration take? 45

Overview of the migration process



This guide explains how to migrate from Analytics Exchange (AX) to the Robots app in the HighBond platform. The guide covers setting up the necessary structure in Robots, migrating scripts and data, recreating automated AX processes in Robots, and post-migration activities. It also summarizes the benefits of migrating.

Migrating from AX to Robots requires a number of steps, but if you follow the guide the overall process is not difficult.

Who is this guide for?

This guide assumes your organization currently uses Analytics Exchange for running automated data analysis processes and that you are ready to migrate those processes to the Robots app in the HighBond platform. The guide also provides valuable information for organizations that are considering migrating.

The guide assumes that you are going to use an on-premise Robots Agent to analyze and process data on your own network rather than use the option of a cloud-based Robots Agent. Using a cloud-based agent is not practical for most, if not all, existing AX customers.

For more information about the considerations involved with migrating, see "Additional information about migrating " on page 43.

Performing the migration: our recommended approach

1. Before you start the migration, you must complete the prerequisite tasks and activities explained in "Prerequisites (before you start the migration)" on the facing page.
2. Once the prerequisites are in place, we recommend performing the migration in four stages:
 - a. "Recreate your AX repository structure in Robots" on page 11
 - b. "Migrate the scripts" on page 17
 - c. "Migrate the data" on page 23
 - d. "Recreate automated AX processes in Robots" on page 35
3. After completing the migration, review "Post-migration activities" on page 41 and perform post-migration tasks as required.

Quickly familiarize with the end-to-end process

The main steps in the end-to-end migration process are outlined below. Some of the steps may not apply to your organization, depending on how you have set up automated processes in AX.

Migration stage	Steps	Applies to:
Recreate your AX repository structure in Robots	<p>Create empty robots in the Robots app in HighBond.</p> <p>A robot is a container that houses uploaded analytic scripts, any Analytics data tables, and any related files. The robot is the object that you configure to carry out scheduled, automated tasks.</p>	All organizations
Migrate the scripts	Export your organization's analytic scripts from AX to an intermediary location	All organizations
	Configure scripts to replicate AX analytic chains in Robots	Organizations that use analytic chains in AX
	Commit (upload) the scripts to an appropriate robot in Robots	All organizations
Migrate the data	If Analytics data tables in AX do not contain historical data, recreate the tables in Robots	Organizations that use Analytics data tables in AX (tables created with the <code>//DATA</code> tag)
	If Analytics data tables in AX contain historical data that you need to retain, use the Diligent migration scripts to migrate the tables from	Organizations that use Analytics data tables in AX (tables created with the <code>//DATA</code> tag)

Migration stage	Steps	Applies to:
	AX to Robots	
	Upload or migrate non-Analytics related files	Organizations that use non-Analytics related files in AX (files contained in Related Files folders)
Recreate automated AX processes in Robots	Re-link tables using shared and linked tables in Robots	Organizations that use master and linked tables in AX
	Set up automated processes in Robots: <ol style="list-style-type: none"> In development mode, create a robot task for each automated process. Test a task by running it ad hoc. Once a task is running correctly, activate the scripts to production. In production, create and schedule a robot task for the automated process. 	All organizations

Take an iterative approach

We also recommend that you take an iterative approach to the migration, especially if your organization has a large number of automated processes currently running in AX:

- **Run in parallel** – Run AX and Robots in parallel for a period of time.
- **Pilot the migration** – Pilot the migration by initially replicating in Robots only one or two automated processes. Recreate the structure, and migrate the scripts and data, for only this limited number of processes.
- **Compare output results** – Compare the output results from the same processes running in AX and Robots. Once you are satisfied that the initial processes are working correctly in Robots, you can recreate the structure, and migrate the scripts and data, for the remainder of your automated processes.

Prerequisites (before you start the migration)

In preparation for the migration, you must complete several tasks and activities.

Task/Activity	Details
Required Verify your ACL Robotics subscription and	Work with your Diligent account representative to make sure that your organization has the correct type of ACL Robotics subscription, with enough licenses for all your intended users. <ul style="list-style-type: none"> ◦ Required ACL Robotics edition: Enterprise

Task/Activity	Details
permissions	<p>Note</p> <p>Because of functional restrictions, ACL Robotics Professional Edition is not suited as a replacement for AX.</p> <p>Make sure that you have the HighBond administrator permissions required to perform the migration:</p> <ul style="list-style-type: none"> ◦ Launchpad user type: System Admin ◦ Subscription: Professional <p>Note</p> <p>The assumption is that you are also an AX Server administrator.</p>
<p>Required</p> <p>Confirm your Launchpad and Robots administrator access</p>	<p>Once the ACL Robotics subscription and permissions are in place, sign in to HighBond, and make sure you can access the following admin-only locations:</p> <ul style="list-style-type: none"> ◦ In Launchpad, the page for adding and managing HighBond users: Options > Users ◦ In Robots, the global settings: <ul style="list-style-type: none"> • Settings > Agent management • Settings > User management
<p>Required</p> <p>Provision an additional Windows server</p>	<p>Provision an additional Windows server for your network.</p> <p>The additional Windows server is for installing an on-premise Robots Agent, the data processing component of Robots.</p> <p>For server and operating system requirements, see On-premise Robots Agent system requirements.</p> <p>If you intend to use multiple on-premise agents for increased processing capacity and load-balancing, you need to provision an additional Windows server for each additional Robots Agent. You can choose to perform this portion of the migration later, after you have successfully migrated automated processes from AX to the first Robots Agent server. If your organization currently uses additional servers for AX Engine Node, you can probably repurpose those servers for installation of additional instances of the Robots Agent.</p> <p>Caution</p> <p>Do not install the Robots Agent on the same Windows server as AX Server. Resource conflicts and unexpected behavior can occur and troubleshooting can be difficult.</p>
<p>Required</p> <p>Install the Robots Agent on the Windows server</p>	<p>Install and configure the on-premise Robots Agent by following the instructions in these topics:</p> <ol style="list-style-type: none"> a. On-premise Robots Agent security b. Installing or upgrading a Robots Agent c. Character encoding used by the Robots Agent d. Configuring a Robots Agent
<p>Optional</p> <p>Install one or more additional instances of the Robots Agent</p>	<p>Use the same process that you used for installing the first instance of the Robots Agent.</p> <p>For more information, see Installing or upgrading a Robots Agent.</p>

