

Aras Import Manager 33

Last Modified: 08/28/2026

Copyright Information

Copyright © 2026 Aras Corporation. All Rights Reserved.

Aras Corporation
100 Brickstone Square
Suite 100
Andover, MA 01810
Phone: 978-691-8900

Notice of Rights

Copyright © 2026 by Aras Corporation and/or its affiliates. All rights reserved.

This document is protected by U.S. and international copyright laws and conventions. No copyright may be obscured or removed from this document. This document may not be modified or altered, or reproduced or transmitted in any form, without the explicit permission of the copyright holder.

Aras Innovator, Aras, and the Aras Corp "A" logo are registered trademarks of Aras Corporation in the United States and other countries.

All other trademarks referenced herein are the property of their respective owners.

Notice of Liability

This document is provided for informational purposes only, and the contents hereof are subject to change without notice. The information contained in this document is distributed on an "as is" basis, without warranty of any kind, express or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose or a warranty of non-infringement. Aras shall have no liability to any person or entity with respect to any loss or damage caused or alleged to be caused directly or indirectly by the information contained in this document or by the software or hardware products described herein.

Table of contents

Release Notes	3
Platform Compatibility	4
Key Features	5
Admin Guide	7
Introduction	8
Prerequisites	9
Configuration	10
Installation Guide	16
Introduction	17
Prerequisites	18
Aras Innovator Setup	20
Installing Aras Import Manager	21
SaaS Installation Guide	37
Introduction	38
Definitions	39
Aras Import Manager Installation	40
User Guide	47
Introduction	48
Import Items	49
Features	57



Release Notes



Platform Compatibility

The following Aras Innovator Releases are supported with this release of Aras High Tech Electronics:

- Aras Innovator Release 33
- Aras Innovator Release 34
- Aras Innovator Release 35

For the latest information regarding Innovator version support, please refer to the Support Matrix accessible on the subscriber portal: <https://www.aras.com/community/subscriber-portal/p/support-matrix>



Key Features

The Aras Import Manager (AIM) delivers business value by automating and standardizing data import processes across the enterprise. It reduces manual effort and human error, enforces data integrity through configurable validation rules, and seamlessly integrates with existing change management workflows. By leveraging AIM in a SaaS environment, organizations can accelerate time-to-value, ensure traceability of imported data, and improve efficiency in PLM data migration, synchronization, and data loading use cases.

Aras Import Manager (AIM) is a configurable import framework for Aras Innovator, designed to streamline and govern the data import process. AIM supports both flat and hierarchical (parent-child) imports with the ability to enforce change control, apply validation rules, and support custom logic per use case.

Flexible Import Modes

- Ad Hoc Mode: Directly creates or updates items during import.
- With Change Mode: Routes imports through change management workflows using ECO/DCO or other change items.

Advanced Configuration

- Central Import Configuration: Only one active configuration per system, managed under Administration > Import Manager > Import Configurations.
- Available Imports: Define what item types can be imported, the import structure (Flat or Parent-Child), and relationship handling (append vs. replace).

Custom Behavior and Hooks

- Fetch Query Builder: Override default item lookup logic.
- On Add/Update/Delete Methods: Attach methods to customize behavior for each import action.
- Ambiguity Fixers: Handle cases where imported data might match multiple existing items.

Change Management Integration

- Change ItemType Selection: Define which change object (e.g., ECO) governs the import.



- Workflow Integration: Methods to create, edit, and route change items as part of the import lifecycle.

Validation Framework

- Rule-Based Validation: Add custom rules with severity levels (info, warning, error) for pre-import checks.
- Applicability Matrix: Target rules by Import Config, ItemType, Mode, and Type.



Admin Guide



Introduction

Purpose and Target Audience

This document provides detailed instructions for configuring the Aras Import Manager (AIM). It is intended to allow a novice administrator to configure AIM. However, general knowledge of system administration, Microsoft Reporting Services, and Aras Innovator administration is recommended.



Prerequisites

The three significant steps for setting up your Aras Innovator instance for use with Aras Import Manager are:

- 1. Install Aras Innovator.
- 2. Install Aras Update Tool.
- 3. Install Aras Import Manager using Aras Update Tool.

Important

This document is based on a new installation Aras Innovator. However, the same steps for installing to an existing instance of Aras Innovator can be used. Aras Import Manager is a stand-alone solution and does not depend on the other Aras business ready solutions.

Checking for Eligibility

The following steps outline the process to verify the code tree and database are eligible for installing the Aras Import Manager 33 application:

- 1. Log into **Aras Innovator** as an administrator.
- 2. Click the **User** icon in the top-right corner and select **About**.
- 3. Close the dialog.
- 4. Verify that the dialog shows Release 33.
- 5. From the **TOC**, expand **Administration** and select **Variables**.
- 6. Verify the following variables and values are listed:

VersionMajor	VersionMinor	VersionServiceUpdate	VersionLabel
14	30	0	Release 33

If the service pack requirements listed in the previous steps are not met, contact Aras to discuss options.

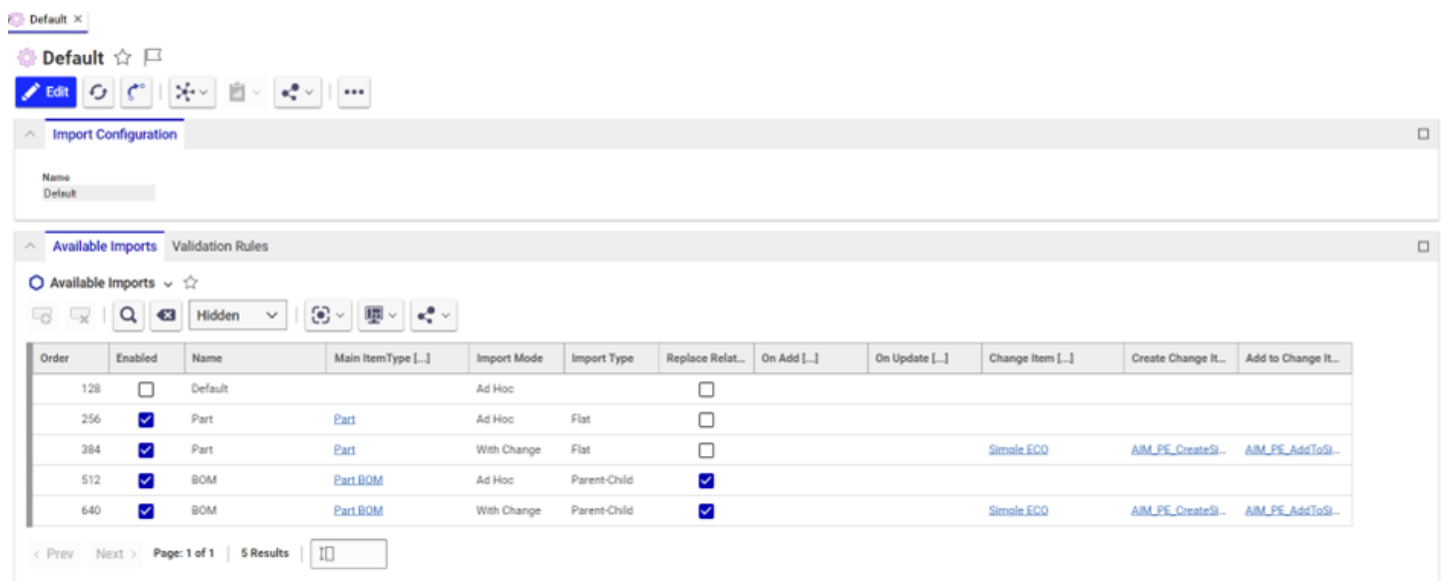


Configuration

After installation or integration of the **AIM**, it is possible to use import with default configuration. Import process configuration is represented by two ItemTypes: **Import Configuration** and **Import**.

Import Configuration

This Item (**AIM_ImportConfiguration**) is the main configuration item for Aras Import Manager. It is available only for the "Administrators" identity by default. To find Default Import Configuration, navigate in the TOC to the *Administration > Import Manager > Import Configurations*.



Order	Enabled	Name	Main ItemType [...]	Import Mode	Import Type	Replace Relat...	On Add [...]	On Update [...]	Change Item [...]	Create Change It...	Add to Change It...
128	☐	Default		Ad Hoc		☐					
256	☒	Part	Part	Ad Hoc	Flat	☐					
384	☒	Part	Part	With Change	Flat	☐			Simple ECO	AIM_PE_CreateSi...	AIM_PE_AddToSi...
512	☒	BOM	Part.BOM	Ad Hoc	Parent-Child	☒					
640	☒	BOM	Part.BOM	With Change	Parent-Child	☒			Simple ECO	AIM_PE_CreateSi...	AIM_PE_AddToSi...

Only one *Configuration* item can be in the system, another one cannot be created. AIM is designed to work with **default configuration item only**. Import Configuration has two Relationships:

- **Available Imports**, to configure Import.
- **Validation Rules**, to manage imports based on validation rules.

Available Imports

This Relationship allows you to manage what kind of items and how they can be imported to the Innovator:

Available Imports											
Order	Enabled	Name	Main ItemType [...]	Import Mode	Import Type	Replace Relat...	On Add [...]	On Update [...]	Change Item [...]	Create Change It...	Add to Change It...
128	☐	Default		Ad Hoc		☐					
256	☒	Part	Part	Ad Hoc	Flat	☐					
384	☒	Part	Part	With Change	Flat	☐			Simple ECO	AIM_PE_CreateSi...	AIM_PE_AddToSi...
512	☒	BOM	Part.BOM	Ad Hoc	Parent-Child	☒					
640	☒	BOM	Part.BOM	With Change	Parent-Child	☒			Simple ECO	AIM_PE_CreateSi...	AIM_PE_AddToSi...

On the right mouse button click on the Available Import row, you can open the form to see more details:

Part Flat AdHoc

Edit

Available Import

Name

Part

Enabled

☒

Main ItemType

[Part](#)

Import Mode

Ad Hoc

Import Type

Flat

Replace Relationships

☐

Item Action

Fetch Query Builder

On Add

On Update

On Delete

Change Management

Change Item

Create Change Item

Add to Change Item

Move Change Item to Edit State

Ambiguity Fixers

On Main Item

On Parent Item

On Child Item

Here is the list of properties and a description of their influence on the import behavior:

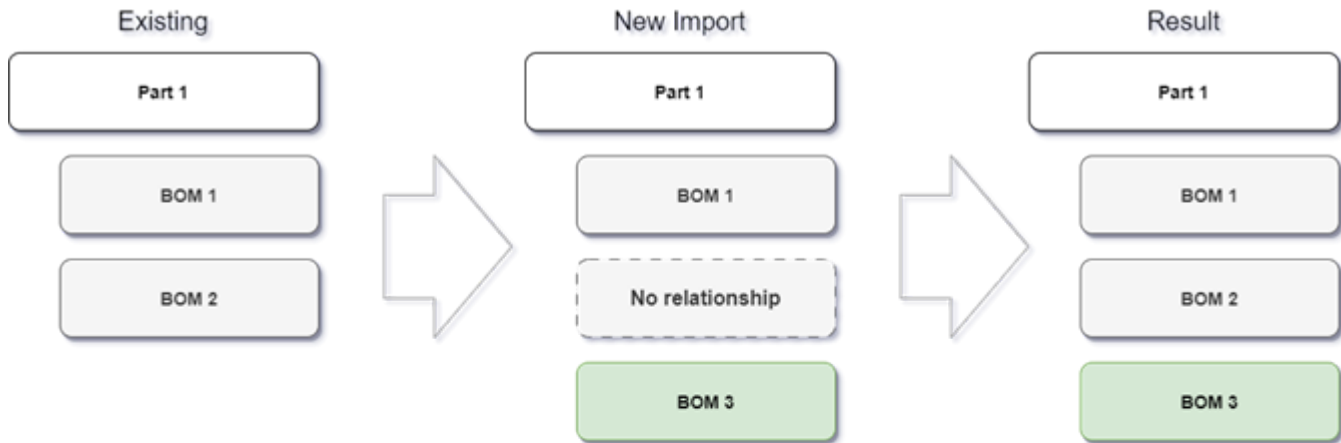
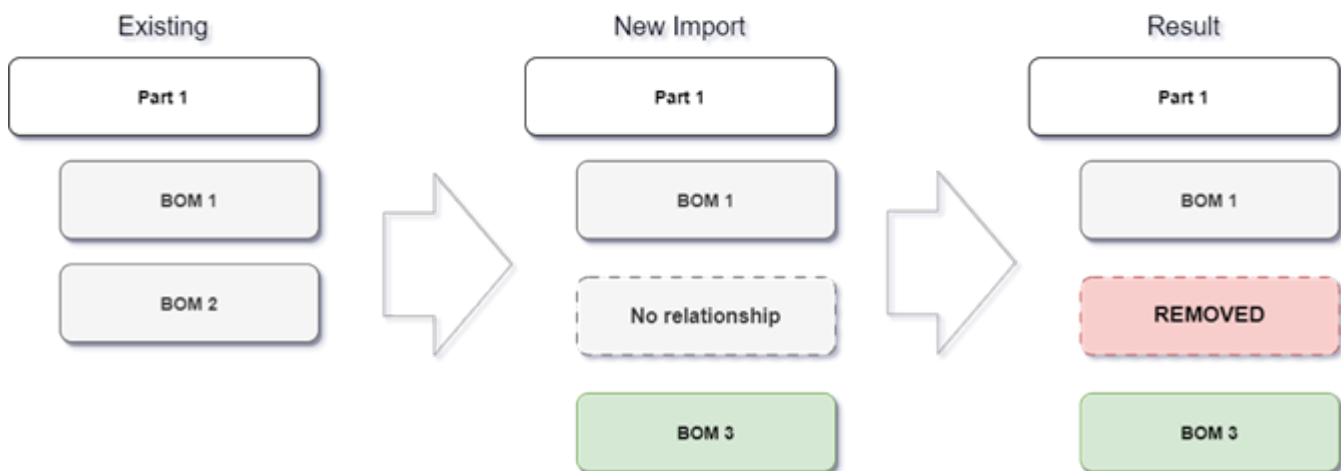
- **Name** - technical name of the import, the same names are in the AIM_Import_ConfigurationList list.
- **Enabled** - flag, meaning configuration is available right now and can be used.
- **Main Item Type** – select an IT for which this import is defined.
- **Import Mode**
 - **Ad Hoc** – direct change/creation of Items during import.

- **With Change** – create a change item to track import:
 - to get into states where you can modify the item and then do the change set (with an ECO item),
 - to let you release the item directly after adding,
 - to follow the organization process.
- **Import Type**
 - **Flat** – import items without relationships (eg. Part).
 - **Parent-Child** – import with connection parent-child (eg. Part BOM).
- **Replace Relationship** - flag for append or fully replace parent relationship with the import data.
Only for Parent-Child Import.

Example: Part 0 exists in the system with Part 1 as BOM. We imported Part0 with Part2 as BOM.

Replace Relationship = falseAs a result, we'll have Part0 with Part1, Part2 as BOM.
Replace Relationship = trueAs a result, we'll have Part0 with Part2 as BOM.



ReplaceRelationships = false**ReplaceRelationships = true**

- **Item Action**

- **Fetch Query Builder** - replace default query to find an item to apply an update (search for an Item by required fields).
 - You have two parts with the same item_number and an additional field to define the current one. Write your own logic to find an item by these specific conditions. Otherwise, a set of required fields from the import item is used.
- **On Add, On Update, On Delete** – replace default Innovator actions called during the import (in case you want to apply some specific logic before it).
- **Change Management** (mandatory for Change Mode)
 - **Change Item** - define the change ItemType (ex. ECO, DCO).