Task/Activity	Details
<p>on one or more additional Windows servers</p>	<p>Tip</p> <p>Focus on installing a single Robots Agent on a single server first. Once this first installation is operating successfully, you can set up a multi-agent cluster.</p>
<p>Strongly recommended</p> <p>Confirm that the on-premise Robots Agent is working</p>	<p>See Test the Robots Agent.</p>
<p>Required, if applicable</p> <p>Install Direct Link on the server where the Robots Agent is installed</p>	<p>If you have been using Direct Link with AX to access SAP data, do one of the following:</p> <ul style="list-style-type: none"> ◦ Install Direct Link to work with the Robots Agent ◦ Transition to using the ACL Connector for SAP (the SAP Connector) instead of Direct Link <p>See Connecting to SAP.</p> <p>Note</p> <p>If you take an iterative approach to the migration, and some of your automated processes do not require importing SAP data, you could install Direct Link as a post-migration task.</p>
<p>Strongly recommended</p> <p>Familiarize with the Robots app</p>	<p>The migration will go more smoothly if you first take the time to familiarize with the Robots user interface and the workflow.</p> <p>The easiest way to get familiar with Robots is to do the lightweight tutorial: Getting started with ACL Robotics.</p>

This page intentionally left blank

Recreate your AX repository structure in Robots

Important

Only proceed with the migration if you have completed the prerequisite tasks. For more information, see "Prerequisites (before you start the migration)" on page 7.



The best approach for migrating scripts and data from AX to Robots is to set up the structure of the destination before you start moving anything. If you know exactly where each file is going before you move it, the migration will proceed more smoothly. The greater the amount of content you intend to move, the more important a well-organized destination becomes.

Suggested guidelines

Both AX and Robots offer a certain amount of flexibility when structuring your repository content. Ultimately, you are free to structure content in whatever manner best suits your organization's business processes.

- **One AX folder could equate to one robot**

If you have been using a single AX folder for the scripts and tables from a single Analytics project, you can continue that one-to-one relation. Instead of an AX folder, you will now use a container in Robots called a **robot**. The Robots method for structuring content is explained in more detail below.

- **An iterative approach**

You do not need to recreate your entire AX repository structure all at once – although you can if you are ready to move forward. If you are piloting the migration by initially replicating only one or two automated processes in Robots then you only need to recreate the structure for those processes.

- **Consider dependencies**

Do some of your automated processes depend upon other processes completing first? For example, you may have a process that imports data from a data source, and then subsequent processes that analyze the data. If you take an iterative approach, make sure you follow a logical sequence and replicate any parent processes first, before replicating the dependent processes.

- **Do some housecleaning**

The migration is an ideal opportunity to rethink the organization of your repository. Is the organization confusing or less effective in some areas? Should you merge the content of some folders, and split the content of others?

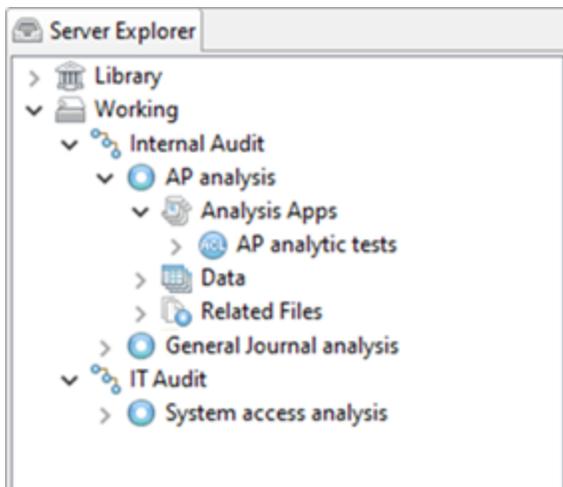
- **Do not migrate content you no longer need**

Do not migrate any old content that you no longer require.

Comparing how AX and Robots structure content

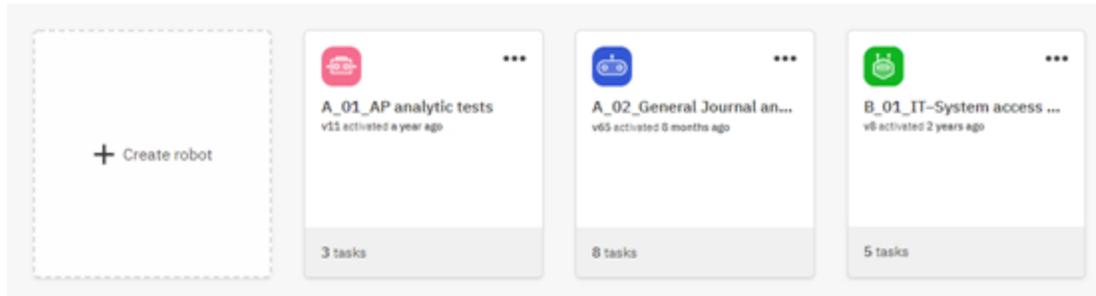
AX method

AX uses a traditional expandable tree view with a multi-level hierarchy for structuring content. Collections and folders are organizational containers that comprise the second and third levels of the hierarchy. Folders contain analytic scripts, Analytics data tables, and non-Analytics related files such as Excel spreadsheets.



Robots method

Currently, Robots use a single level of organizational containers called robots, arranged in a grid view on a dashboard. Each robot contains a set of tabs that organize the same content as that stored in AX.



Equivalent content and process locations in AX and Robots

The table below maps the locations of AX content and processes to the equivalent locations in Robots.

Content or process	AX location	Robots location
repository	Server Explorer tree view	dashboard grid view
organizational container	collection > folder	robot
analytic script	folder > Analysis Apps > analysis app	robot > Script versions tab
Analytics data table	folder > Data	robot > Input/Output tab > Other tables section or Linked tables section
Related files such as Excel and *.csv	folder > Related Files	robot > Input/Output tab > Files section
Run or schedule analytic job or robot task	Run or Schedule dialog box	robot > Tasks tab
View status of analytic job or robot task	Job Status window > Analytic Jobs tab	robot > Task runs tab
View details of output results	Properties window > Analytic Job Details tab	robot > Task runs tab > Task run details side panel

Create robots to structure content and automated processes

Regardless of how you structure your organization's content in Robots you need to create a number of robots to contain the content and organize automated processes.