- **Create Change Item** – define a method to create this change item.
- **Add to Change Item** - define the method used to add an item built from the File to the Change Item (for Parent-Child import only Parent will be added as an Item to Affected Items).
- **Move Change Item to Edit State** - define the steps and how to vote a change Item to a step where the edit is possible (i.e. move an ECO to In Work)
- **Ambiguity Fixers** - called when the Fetch (default or custom from item action) methods could not pull a single entry from the DB to match the File row. Administrators define a method that would choose between the retrieved Items. If no single Item is chosen, the row will be flagged with a **Warning**.
 - **On Main Item.**
 - **On Parent Item.**
 - **On Child Item.**

Validation Rules

This relationship is used to collect validation rules for imports:

Available Imports Validation Rules									
Validation Rules									
Order	Enabled	Name	Severity	Config. Name	ItemType [...]	Import Mode	Import Type	Method [...]	Short Message
100	☒	No Create Permission	Error				Flat	AIM_VaiRule_CanAddItem	User cannot create this ItemType
200	☒	No Update Permission	Error			Ad Hoc	Flat	AIM_VaiRule_CheckItemEditable	User cannot lock item
300	☒	Missing Mandatory Pro...	Error					AIM_VaiRule_CheckMandatoryPr...	Missing mandatory properties
400	☒	Value not matching Pro...	Error					AIM_VaiRule_CheckPropertyValu...	Value does not fit the Property definition (length or type)
500	☒	List value does not exist	Error					AIM_VaiRule_CheckListValueExist	List value used does not exist
510	☒	Classification value do...	Error					AIM_VaiRule_CheckClassification	Classification value does not exist.
600	☐	New Item will be created	Log					AIM_VaiRule_CheckForItemCreat...	No match for main item
700	☒	Missing Parent	Warning				Parent-Child	AIM_VaiRule_CheckParentDefined	No parent item defined
800	☒	Missing Child	Info				Parent-Child	AIM_VaiRule_CheckChildrenDefi...	No child item defined
1000	☒	Part state for MCO	Error	AML	Part AML	With Change	Parent-Child	AIM_PE_VaiRule_MCO_CheckPar...	Part (parent) of a Part AML must be released to be added to an MCO.

All rules are applied even if the previous Rule failed. They are applied based on the type of Import, matching the columns valued for the Rule, if a determinant is left empty it is applicable for all values.

Important

A rule can be overridden (and disabled) by a line of a greater order.



The rule is defined by:

1. Order, Name, Severity, message(s), and a Method.
2. Applicability: Config Name, ItemType, Import Mode, Import Type and Enabled.



Installation Guide



Introduction

Purpose

This document provides detailed instructions for installing the Aras Import Manager 33.

Scope

The Installation Guide explains the installation process for Aras Import Manager 33 for Aras Innovator. The installation involves updates to both the code tree and database.

Target Audience

The steps detailed in this document are intended to allow a novice administrator to install Aras Import Manager 33. However, general knowledge of system administration, Microsoft Reporting Services, and Aras Innovator administration is recommended.



Prerequisites

To install Aras Import Manager 33, users need to confirm that the installation of Aras Innovator meets the minimum requirements for running it.

The first step in installing Aras Import Manager is to set up the Aras Innovator instance that will be used. The three significant steps for setting up your Aras Innovator instance for use with Aras Import Manager are:

- 1. Install Aras Innovator.
- 2. Install Aras Update Tool.
- 3. Install Aras Import Manager using the Aras Update Tool.

Important

This document is based on a new installation Aras Innovator. However, the same steps for installing into an existing instance of Aras Innovator can be used. Aras Import Manager is a stand-alone platform component and does not depend on the other Aras business-ready solutions.

Checking for Eligibility

The following steps outline the process to verify the code tree and database are eligible for installing the Aras Import Manager 33 application:

- 1. Log into **Aras Innovator** as an administrator.
- 2. Click the **User** icon in the top-right corner and select **About**.
- 3. Close the dialog.
- 4. Verify that the dialog shows either Release 33, Release 34 or Release 35
- 5. From the **TOC**, expand **Administration** and select **Variables**.
- 6. Verify one of the following variable sets and values are listed:

VersionMajor	VersionMinor	VersionServiceUpdate	VersionLabel
14	30	0	Release 33
14	34	0	Release 34
14	35	0	Release 35



If the service pack requirements listed in the previous steps are not met, contact Aras to discuss options.



Aras Innovator Setup

The Aras Innovator instance must be set up before installing Aras Import Manager. This section describes the major steps in setting up the Aras Innovator instance for use with Aras Import Manager.

Aras Import Manager 33 CD Image

Subscribers can get the Aras Import Manager 33 CD Image from the [Aras File Sharing](#) site in the `Aras/Product Releases/` folder.

The documentation for installing Aras Innovator is in the `Documentation` folder of the Aras Innovator CD Image, also available in the above-mentioned folder.

The Aras Innovator releases that support Aras Import Manager 33 are:

- Aras Innovator 33
- Aras Innovator 34
- Aras Innovator 35

If using an earlier release of Aras Innovator, upgrade to Release 33 before installing the application.

Check the latest information on Innovator release support by consulting the Support Matrix accessible from the Subscriber Portal: <https://www.aras.com/community/subscriber-portal/p/support-matrix>

For upgrade assistance, consult Aras Support (<mailto:support@aras.com>) before installing Aras Import Manager.

Installing Aras Update Tool

Aras Import Manager is installed using the Aras Update Tool. For instructions on installing this tool, refer to the *Aras Update 1.23 – Installation Guide*. Alternatively, Aras Import Manager can be installed manually using the Aras Import Tool.



Installing Aras Import Manager

After the Aras Innovator installation is complete and confirmed, Aras Import Manager installation modifies the Aras Innovator database with metadata required to run the application. Before applying these changes, verify the following requirements:

- Aras Innovator 33
- Aras Update Tool 1.23+

or

- Aras Innovator Package Import Export Utilities

Notification and Backup

1. Notify users that the system will be down at a scheduled time, and that they should log out before starting the process.

Important

It is recommended to provide at least 24 hours' notice, as well as a reminder 15 minutes before the upgrade.

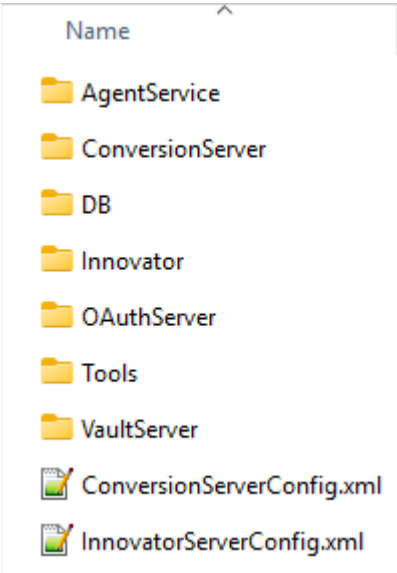
1. Back up the code tree:

The “Code Tree” refers to files and folders installed on the disk with Aras Innovator installation. The default path for the Code Tree installation is typically:

```
C:\Program Files(x86)\Aras\Innovator
```

With the following contents:





The following steps can be used to confirm the installation folder:

- From the Windows search field, enter **Control** to access the **Control Panel**.
- Select **Programs and Features**.
- Right-click the **Name** column and select **More...**
 1. Select the **Comments** header if it is not checked.
 2. Click **OK**.
 - Search for the **Aras Innovator** entry in the **Programs and Features** window.
 - View the value set in the **Comments** column for the **Aras Innovator** entry:

Organize ▾					
Name	Publisher	Installed On	Size	Version	Comments
Aras Innovator	Aras Corporation	10/22/2024	1.24 GB	14.0.28	Installed to C:\Program Files (x86)\Aras\Innovator
Aras Update	Aras Corporation	7/26/2024	42.1 MB	1.23.1357	Aras Update

After verifying the code installation path, back up the folder and its contents.

2. Disconnect all users from the database.

The easiest way to prevent client sessions from committing any further changes to the database is to change the database connection string in the `InnovatorServerConfig.xml` from `<DB-Connection...` to `<xDB-Connection...` and restart the **w3svc service(IIS)**. This expires all



sessions and prevents new connections to the Aras Innovator database through the existing instance.

3. Back up the database.

Store these files in a safe location, as they will be needed for restoration if the process fails.

4. Enable database connections.

After the database backup has completed, enable the database connection string in the `InnovatorServerConfig.xml` by simply changing it from `<xDB-Connection...` to `<DB-Connection...` and restarting the **w3svc service (IIS)**.

Installing the Aras Import Manager Application

The Aras Import Manager application can be installed using either the Automated Installation option (using the **Aras Update** Tool) or the Manual Installation option (using the **Aras Import** Tool).