You can start by creating one or two test robots, or you may be ready to build a full collection of robots capable of housing all the content and processes in your current AX repository. For now, the robots will be empty because you have not yet migrated scripts and data from AX.

For detailed information about creating robots, see [Creating and managing a robot](#).

1. From the Launchpad homepage (www.highbond.com), under **Automation**, click the Robots app.
You need to sign in to HighBond if you are not already signed in.
2. From the dashboard in Robots, click **Create robot**.
3. Select an icon and enter a name for the robot.

Note

Do not use currency symbols anywhere in the robot name, for example: "\$", "€".

4. Optional. In the **Description** field, describe the robot so that other users understand what the robot does.
5. Click **Create robot**.

The robot is created and added to the dashboard.

6. Create the additional robots that you require to build out your organizational structure in Robots.

If you are taking an iterative approach to the migration, you can create just the robots that you need at this point. Creating new robots is quick and easy and you can create additional robots at any time.

A word about the AX Working directory and the Library

The AX repository has two separate areas, the **Working** directory and the **Library**, that allow you to segregate working content and more permanent, admin-controlled content. Robots does not have a similar high-level division. Instead, you can configure one or more robots to restrict access to Robots Admins only, and any regular users that you specifically designate. For users without access to a

restricted robot, the robot does not even appear, although you can set up shared tables to provide access to data in a restricted robot.

For more information, see [Robots app permissions](#) and [Sharing and linking data tables](#).

What to do next

Once you have created the robots to contain one or more automated processes you are ready to start migrating scripts: "Migrate the scripts" on page 17.

This page intentionally left blank

Migrate the scripts

Important

Only proceed with migrating the scripts if you have completed the previous stage of the migration. For more information, see "Recreate your AX repository structure in Robots" on page 11.



For each automated process in AX that you intend to recreate in Robots you need to migrate the associated scripts. You need to migrate all analytic scripts and any subscripts or helper scripts.

You cannot migrate scripts directly from AX to Robots. You need to export the scripts from AX to an intermediary location where Analytics is installed, such as your own computer. After completing any preparatory work that may be necessary, you **commit** (upload) the scripts from the intermediary location to Robots.

Backward compatibility of Robots

In general, Robots is designed to be as backward compatible as possible with AX. Most AX scripts should work in Robots without requiring any modification.

Any scripts that depend on AX-specific design will need to be adjusted. For example:

- scripts that access the AX operational database to read job schedules or logs
- scripts that include absolute paths to server resources that are no longer valid on the Robots Agent server

If you need assistance with adjusting analytic scripts for use in Robots, contact Diligent Support.

Export the scripts from AX

To begin the migration, use the **Export** option to export scripts from AX. Scripts are exported in the analysis app that contains them. Each exported analysis app becomes a separate Analytics project in

Migrate the scripts

the export destination.

This procedure assumes that you are using your own computer as the intermediary location.

1. Create a folder on your computer for storing exported scripts.
2. In the AX **Server Explorer**, select the analysis app containing the scripts that you want to export.

You can **Ctrl+click** to select multiple analysis apps in the same **Analysis Apps** parent folder, or select the parent folder to export all the analysis apps in the folder.

3. Select **File > Export**.
4. In the **Export** dialog box, make sure both options are **not selected**:
 - **Work with the exported file(s)?**
 - **Export data files along with selected definitions?**
5. Click **Browse**, select the folder that you created for storing scripts, and click **OK**.
6. Click **OK** to export the analysis app or apps.

The scripts are exported in their analysis app to the specified location. Each analysis app is converted to an Analytics project.

7. Repeat the process for any additional scripts that you are ready to migrate.

Linked analytic scripts

Unlike AX, Robots does not support linked analytic scripts. If you use linked analytic scripts in AX, export the master analytic script and then commit it to each robot in which it is required.

Do any necessary preparatory work

Following the export from AX, scripts are ready to be committed to Robots if:

- all the scripts associated with an automated process are in a single Analytics project
- all analytic scripts that must execute sequentially in an automated process:
 - use an appropriate alphanumeric naming scheme
 - if implemented, have an appropriate analytic type

Analytic chains

Unlike AX, Robots does not support analytic chains. If you use analytic chains, you may need to do some preparatory work before scripts are ready to commit to Robots. For more information, see "Replicating analytic chains in Robots" on the facing page.

Commit (upload) the scripts to Robots

Once all the scripts associated with an automated process are contained in a single Analytics project, and appropriately named, you can commit the scripts to Robots.

Committing scripts uploads them from a local environment to the cloud-based Robots app. We use the word "commit" rather than "upload" because each time that you commit scripts to the same robot you create a new version of the scripts in the script history.

For detailed information, see [Committing scripts \(uploading\) from Analytics to a robot](#).

1. Open the Analytics project with the scripts that you want to commit.
2. From the Analytics main menu, select **File > Commit Scripts**.

If an error message appears, there may be a problem with the analytic header, or the script syntax, in one or more of the scripts in the project.

For more information, see [Script development workflow in Analytics and Robots](#).

3. If required, in the **Select Destination** dialog box, double-click the appropriate HighBond instance.

The **Robot Collection** appears with the list of existing robots.

4. In the list of robots, select the robot that you want to commit the scripts to and click **OK**.

The Analytics project that contains the scripts, and the existing robot, are now associated so that subsequent commits do not require that you manually locate the robot.

5. Enter a short commit message that describes the committed scripts, and click **OK**.

The scripts are committed to development mode in the robot.

6. In the **Commit Scripts Successful** dialog box, click the second link to inspect the newly committed scripts in the Robots app.

The **Script versions** tab opens with the committed version of the scripts.

7. Select the script version (**vn**).

The **Version details** side panel opens. The names of the individual analytic scripts and auxiliary scripts are listed beneath **Script version changes**.

8. In Analytics, click **OK** to exit the **Commit Scripts Successful** dialog box.

Replicating analytic chains in Robots

Note

This section only applies if you use analytic chains in AX.