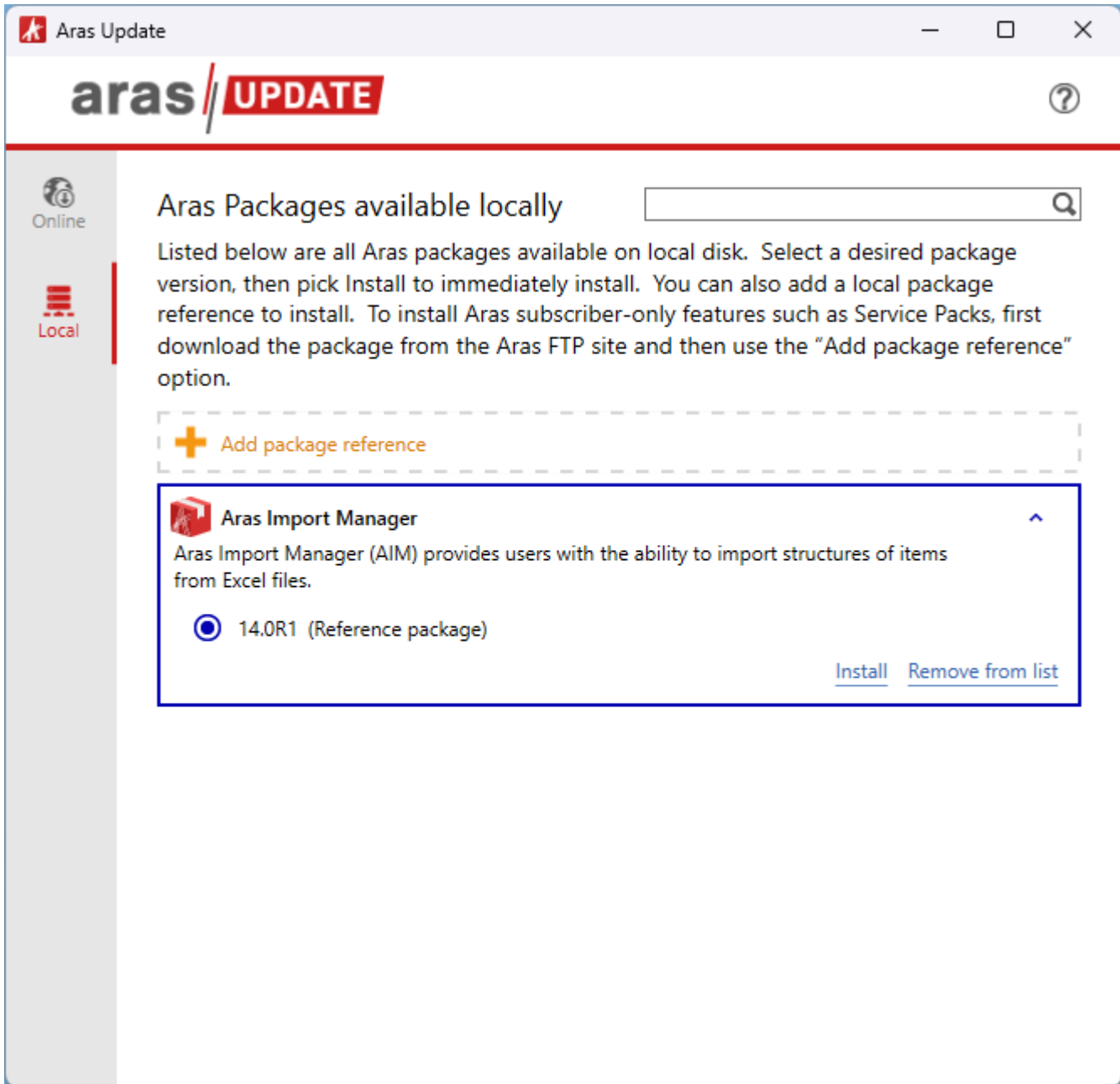
Automated Installation Option

1. Copy and unzip the Aras Import Manager CD Image on the local computer.
2. Enable the **Super User (root)** login.
3. Launch the **ArasUpdate.exe** file as administrator.

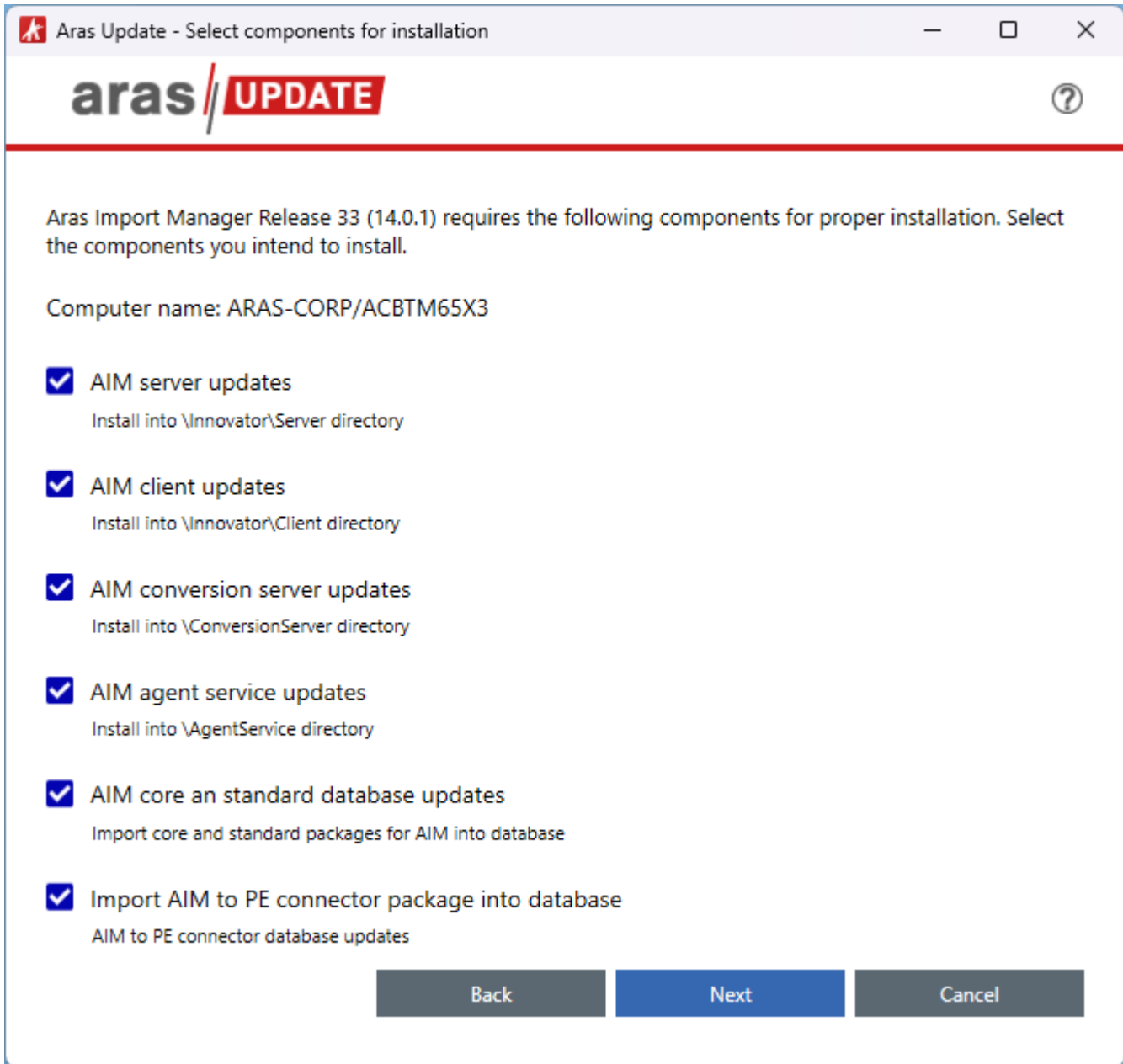
If default options were selected during installation, the **ArasUpdate.exe** file is found in the directory `C:\Program Files (x86)\Aras\Aras Update\`.

4. On the **Local** tab, click **Add package reference** and navigate to the folder where the Aras Import Manager package was extracted in the first step.
5. Expand the **Aras Import Manager** package and click **Install**.