Migrate the scripts

Unlike AX, Robots currently does not support assembling analytic scripts from anywhere in the repository into an analytic chain that executes sequentially. To implement sequential execution of a series of analytic scripts in Robots, you need to do two things:

- contain the analytic scripts in a single robot, and create a single task to execute the scripts
- use script names, and analytic type, to create a script sequence

Contain the analytic scripts in a single robot

Contain the analytic scripts that you want to execute sequentially in a single robot.

To contain the scripts in a single robot requires that you first contain all the related scripts in a single Analytics project. You commit the scripts as a single bundle called **a script version** from the Analytics project to the robot.

Once the scripts are in a single robot, you create a single task to execute the series of analytic scripts.

Contain associated scripts in a single Analytics project

If the scripts associated with a single automated process are distributed between two or more Analytics projects, you must unite the scripts in a single project before committing them to Robots. This situation could arise if analytic scripts were part of an analytic chain in AX.

1. Open the Analytics project that you want to use to contain all the scripts associated with an automated process.
2. In the **Overview** tab of the **Navigator**, right-click the project name at the top of the tree view and select **Copy from another Project > Script**.
3. In the **Locate Project File** dialog box, locate and select the Analytics project that you want to copy the script or scripts from and click **Open**.
4. In the **Import** dialog box, complete any of the following steps to add one or more scripts to the **To project_name** list:
 - Double-click a script.
 - **Ctrl+click** multiple scripts and then click the right-arrow button.
 - Click **Add All** to add all the scripts.
5. Click **OK** to copy the script or scripts into the destination project.

If a script with the same name already exists in the project, the copied script is given an incrementing numeric suffix.

6. Repeat the process for any scripts in other projects that need to be included in the destination project.

Use script names, and analytic type, to create a script sequence

Script names

Analytic scripts are sorted alphanumerically by name in a robot task, and execute in order, so name the analytic scripts in a way that creates the execution sequence you want.

For example, the first script in the sequence could use the prefix A_01, the second script could use A_02, or B_01, and so on. A prefixing system makes subsequent insertion of additional scripts easier. Subscripts or helper scripts are not subject to the sequential naming requirement.

Note

You cannot rename scripts in Robots, so make sure all scripts are correctly named in Analytics before you commit them to Robots.

Analytic type

As an additional aid for sequencing, you can optionally specify a type for analytic scripts. Specifying a type groups the scripts into separate areas in a task. The areas are ordered as follows, and execute in order:

- **Import**
- **Preparation**
- **Analysis**

Within each area, the scripts are sorted alphanumerically. Regardless of script name, a task executes all the scripts in an area before moving to the next area.

For more information, see [ANALYTIC tag](#).

Rename analytic scripts to create an execution sequence

If you want two or more analytic scripts to execute in sequence they need to be named in a way that creates the execution sequence you want. For more information, see "Replicating analytic chains in Robots" on page 19.

Note

The analytic script name specified in the analytic header is the name used in Robots. The name of the script in the **Overview** tab of the **Navigator** is not used.

Migrate the scripts

1. In Analytics, open the analytic script that you want to rename.
2. In the `//ANALYTIC` tag in the analytic header, update the analytic name and click **Save the Open**

Project .

For more information, see [ANALYTIC tag](#).

3. Repeat the process for all analytic scripts that require renaming.

Specify an analytic type to group analytic scripts

If you want to group analytic scripts into separate **Import**, **Preparation**, and **Analysis** areas in a task you need to specify a corresponding analytic type for each script. Grouping analytic scripts gives you an additional option for creating a script sequence but it is not required. For more information, see "Replicating analytic chains in Robots" on page 19.

1. In Analytics, open the analytic script that you want to specify a type for.
2. In the `//ANALYTIC` tag in the analytic header, add an analytic type and click **Save the Open**

Project .

For more information, see [ANALYTIC tag](#).

3. Repeat the process for all analytic scripts that require an analytic type.

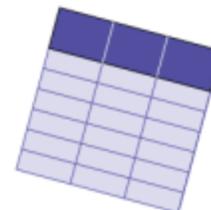
What to do next

Once you have migrated the scripts for one or more automated processes you are ready to start migrating data: "Migrate the data" on page 23.

Migrate the data

Important

Only proceed with migrating the data if you have completed the previous stages of the migration. For more information, see "Recreate your AX repository structure in Robots" on page 11 and "Migrate the scripts" on page 17.



If your AX repository contains either of the types of data listed below, you may need to migrate the data to Robots. Depending on the nature of the data, using a script to import a fresh instance of the data to Robots may be quicker and easier, and more appropriate, than physically migrating an existing table from AX.

If your AX repository does not contain any data that needs to be migrated or recreated, you can move to the next stage in the migration: "Recreate automated AX processes in Robots" on page 35.

Type of data	AX location
Analytics data tables	Data subfolder
Input files such as Excel and *.csv	Related Files subfolder

Recreating or migrating Analytics data tables

This general guideline can help you decide whether to recreate Analytics data tables in Robots or migrate them from AX:

- **Recreate** – Data tables that are overwritten every time an analytic script runs can be recreated in Robots rather than migrated from AX.
- **Migrate** – Data tables that have records appended to them every time an analytic script runs should be migrated.

Note

When you migrate data tables, all the data tables in a **Data** folder are migrated as a group. So if a **Data** folder contains some tables that are overwritten and some tables that are appended to, all the tables are migrated.

Create the robots first

Whether you recreate or migrate Analytics data tables, the first thing you need to do is make sure that you have created the appropriate robots to contain the tables in Robots. You may already have created the required robots, but if not, you need to create them now. For more information, see "Create robots to structure content and automated processes" on page 14.

How you intend to use the data tables in Robots dictates how you should structure individual robots to contain them.

Intended use of Analytics data table	Robot structure
Store and protect a single copy of data imported from a data source that can then be linked to by multiple automated processes in other robots	Create a robot with a name such as "Source Data". Create multiple robots if you want separate containers for different types of source data. For example, "SAP Source Data", "Oracle Source Data", and so on.
Provide input for subsequent analytic scripts as part of the same automated process, or several automated processes	Create a robot to contain all the scripts and data tables required for the automated process or processes. For example, "A_01_AP analytic tests".