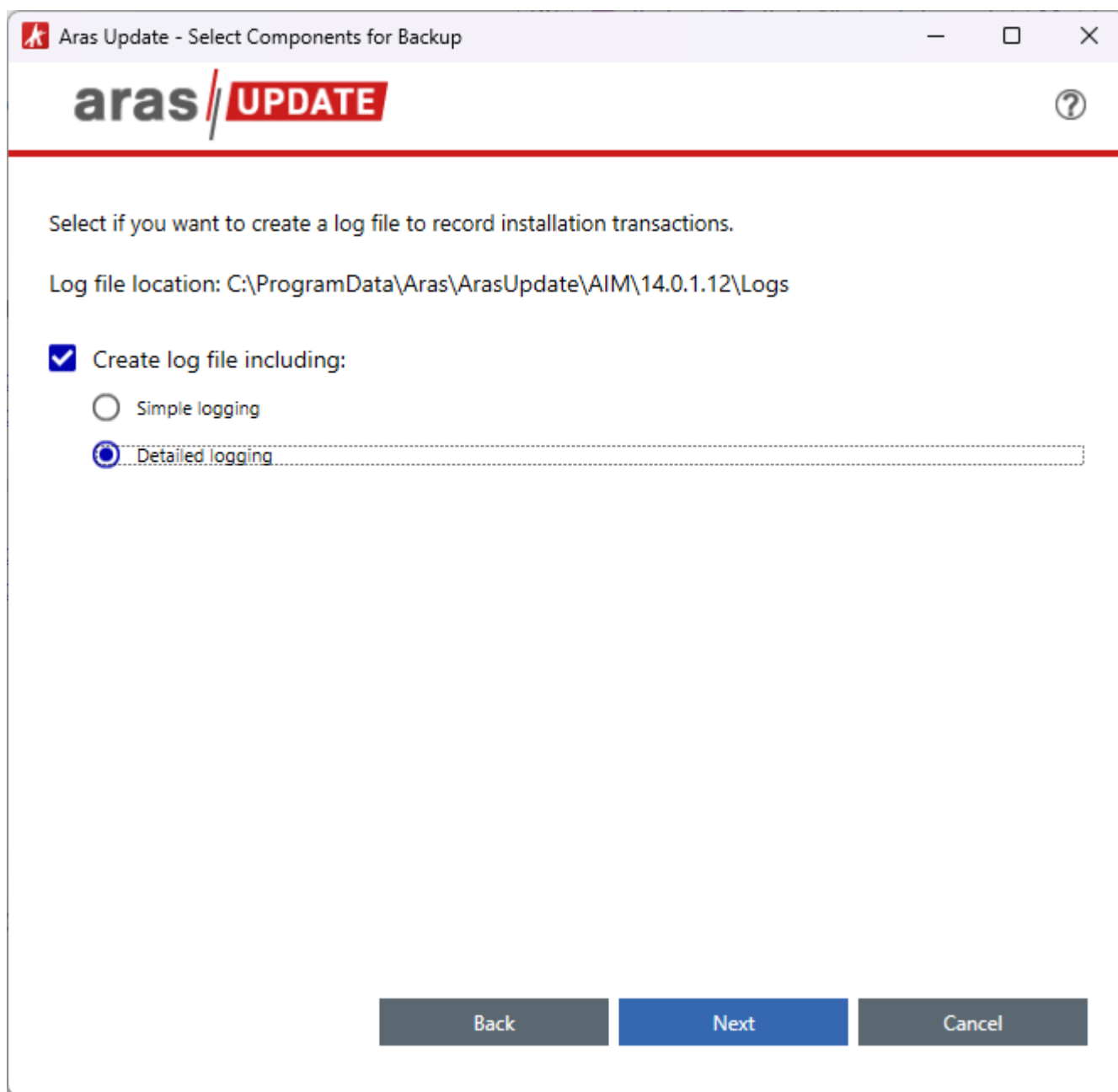




6. Select **Aras Innovator Server Code Tree Updates**, **Aras Innovator Client Code Tree Updates**, and **Import into Innovator DB** options.



7. Click **Next**.
8. Select the **Detailed logging** option and click **Next**.



The logging option records the installation attempt and can be used to troubleshoot issues.

9. Input the connection information and click **Install**.



 Aras Update

 **UPDATE**



Enter the following parameters for each component you are going to install and then click Install button to continue.

✓ AIM server updates

Path to Innovator

✓ AIM client updates

✓ AIM conversion server updates

Conversion Server directory

✓ AIM agent service updates

Agent Service directory

Database name of Innovator instance

✓ AIM core an standard database updates

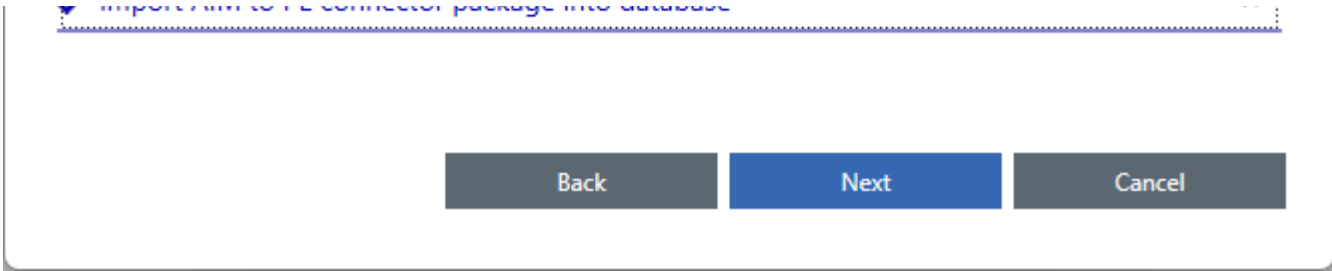
Innovator URL

Innovator Username

Innovator Password

Import AIM to PF connector package into database





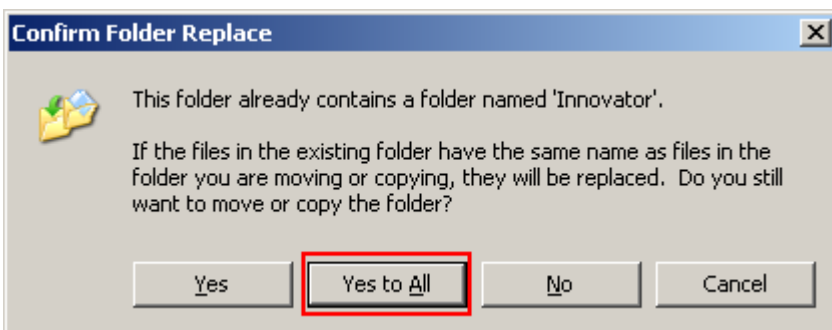
- **Path to Innovator:** The physical path to the Aras Innovator Code Tree (should be the same as the one defined in IIS application settings)
- **Innovator URL:** The connection URL for Aras Innovator
- **Database name of Innovator instance:** The target Aras Innovator database
- **Username:** root
- **Password:** Password for “root” login (Default is “innovator”)
- **Conversion Server:** The physical path to the Aras Innovator Conversion Server (should be the same as the one defined in IIS application settings)
- **Agent Service:** The physical path to the Aras Innovator Agent Service (should be the same as the one defined in IIS application settings)

Aras Update will import everything required for Aras Import Manager.

10. Disable the **Super User (root)** login after the import is complete.

Manual Installation Option

1. Download the Aras Import Manager CD image from the [Aras MFT](#) site, and unzip the file on the local machine.
 1. Copy the `\Innovator` folder to the root of the install directory, overwriting the existing `\Innovator` folder and all its contents.
 2. Copy the `\ConversionServer` folder to the root of the install directory, overwriting the existing `\ConversionServer` folder and all its contents.