Recreating data tables

Analytics data tables that are overwritten every time an analytic script runs should be recreated in Robots rather than migrated from AX. Migrating tables in this situation is wasted effort because the tables will be overwritten as soon as the scripts that create them run in Robots.

For example, if you import a monthly transaction table from your organization's SAP or Oracle installation, and overwrite the previous month's table, the easiest approach is to start importing the monthly table to Robots using the same analytic script that you use to import the table to AX.

Likewise, any data tables that are overwritten each time automated analysis runs should be recreated by running the analysis in Robots rather than migrated.

Options for recreating data tables in Robots

You can use any of the following options to recreate data tables in Robots:

- Run an entire automated process
- Run only a portion of an automated process
- Use a set-up script to initialize data tables

Run an entire automated process

After migrating the scripts associated with an automated process, create a task in the robot that contains the scripts and run the task ad hoc. If the task runs successfully, all data tables output by the scripts are recreated in the robot containing the task.

This option runs all the analysis and creates all the output results associated with the automated process. If you want to recreate only the data tables at this point, use one of the other options described below.

Run only a portion of an automated process

If you have organized an automated process into separate portions, create a task that runs only those scripts that create data tables. For example, run only the scripts in the import portion of an automated process, or in the import and prepare portions.

If the task runs successfully, the data tables containing imported data, or imported and prepared data, are recreated in the robot containing the task. You avoid creating output results.

Use a set-up script to initialize data tables

If you have created a centralized location for imported data such as a "Source Data" robot, you can create a set-up analytic script to perform a one-time initialization of the data tables in the robot.

In Analytics, create a single analytic script that gathers all the ACLScript commands that you use to import data, such as ACCESSDATA and IMPORT ODBC. Commit the script to the "Source Data" robot and create a task to run the script. You can run the task ad hoc, or schedule it to run overnight. If the task runs successfully, the data tables containing imported data are recreated in the "Source Data" robot.

Note

If a single import in the set-up script fails, the task fails and discontinues any further processing. If you have a number of imports that you want to perform, you can divide them among several set-up scripts and create separate tasks for each script. This approach prevents the failure of a single import from disrupting the other imports.

Migrating data tables

Analytics data tables that have records appended to them every time an analytic script runs should be migrated from AX to Robots rather than recreated in Robots. Migrating tables in this situation is required to preserve the accumulating, historical data that the tables contain.

For example, if you import a monthly transaction table from your organization's SAP or Oracle installation, and append the records to an accumulating yearly table, the yearly table needs to be migrated to Robots.

Likewise, any data tables that have records appended each time automated analysis runs should be migrated to Robots rather than recreated.

The process for migrating data tables to Robots

Diligent Support has made scripts available to assist with migrating historical data tables from AX to Robots. The first script exports the data tables from AX to an intermediary location on your network. The second script imports the data tables from the intermediary location to Robots.

The migration scripts are necessary because, unlike AX, you cannot directly upload Analytics tables to Robots through the user interface.

Note

The scripts work at the individual AX folder level and the individual robot level. You need to run the scripts repeatedly for each folder containing data tables that you want to export, and for each robot that you want to import data tables into.

Steps for migrating data tables

1. "Create an intermediary location on your network" on the facing page
2. "Enable the EXECUTE command in AX" on the facing page
3. "Organize the contents of the Data folder" on the facing page
4. "Download the migration scripts from Diligent Community" on page 29
5. "Import the Analytics project with the migration scripts to AX" on page 29
6. "Run the export script in AX" on page 29
7. "Commit the migration scripts to Robots" on page 30
8. "Create a task and run the import script in Robots" on page 31
9. "Repeat the process for additional AX folders and robots" on page 31

Create an intermediary location on your network

Create a temporary folder for storing the historical data tables as you migrate them from AX to Robots. You can create the folder on your own computer, but if the amount of data you need to move is large, you will probably find it more convenient to use a network location for the folder.

The folder must be accessible to the service accounts that run the AX and the Robots Agent Windows services on their respective servers. Both accounts need **Full control** permissions for the folder. The names of the Windows services are:

- **ACL Analytics Exchange Service**
- **Robots Agent**

Enable the EXECUTE command in AX

If the EXECUTE command is not enabled for AX, you need to enable it.

Caution

The EXECUTE command is disabled by default in AX. If you normally have the command disabled, you should disable it again after you have completed the migration of the historical data tables.

1. In a browser, sign in to the AX Server Configuration web application.

By default, the AX Server Configuration web application is located at https://<ax_server_name>/aclconfig

2. Under **Server > Settings**, click **Yes** for **Allow execute command?**
3. Click **Update Server Settings**.

Organize the contents of the Data folder

Important

Think ahead to the tables that you want in the destination robot in Robots. After you have organized the **Data** folder, only tables that will be contained in the same destination robot should remain.

Before you run the export script in AX, make sure that the **Data** folder from which you are exporting contains only tables that you want in the same destination robot in Robots.

The export and import scripts use a bulk process to copy tables and associated data files **as a group** between a **Data** folder, the temporary storage folder, and a single destination robot in Robots. You cannot select individual tables for export or import.

Migrate the data

If all the tables in a **Data** folder align with a single robot, no additional organization is required. If there is not exact alignment between the **Data** folder tables and a single robot, you need to create the alignment before you export.

How to create alignment between a Data folder and a robot

You can align tables between a **Data** folder and a robot using either of these approaches, depending on the situation:

- **Create a staging Data folder** – Create a new collection and folder for staging data tables before exporting. Copy tables aligned with a single destination robot to the staging **Data** folder and export them from there.

Use this approach if you intend to run AX and Robots in parallel for a period of time and you do not want to disrupt the contents of an existing **Data** folder.

- **Edit an existing Data folder** – Delete or move non-aligned tables from an existing **Data** folder before exporting. Delete or move any tables that are not aligned with a single destination robot.

Caution

Only use this approach if the required modifications to the existing **Data** folder do not disrupt any ongoing automated process in AX.

How to handle different table and layout types in a Data folder

Important

Before you run the export script in AX, review the information below regarding the different types of tables and layouts that can appear in a **Data** folder. If you are editing an existing **Data** folder, delete or move tables or layouts as required.

Table or layout type	Data folder requirements
 standalone table	Include only standalone tables that should be contained in the destination robot.
 shared table	Include only shared tables that should be contained in the destination robot. AX shared table functionality is not migrated to Robots.
 master table	Include only master tables that should be contained in the destination robot. Linking relations between tables are not migrated and can be recreated in Robots.
 linked table	If you export a linked table it becomes an unlinked table with an associated data file like any other table in the temporary storage location. This result may or may not match your intentions.

Table or layout type	Data folder requirements
linked table	If you want the linked table to also be a linked table in Robots, delete the linked table from the Data folder in AX, and recreate the link in Robots.
 standalone layout	Remove any table layout from the Data folder. The script requires that all data tables are associated with a source data file. For a layout, the Data source property in the table Summary tab states No data source is associated with this data definition.
 master layout	
 linked layout	

Download the migration scripts from Diligent Community

1. Go to: <https://community.wegalvanize.com/s/article/Project-migration-from-AX-to-Robots>
2. Under **Attachments**, click the **Table_Migration** Analytics project to download it.

Note

If you do not see the project, you may need to log in to Community using your HighBond account credentials.

Import the Analytics project with the migration scripts to AX

1. Open AX Client and sign in.
2. In the **Server Explorer**, select the folder containing the historical data tables that you want to export.
3. Select **File > Import** and navigate to the location where you downloaded the **Table_Migration** Analytics project.
4. Select the project and click **Open**.
5. In the **Select the Import Destination** dialog box, make sure the correct folder is selected and click **Import**.

The project is imported and added to the **Analysis Apps** folder.

Run the export script in AX

1. In the **Table_Migration** analysis app, select the **Data Tables Export** script and select **Analytics > Run**.

Note

The **Data Tables Import** script is for use in Robots, not AX.

2. In the **Run** dialog box, leave **Create a new set** selected, enter a name for the new input value set such as **Export destination**, and click **Next**.
3. In **Tables files common folder path**, enter the path to the temporary storage folder that you created for the data files, and click **Next**.
 - If the folder is on the same server as AX, specify a local file path such as **C:\Temp_data_files_storage**
 - If you are exporting the files to a network location, specify a UNC path such as **\\<server-name>\Temp_data_files_storage**
4. Click **Run**.

Result – The tables in the **Data** subfolder are exported to the storage folder.

Note

If the export script does not work when you specify a UNC path, create a temporary storage folder on the server where AX is installed, and specify a local file path for **Tables files common folder path**.

This method may require that you manually copy the data tables in the temporary storage folder at a subsequent point in the migration process.

Commit the migration scripts to Robots

1. Open the **Table_Migration** Analytics project.
2. From the Analytics main menu, select **File > Commit Scripts As**.
3. If required, in the **Select Destination** dialog box, double-click the appropriate HighBond instance.

The **Robot Collection** appears with the list of existing robots.

4. In the list of robots, select the appropriate destination robot for the data tables and click **OK**.
5. Enter a short commit message such as **Commit data table migration scripts**, and click **OK**.

The scripts are committed to development mode in the robot.

6. In the **Commit Scripts Successful** dialog box, click the second link to inspect the newly committed scripts in the Robots app.

The **Script versions** tab opens with the committed version of the scripts.

7. Select the script version (**vn**).

The **Version details** side panel opens. The names of the individual analytic scripts and auxiliary scripts are listed beneath **Script version changes**.

8. In Analytics, click **OK** to exit the **Commit Scripts Successful** dialog box.

Create a task and run the import script in Robots

1. Select the **Tasks** tab and click **Create task**.

The **Task Designer** opens.

2. Type a name for the task such as **Import data tables**, and click **Save**.
3. Select **Activate analytic** for **Data Tables Import**.

Note

The **Data Tables Export** script is for use in AX, not Robots.

4. Click the down arrow  to expand the parameter section.
5. In **Tables files common folder path**, enter the path to the temporary storage folder that you created for the data files.
 - If the folder is on the same server as the Robots Agent, specify a local file path such as **C:\Temp_data_files_storage**
 - If you are importing the files from a network location, specify a UNC path such as **\\<server-name>\Temp_data_files_storage**
6. At the bottom of the page, click **Skip to end** and then **Confirm and create task**.
7. On the **Tasks** tab, select the task that you just created and click  > **Run now**.

Result – The data tables in the temporary storage folder are imported to the robot. You can monitor the progress of the task in either the **Tasks** tab or the **Task runs** tab.

8. Select the **Input/Output** tab to view the imported data tables.

The tables appear in the **Other tables** section. Select individual tables to open the **Table details** side panel.

Once you are satisfied that the data tables were successfully imported to Robots, you can delete the entire contents of the temporary storage folder.

Note

If the import script does not work when you specify a UNC path, manually copy the temporary storage folder to the server where the Robots Agent is installed, and specify a local file path for **Tables files common folder path**.

You may be able to copy-and-paste the temporary storage folder, or use FTP. Your IT department should be able to assist you if you have difficulty copying the temporary storage folder.

Repeat the process for additional AX folders and robots

Repeat the data table migration process for additional AX **Data** folders and robots as required.

Migrate the data

To match your planned organization of data in Robots, you can migrate data tables from different AX **Data** folders to the same robot. Empty the temporary storage folder before each export operation.

- **Copy the scripts in AX** – Copy the **Table_Migration** analysis app between AX folders as a quick way to replicate the scripts. You do not need to repeatedly import the **Table_Migration** project.

Note

If you are using a staging **Data** folder, you do not need to copy the scripts. Update the contents of the staging **Data** folder as required, and run the export script from the same location.

- **Empty the temporary storage folder** – Once you are satisfied that the previous group of data tables was successfully imported to Robots, delete the entire contents of the temporary storage folder. The migration scripts are not designed to support the accumulation of data tables from multiple **Data** folders in the temporary storage folder.
- **Re-commit the scripts to Robots** – You need to commit the scripts in the **Table_Migration** Analytics project to each robot in which you want to use them. Copying scripts between robots is not supported.

Create linked tables in Robots

If you use master and linked data tables in AX and you want to continue this arrangement in Robots, you need to recreate the linking relations in Robots. For set-up information, see [Sharing and linking data tables](#).

Uploading or migrating non-Analytics related files

You have two options for populating non-Analytics related files such as Excel or *.csv in Robots:

- migrate the files from AX to Robots
- upload the files from a network location

Uploading the files requires fewer steps. You may need to use both options depending on the availability of files. Regardless of which approach you take, make sure that you populate Robots with the current version of each file.