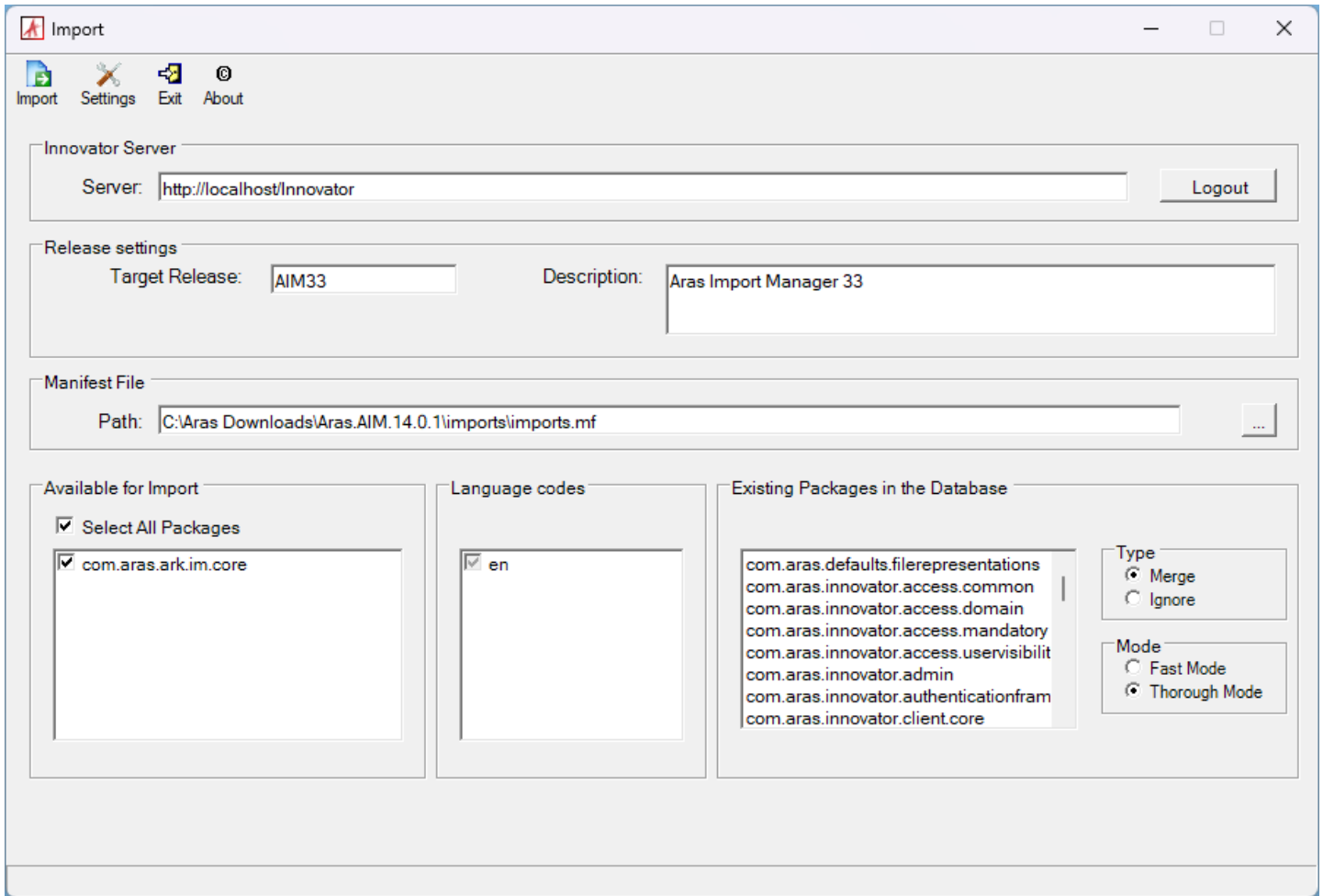
It is recommended that an administrator perform this step on the server.

3. Increment the `filesRevision` setting in the `\Client\InnovatorClient.config` file.
 - Navigate to the `\Innovator\Client\` directory and open the `InnovatorClient.config` file for editing.
 - Change the value of the `filesRevision` attribute. The value should be changed from `std` to `2` , `3` , `4` , etc.

```
<catchingModule moduleEnabled="true" filesRevision="2" />
```

- Save the changes to the `InnovatorClient.config` file and close it.
4. Import the Aras Import Manager database package with the **Package Import Export Utilities**.
For more information using this tool, refer to the *Aras Innovator 33 – Package Import Export Utilities* documentation.
 2. Enable the **Super User (root)** login to perform this step.
 - Browse to the `\PackageImportExportUtilities\Import\` folder and run the **Import.exe** file.





Import

Import Settings Exit About

Innovator Server

Server:

Release settings

Target Release: Description:

Manifest File

Path:

Available for Import

☒ Select All Packages

☒ com.aras.ark.im.core

Language codes

☒ en

Existing Packages in the Database

com.aras.defaults.filerepresentations
com.aras.innovator.access.common
com.aras.innovator.access.domain
com.aras.innovator.access.mandatory
com.aras.innovator.access.uservisibilit
com.aras.innovator.admin
com.aras.innovator.authenticationfram
com.aras.innovator.client.core

Type
☒ Merge
☐ Ignore

Mode
☐ Fast Mode
☒ Thorough Mode

- Input the connection information.
 - **Server:** The connection URL for Aras Innovator

Click the **Login** button and enter:

- **Database:** The target Aras Innovator database
- **Username:** root
- **Password:** Password for “root” login (Default is “innovator”)
- **Target Release:** AIM33
- **Description:** Aras Import Manager 33
- **Manifest File Path:** The manifest file `\Imports\imports.mf`
- **Available for Import:** Select All Packages
- **Type:** Merge
- **Mode:** Thorough Mode



- Click the **Import** button.
- Disable **Super User (root)** login.

Important

The **Super User (root)** login should not be enabled in production.

- Repeat the import with the second manifest `imports.standard.mf`.

- Input the connection information.
 - **Server:** The connection URL for Aras Innovator

Click the **Login** button and enter:



- **Database:** The target Aras Innovator database
- **Username:** root
- **Password:** Password for “root” login (Default is “innovator”)
- **Target Release:** AIM33
- **Description:** Aras Import Manager 33
- **Manifest File Path:** The manifest file `\Imports\imports.standard.mf`
- **Available for Import:** Select All Packages
- **Type:** Merge
- **Mode:** Thorough Mode
- Click the **Import** button.
- Disable **Super User (root)** login.

Important

The **Super User (root)** login should not be enabled in production.

- **Optional:** import with the manifest `imports.pe.mf` .



The screenshot shows the 'Import' application window. It contains the following fields and settings:

- Innovator Server:** Server: `http://localhost/Innovator`, Logout button.
- Release settings:** Target Release: `AIM33`, Description: `Aras Import Manager 33`.
- Manifest File:** Path: `C:\Aras Downloads\Aras.AIM.14.0.1\imports\imports.pe.mf`.
- Available for Import:** ☒ Select All Packages. List: `com.aras.ark.im.pe`.
- Language codes:** ☒ en.
- Existing Packages in the Database:**
 - com.aras.defaults.filerepresentations
 - com.aras.innovator.access.common
 - com.aras.innovator.access.domain
 - com.aras.innovator.access.mandatory
 - com.aras.innovator.access.uservisibilit
 - com.aras.innovator.admin
 - com.aras.innovator.authenticationfram
 - com.aras.innovator.client.core
- Type:** ☒ Merge, ☐ Ignore.
- Mode:** ☐ Fast Mode, ☒ Thorough Mode.
- Status bar:** completed.

- Input the connection information.
 - **Server:** The connection URL for Aras Innovator

Click the **Login** button and enter:

- **Database:** The target Aras Innovator database
- **Username:** root
- **Password:** Password for “root” login (Default is “innovator”)
- **Target Release:** AIM33
- **Description:** Aras Import Manager 33
- **Manifest File Path:** The manifest file `\Imports\imports.pe.mf`
- **Available for Import:** Select All Packages
- **Type:** Merge
- **Mode:** Thorough Mode



- Click the **Import** button.
- Disable **Super User (root)** login.

Important

The **Super User (root)** login should not be enabled in production.

Configure instance

Update method-config.xml

- Open %InstallPath%\Innovator\Server\method-config.xml .
- Check that the content of `ReferencedAssemblies` is similar to the following content.
- Lines highlighted in yellow have been added
- Check the lines highlighted in yellow and red text. If any already exist, don't add them.

```
<ReferencedAssemblies>
<name>$(binpath)/Aras.Server.Core.dll</name>
<name>$(binpath)/Aras.Server.Configurator.Abstractions.dll</name>
<name>$(binpath)/Aras.Server.Workflow.Abstractions.dll</name>
<name>$(binpath)/Aras.Server.Lifecycle.Abstractions.dll</name>
<name>$(binpath)/Aras.TDF.Base.dll</name>
<name>$(binpath)/Aras.TDF.Base.Extensions.dll</name>
<name>$(binpath)/Conversion.Base.dll</name>
<name>$(binpath)/ConversionManager.dll</name>
<name>$(binpath)/FileExchangeService.dll</name>
<name>$(binpath)/IOM.dll</name>
<name>$(binpath)/Newtonsoft.Json.dll</name>
<name>$(binpath)/SixLabors.ImageSharp.dll</name>
<name>$(binpath)/Microsoft.Data.SqlClient.dll</name>
<name>Microsoft.Extensions.Logging.Abstractions.dll</name>
<name>Microsoft.Extensions.Primitives.dll</name>
<name>$(binpath)/Aras.Server.Core.Abstractions.dll</name>
<name>$(binpath)/Aras.Server.AccessControl.Abstractions.dll</name>
<name>$(binpath)/Aras.Server.MetadataCache.Abstractions.dll</name>
<name>$(binpath)/System.Runtime.Caching.dll</name>
<name>$(binpath)/Aras.Ark.ImportManager.dll</name>
<name>$(binpath)/Aras.Ark.ImportManager.CustomReader.dll</name>
</ReferencedAssemblies>
```