Migrate the files from AX to Robots

You cannot migrate related files directly from AX to Robots. Export related files from AX to an intermediary location, accessible to both AX and Robots, such as your own computer.

This procedure assumes that you are using your own computer as the intermediary location.

1. Create a folder on your computer for storing exported files.
2. In the AX **Server Explorer**, do one of the following:
 - select a **Related Files** folder to export all the files in the folder
 - select individual files in a **Related Files** folder using **Ctrl+click**
3. Select **File > Export**.
4. In the **Export** dialog box, deselect **Work with the exported file(s)?**.
5. Click **Browse**, select the folder that you created for storing files, and click **OK**.
6. Click **OK** to export the file or files.

The files are exported to the specified location.

7. Repeat the process for files in other **Related Files** folders that you want to migrate.
8. Continue with the procedure below for uploading files.

Upload the files from a network location

If you have current versions of related files stored locally, or available on your network, you can upload them directly to Robots.

Note

Related files must not exceed 1 GB in size and must be a permitted file type. For a list of permitted file types, see [Permitted types of non-Analytics related files](#).

1. From the Launchpad homepage (www.highbond.com), under **Automation**, click the Robots app.
You need to sign in to HighBond if you are not already signed in.
2. From the dashboard in Robots, click the robot that you want to upload the related file or files to.
3. In the top right corner of the robot, click the **Dev mode** toggle to switch to development mode.
The production mode header has a light background and the development mode header has a dark background.
4. On the **Input/Output** tab, do one of the following:
 - **Drag and drop** – Select one or more files from your local file system and drag them onto the **Drag and drop** upload section.
 - **Choose files** – Click **Choose files**, browse to the files that you want to upload from your local file system, select them, and click **Open**.

Result – the file or files are uploaded and appear under **Files** in the **Input/Output** tab.

What to do next

Once you have migrated the data for one or more automated processes you can move to "Recreate automated AX processes in Robots" on page 35.

Recreate automated AX processes in Robots

Important

Only proceed with recreating automated processes in Robots if you have completed all previous stages of the migration. For more information, see "Performing the migration: our recommended approach" on page 6.



Once you have migrated or recreated your organization's scripts and any data, you are ready to recreate automated AX processes in Robots. For each automated process you create a **task** inside a robot. You use the task to select which scripts should run, specify any input parameters for the scripts, and create a schedule.

Create a robot task for each automated process

For each automated process, create a task in development mode first so that you can test the process to ensure it is working correctly.

For detailed information, see [Creating and scheduling tasks in Robots](#).

1. From the Launchpad homepage (www.highbond.com), under **Automation**, click the Robots app.
You need to sign in to HighBond if you are not already signed in.
2. From the dashboard in Robots, click the robot that you want to contain the task.
3. In the top right corner of the robot, click the **Dev mode** toggle to switch to development mode.
The production mode header has a light background and the development mode header has a dark background.
4. On the **Tasks** tab, click **Create task**.
The **Task Designer** opens.
5. Type a name for the task that identifies the automated process, and click **Save**.

6. Progress through the **Task Designer** to configure the task settings:
 - a. Select the scripts to run and enter any required parameter values.

Replicate the parameter values from the AX process unless for some reason you are changing the values.
 - b. Optional. Schedule the task.

You are going to run the task ad hoc, so you can skip creating a schedule. Or you can create a schedule if you also want to test the scheduling capability.
 - c. Optional. Select one or more recipients to receive an email notification if the task fails.
7. Review the task configuration settings, and click **Confirm and create task**.

Result – The task is created using the settings that you specified.

Test a task by running it ad hoc

Test each task in development mode and troubleshoot any problems that occur.

1. On the **Tasks** tab, select the task that you just created and click  > **Run now**.

The task starts and you can monitor its progress and view results in either the **Tasks** tab or the **Task runs** tab.
2. In the **Task runs** tab, once the task completes, click the entry for the task.

The **Task run details** side panel opens and displays detailed information about the task run, including a list of any output tables and files generated by the `//RESULT` analytic tag.
3. Review any output results to confirm that the correct tables, logs, or files were generated.

For information about viewing the content of the output results, see [Viewing Robots tables, logs, and files](#).
4. In the **Input/Output** tab, review any output data tables to confirm that the correct tables were generated.

Tables generated by the `//DATA` analytic tag are listed in the **Other tables** section. For information about viewing the content of the data tables, see [Viewing Robots tables, logs, and files](#).
5. If a task fails, or if output tables or files are missing, the "Basic troubleshooting steps" on the facing page might identify the problem.
6. If a task continues to fail after you attempt troubleshooting, contact Diligent Support for assistance.

Basic troubleshooting steps

Type of problem	Possible cause, or action to take
General	Check the Result status in the Task run details side panel. In some cases, a message appears explaining the reason for the task failure.
	If a log file was output, click View to inspect the log. The log should include an entry, marked with a red X  , that indicates why the analytic script failed.
Task input	If the task uses a linked table as input, confirm that the table is present in the Linked tables section in the Input/Output tab. For more information, see Sharing and linking data tables .
	If the task uses a non-Analytics related file such as Excel as input, confirm that the file is present in the Files section in the Input/Output tab. For more information, see Managing non-Analytics related files .
Analytic header	In Analytics, check the analytic headers on all analytic scripts run by the task. Confirm that all required analytic tags, such as <code>//RESULT</code> , <code>//DATA</code> , and <code>//FILE</code> , are present and correctly formatted. For detailed information, see Analytic headers and tags .
Robots Agent service	On the Windows server where the Robots Agent is installed, check that the Robots Agent service is running. Start the service if it is stopped and re-run the task.

Activate a script version to production

Once a task is running correctly, activate the script version used by the task to production.

For detailed information, see [Development mode and production in Robots](#).

1. On the **Script versions** tab, select the script version used by the task that you just tested.
2. In the **Version details** panel, click **Activate version**.
3. Optional. Enter a comment in the version history to explain what you are activating.

Tip

Entering an informative comment when you activate a script version is a best practice that can be very helpful when reviewing automated analysis at a later date.

4. Click **Activate *versionNumber***.

Result – The script version is activated and becomes available in production. The **Scripts** tab in production lists the individual analytic scripts, and any auxiliary scripts, in the script version.

Create and schedule a robot task in production

The final step is to create and schedule a robot task in production.

For detailed information, see [Creating and scheduling tasks in Robots](#).

1. In the top right corner of the robot, click the **Dev mode** toggle to switch to production.

The production mode header has a light background and the development mode header has a dark background.

2. On the **Tasks** tab, click **Create task**.

The **Task Designer** opens.

3. Type a name for the task that identifies the automated process, and click **Save**.

4. Progress through the **Task Designer** to configure the task settings:

- a. Select the scripts to run and enter any required parameter values.

Replicate the parameter values from the AX process unless for some reason you are changing the values.

- b. Schedule the task.

Replicate the schedule from the AX process unless for some reason you are changing the schedule.

- c. Select one or more recipients to receive an email notification if the task fails.

5. Review the task configuration settings, and click **Confirm and create task**.

Result – The task is created using the settings that you specified.

6. Test the task by running it ad hoc.

For more information, see "Test a task by running it ad hoc" on page 36.

If the task fails, or if output tables or files are missing, confirm that you activated the correct script version from development mode. If the task continues to fail, see "Basic troubleshooting steps" on the previous page, or contact Diligent Support for assistance.

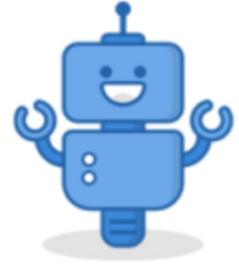
What to do next

Once you have recreated one or more automated processes in Robots you can move to "Post-migration activities" on page 41.

This page intentionally left blank

Post-migration activities

After you have migrated or recreated your organization's scripts and any data, and recreated automated processes in Robots, move on to the post-migration activities.



Task/Activity	Details
If applicable, install Direct Link on the server where the Robots Agent is installed	If you have been using Direct Link with AX to access SAP data you need to: <ul style="list-style-type: none">◦ Install Direct Link to work with the Robots Agent◦ Transition to using the ACL Connector for SAP (the SAP Connector) instead of Direct Link See Connecting to SAP .
Test viewing output results in Analytics	See Viewing tables, logs, and files .
Compare Robots and AX output results	You should run AX and Robots in parallel for a period of time and compare output results from the same automated processes in each application. Assuming you have not changed the logic of the process in Robots, output results should be identical.
Set up user and admin access to the Robots app and individual robots	See Robots app permissions .
Develop a plan for legacy AX data	Your organization's AX installation may contain a large amount of legacy data. Organizational policy, and laws in your jurisdiction, may require that you preserve accessible copies of the data for a specified period of time. Work with the appropriate stakeholders in your organization, and your Diligent account representative, to plan an orderly process for archiving or purging legacy AX data.

This page intentionally left blank

Additional information about migrating

The benefits of migrating

Robots is the next-generation replacement for AX. Both products offer the same basic capability: automating data acquisition and analysis as part of an organization's data-driven governance, risk, and compliance activities.

Diligent is investing heavily in ongoing development and expansion of Robots and its associated capabilities, including the new HighBond Command Language (HCL) for data analysis in the cloud. In September 2020, we announced our end-of-support plans for AX, with the following dates:

- **Spring 2021** – Final release of AX, maintenance release only
- **January 1, 2023** – End of support for AX

Comparison of Robots and AX

The main differences between Robots and AX are outlined below. For detailed or specific questions pertaining to your organization, speak with your Diligent account representative.

Functional area	Robots	AX
User interface	Modern, web-based interface with intuitive design, and easy-to-use workflow	Older, client-server interface with less optimal layout and less intuitive workflow
Installation, administration, and upgrade	Greatly reduced technical complexity associated with installation, administration, and upgrade Reduced reliance on components produced by third parties decreases exposure to enhancement and security update issues	Significant technical complexity and maintenance burden associated with installation, administration, and upgrade of AX and associated database Reliance on internal components produced by third parties that follow their own enhancement and security update schedules (including Apache TomEE, Java, PostgreSQL, Oracle)
Testing environment	Integrated development and production modes allow separated testing of automated analysis before	No comparable capability

Additional information about migrating

Functional area	Robots	AX
	going live	
Script versioning	Script version history with easy rollback to previous versions of scripts	No comparable capability
Scripting languages	The option of traditional ACL scripting or the new, Python-based HighBond Command Language (HCL)	Traditional ACL scripting only
Customizable script sequences	By default, multiple analytic scripts in a single robot task execute sequentially Comparable chaining functionality is on the development roadmap	Chained analytic scripts
Product architecture	Simplified product architecture: <ul style="list-style-type: none"> ◦ single, unified cloud-based platform for all user and administrator activities ◦ optional, on-premise data processing component 	Multiple-component product architecture: <ul style="list-style-type: none"> ◦ network-based server component and associated database ◦ desktop client ◦ web-based client ◦ web-based administrator utility
	Cloud/on-premise hybrid: <ul style="list-style-type: none"> ◦ The option of keeping organizational data and data processing within the security of your Windows network while still benefiting from the web-based interface for task scheduling and administration 	On-premise only: <ul style="list-style-type: none"> ◦ The entire AX client-server architecture is installed on your network
Processing capacity and load-balancing	Multiple Robots Agents for increased processing capacity and load-balancing	AX Engine Nodes for increased processing capacity and load-balancing
Issue remediation	Integrated access to HighBondResults app for issue remediation	No comparable capability
Visualizations and dashboards	Integrated access to HighBondResults and Storyboards apps for visualizations and dashboard creation	Visualizations in AX Web Client, no dashboard capability
Pre-built content	Access to Robotics Toolkits (pre-built data integration and analysis script packages)	No comparable capability

How much time and effort does the migration take?

The time and effort required to migrate from AX to Robots depends on the complexity of your current AX installation and the number of automated processes you are running. Migration projects from one to three weeks in duration have been the norm for customers who have already migrated.

Budget additional time if you plan to transition from using Direct Link to access SAP data, to using the SAP Connector. You do not have to make this transition at the same time that you migrate. You can continue to use Direct Link with Robots, and make the transition to the SAP Connector at a later date.