Configure converter

To make the conversion server work, we need to first configure all the converters we are going to use following these steps :

- Open %InstallPath%\ConversionServerConfig.xml
- Check that the content of ReferencedAssemblies is similar to the following content.
- Line highlighted in yellow has been added

```
<?xml version="1.0" encoding="utf-8"?>
<configuration>
<configSections>
<!-- Common converter service configuration -->
<section name="ConversionServer" type="Aras.ConversionFramework.ConversionServer.Configuration.ConversionServerConfiguration, Aras.ImportManager" />
<sectionGroup name="ConverterSettings">
<!-- Place here class configuration section definitions for converters -->
</sectionGroup>
</configSections>
<ConversionServer>
<InnovatorServer url="http://localhost/Innovator/Server/InnovatorServer.aspx"/>
<Converters>
<Converter name="AIM_Import_GenericConverter" type="Aras.Ark.ImportManager.Conversion.GenericConverter, Aras.ImportManager" />
</Converters>
</ConversionServer>
<ConverterSettings>
<!-- Place here configuration sections for converters -->
</ConverterSettings>
</configuration>
```

Configure conversion rule

In order to have Aras Import Manager background tasks working correctly, we need to configure the conversion rules following these steps :

- Open %InstallPath%\AgentService\conversion.config
- Check that the content of ReferencedAssemblies is similar to the following content.
- Line highlighted in yellow has been added :
 - Replace DB value with you instance name (e.g.: PROD_ImportManager)



```

<configuration>
<configSections>
<section name="Innovator" type="Aras.Server.Configuration.InnovatorConfigurationSection, Aras.Server.Agent" />
</configSections>
<Innovator>
<Conversion>
<ProcessingCycles>
<!--cycle DB="{db}" User="{user}" Interval="20000" MaxBatch="5" /-->
<!--cycle DB="{db}" User="{user}" Interval="20000" MaxBatch="1" ConversionRuleName="SSVC.Email.Digest.Notification" /-->
<!--cycle DB="{db}" User="{user}" Interval="5000" MaxBatch="50" ConversionRuleName="SSVC.Notification.Rule" /-->
<!--cycle DB="{db}" User="{user}" Interval="5000" MaxBatch="50" ConversionRuleName="SSVC.UpdateNotification" /-->
<cycle ConversionRuleName="AIM_Import_GenericConversionRule" DB="<your_database_instance>" Interval="10000" />
</ProcessingCycles>
</Conversion>
</Innovator>
</configuration>

```

Restart the Agent Service

After configuring the conversion server, you may have to restart the AgentService Service in Windows services.

Confirming the Solution Installation

Use the following procedure to check if the database has correctly applied the Aras Import Manager changes to Aras Innovator:

1. Log in to Aras Innovator as an administrator.
 1. From the **TOC**, select **Administration --> Variables**.
 2. Confirm that the following Application Domain are listed:

Aras Import Manager 14.0.1

If at any time the installation fails, revert to your backups, and contact Aras Support at support@aras.com.



SaaS Installation Guide



Introduction

Purpose

The Aras Import Manager – Installation guide for Aras DevOps provides information for Aras DevOps and Enterprise subscribers to add the Aras Import Manager to their Aras Innovator instance.

Scope

This document provides instructions for installing Aras Import Manager which includes application Code Tree Patch, AML application package and configuration transformations.

Target Audience

This document is intended for Aras DevOps users responsible for installing and configuring the applications and platform components for the Aras Innovator instance utilizing SaaS.

Prerequisites

Aras DevOps administrators or contributors must have an installed Aras Innovator instance prior to following the steps outlined in this document. It is recommended that all administrators read the Aras DevOps – User Guide, as it explains how to effectively utilize the Aras DevOps service included as part of the Aras Enterprise subscription to manage Aras Innovator customizations throughout the entire lifecycle of the Aras Innovator implementation..



Definitions

This section defines the terms used in this document.

Term	Definition
Code tree	The set of files that make up the application files for Aras Innovator.
Code tree patch	The set of files in an application installation package that should be applied to the code tree to install the code tree component of an application.
Converters	A toolset included with some applications for executing asynchronous tasks on the conversion server (e.g. CAD to PDF, PDF Watermarking, etc.)
Import package	The set of XML files and the corresponding import.mf file that represent a package or set of packages that need to be installed to apply the database changes of an application.
Config transformation	The transformation to be sourced into project repository corresponding to changes in configuration files of Aras Innovator code tree
Installation package	The set of files utilized to install an application that may include a code tree patch, an import package, and/or converters.
Platform Component	Aras Platform Components are not release-specific and can be installed without a direct dependency on the installed Aras Innovator release.



Aras Import Manager Installation

Prerequisites

1. Both Conversion Server and Agent Service must be installed and configured with the instance of Aras Innovator
2. Install Aras Import Manager 33.

Retrieve Application package

1. Download the Aras Import Manager 33 CD Image zipped archive from the [Aras File Sharing](#) site, located in the `Aras/Product Releases/` folder.
2. Unzip the Aras Import Manager 33 CD Image on the local machine.

Include application Code Tree Patch into the repository

1. Navigate to the (unzipped) Aras Import Manager 33 CD Image folder on the local machine.
 1. Copy the `'Innovator'` folder into the `\CodeTree\` folder of the Work.git repository.
 2. Copy the `'ConversionServer'` folder into the `\CodeTree\` folder of the Work.git repository.

Include application package into the repository

1. Navigate to the (unzipped) Aras Import Manager CD Image folder on the local machine.
 1. Open the `'imports'` folder and copy the `'ark'` folder into the `\AML-packages\` folder of the Work.git repository.

Important

Do not copy the `imports.mf` file, as it will overwrite the existing `imports.mf` file in the Work.git repository.

2. Open the `\imports\imports.mf` Aras Import Manager CD Image folder file.
3. Copy the following `<package>` elements:



```
<imports>
  <package name="com.aras.ark.im.core" path="ark\im\core" />
</imports>
```

4. Paste the `<package>` elements into the `\AML-packages\imports.mf` file in the Work.git repository.
5. Open the `\imports\imports.standart.mf` Aras Import Manager CD Image folder file.
6. Copy the following `<package>` elements:

```
<imports>
<package name="com.aras.ark.im.standard" path="ark\im\standard">
<dependson name="com.aras.ark.im.core" />
</package>
</imports>
```

7. Paste the `<package>` elements into the `\AML-packages\imports.mf` file in the Work.git repository.
8. Open the `\imports\imports.pe.mf` Aras Import Manager CD Image folder file.
9. Copy the following `<package>` elements:

```
<imports>
<package name="com.aras.ark.im.pe" path="ark\im\pe">
<dependson name="com.aras.innovator.solution.PLM" />
<dependson name="com.aras.ark.im.core" />
</package>
</imports>
```

10. Paste the `<package>` elements into the `\AML-packages\imports.mf` file of the Work.git repository.
11. Save and close `\AML-packages\imports.mf` file in the Work.git repository.

Include Config Transformations

1. Add a configuration transformation file for the **ConversionServerConfig.xml** in the `\TransformationsOfConfigFiles\` folder of the Work.git repository with the following content:



```
<?xml version="1.0" encoding="utf-8"?>
<configuration xmlns:xdt="http://schemas.microsoft.com/XML-Document-Transform">
  <configSections xdt:Transform="Replace" xdt:Locator="XPath(/configuration/configSections)">
    <section name="oauth" type="Aras.OAuth.Configuration.OAuthSection, Aras.OAuth.Configuration"/>
    <!-- Common converter service configuration -->
    <section name="ConversionServer" type="Aras.ConversionFramework.ConversionServer.Configuration.Co
  </configSections>
  <ConversionServer>
    <Converters>
      <Converter name="AIM_Import_GenericConverter" type="Aras.Ark.ImportManager.Conversion.GenericCon
      <Converter name="MDV_ConversionType" type="MDVExporter.ConversionServer.DefaultConversionServer,
    </Converters>
  </ConversionServer>
</configuration>
```

Important

The **ConversionServerConfig.xml** is available from the Baselines Artifact feed of the SDE for reference when building configuration transformation files.

2. Add a configuration transformation file for the **method-config.xml** in the `\TransformationsOfConfigFiles\Innovator\Server` folder of the Work.git repository with the following content:



```

<MethodConfig xmlns:xdt="http://schemas.microsoft.com/XML-Document-Transform">
<ReferencedAssemblies>
<name xdt:Transform="Insert">$(binpath)/Aras.Ark.ImportManager.dll</name>
<name xdt:Transform="Insert">$(binpath)/Aras.Ark.ImportManager.CustomReader.dll</name>
</ReferencedAssemblies>
  <Template
    line_number_offset="34"
    name="CSharp:Aras.Ark.ImportManager.ValidationRule.Validator"
    xdt:Transform="Insert"><![CDATA[
using System;
using System.IO;
using System.Xml;
using System.Text;
using System.Collections;
using System.Collections.Generic;
using System.Data;
using System.Linq;
using System.Net;
using System.Globalization;
using Aras.IOM;
using Aras.Server.Core;
using Aras.I18NUtils;
using Aras.Ark.ImportManager;
using Aras.Ark.ImportManager.Model;
using Aras.Ark.ImportManager.Validation;
namespace $(pkgname)
{
    public class $(clsname)
    {
        private readonly Aras.Server.Core.CallContext CCO;
        private readonly Innovator Innovator;
        public $(clsname)(IServerConnection serverConnection)
        {
            this.Innovator = new Innovator(serverConnection);
            this.CCO = ((Aras.Server.Core.IOMConnection) serverConnection).CCO;
        }
        public Item Validate(Item context, Import import, MappedRow row)
        {
            $(MethodCode)
        }
    }
}
]]>
  </Template>

```



```

    <Template
      line_number_offset="34"
      name="CSharp:Aras.Ark.ImportManager.Doer" xdt:Transform="Insert"><![CDATA[
using System;
using System.IO;
using System.Xml;
using System.Text;
using System.Collections;
using System.Collections.Generic;
using System.Data;
using System.Linq;
using System.Net;
using System.Globalization;
using Aras.IOM;
using Aras.Server.Core;
using Aras.I18NUtils;
using Aras.Ark.ImportManager;
using Aras.Ark.ImportManager.Model;
using Aras.Ark.ImportManager.Validation;
namespace $(pkgname)
{
    public class $(clsname)
    {
        private readonly Aras.Server.Core.CallContext CCO;
        private readonly Innovator Innovator;
        public $(clsname)(IServerConnection serverConnection)
        {
            this.Innovator = new Innovator(serverConnection);
            this.CCO = ((Aras.Server.Core.IOMConnection) serverConnection).CCO;
        }
        public Item Do(Item context, Import import, MappedRow row)
        {
            $(MethodCode)
        }
    }
}
]]>
    </Template>
</MethodConfig>

```

3. Add a configuration transformation file for the **conversion.config** in the `\TransformationsOfConfigFiles\AgentService\` folder of the Work.git repository with the following content:



```
<?xml version="1.0" encoding="utf-8"?>
<configuration xmlns:xdt="http://schemas.microsoft.com/XML-Document-Transform">
  <Innovator>
    <Conversion>
      <ProcessingCycles>
        <cycle xdt:Transform="Insert" ConversionRuleName="AIM_Import_GenericConversionRule" DB="${DBName}">
        </cycle>
      </ProcessingCycles>
    </Conversion>
  </Innovator>
</configuration>
```

Configure environment variables

1. Navigate to your Implementation Project > Pipelines > Library and select + Variable group.
2. Enter a value in the Variable group name field.
3. To add a variable, click + Add and specify the following:

Name: *DBName*

Value: *\${ARAS_DB_NAME}*

4. Click Save to create the new variable group.

Test and deploy your changes

1. Stage and review the changes to ensure that no customizations are overwritten.
 1. Commit the changes.
 2. Run the “**continuous-integration**” pipeline.
 3. Run the “**deploy-innovator**” specifying “Custom config variable group name” parameter in the pipeline with the name of Variable created in step 2 of “Configure environmental variables” section to complete the installation.

Confirming the Solution Installation

Use the following procedure to check if the database has correctly applied the Aras Import Manager changes to Aras Innovator:

1. Log in to Aras Innovator as an administrator.
 1. From the **TOC**, select **Administration --> Variables**.



2. Confirm that the following Application Domain are listed:

Aras Import Manager 14.0.1

If at any time the installation fails, revert to your backups, and contact Aras Support at support@aras.com.



User Guide



Introduction

Purpose and Target Audience

This document provides detailed instructions for using the Aras Import Manager (AIM). AIM provides a user-friendly, fail-safe, and flexible approach to migrating Items from external systems to Innovator via Excel spreadsheets. This document is intended for end-users.

Import

Configure

Mapping

Staging

Validating

Imported

Import Number

IMP-0008

Name

hrthr

Status

Imported

File

AIM_WithChange_Import.xlsx

Worksheet

Change Item

ECO-100003

Header Row Position

☒ 1

Main Item Type

Part BOM

Configuration Name

BOM

Import Mode

With Change

Import Type

Parent-Child

First Row

2

Last Row

-1

Validation Result

Started On

10/9/2023 3:29:32 PM

Finished On

10/9/2023 3:29:45 PM

Lines in File

10

Correct

10

with Error

0

with Warning

0

with Note

0

Import Result

Started On

10/9/2023 3:30:46 PM

Finished On

10/9/2023 3:30:57 PM

Lines

with Success

0

with Error

0

Items

Created

10

Updated

0

Columns Mapping

Validation Results

Import Logs

Columns Mapping

Hidden

Mapping

Index	Column	Header	Target	ItemType [...]	Mapped To [...]	Property Label	Property Type	Required	Meta
1	A1	Releated ID	Child	Part	item_number	Part Number	String	☒	
2	B1	Sequence	Main Item	Part BOM	sort_order	Sequence	Integer	☐	
3	C1	Quantity	Main Item	Part BOM	quantity	Quantity	Float	☒	
4	D1	Part Number	Parent	Part	item_number	Part Number	String	☒	

< Prev

Next >

Page: 1 of 1

4 Results



Import Items

After performing the administrative configuration of the application, you can import items. For it, you should use Import item (**AIM_Import**).

Important

AIM_Import is available only for **Import Managers** identity by default.

For AIM to perform updates on items you need to:

- **Establish mapping** between columns, checked from the Excel file, and properties on Main Item Type (and/or relationships). Properties mapping change is **available only** in the Mapping stage.

Import Data from Excel

Make sure to fill the excel file like in the example:

1	Part Number	Description	Name	Classification	Control Type	Unit
2	AIM-Part-013	Part 13	Imported AIM-Part-013	Assembly	Serial	IN
3	AIM-Part-014	Part 14	Imported AIM-Part-014	Assembly	Lot	IN
4	AIM-Part-015	Part 15	Imported AIM-Part-015	Assembly	Serial	IN
5	AIM-Part-016	Part 16	Imported AIM-Part-016	Material	Serial	IN
6	AIM-Part-017	Part 17	Imported AIM-Part-017	Material	Serial	IN
7	AIM-Part-018	Part 18	Imported AIM-Part-018	Material	Serial	IN
8	AIM-Part-019	Part 19	Imported AIM-Part-019	Material	Serial	IN
9	AIM-Part-020	Part 20	Imported AIM-Part-020	Material	Serial	IN
10	AIM-Part-021	Part 21	Imported AIM-Part-021	Material	Serial	FT

To create a new **Import**:

1. Navigate in TOC to the *Import Manager > Import*
2. Create a new **Import** item.
3. Fill in the required fields:
 1. **Import number** – unique identifier from Innovator.



2. **Name** – an identifier for users.
3. **Main Item Type** – item type will be used as item after import.
4. **Import Mode** – use change item or import as-is.

Important

Note: Click **With Change** to see a new field available – *Change Item*, where you can specify a Change Item (e.g. ECO) to use for Import.

5. **Import Type** – use relationship structure or import flat structure.
 6. **Configuration Name** – use one of the existing configurations.
 7. **Worksheet** – name of worksheet.
 8. **Header Row Position** – specify a header position, so the import will locate where data should be taken as headers.
 9. **First Row** – index of the first row to be checked from the Excel file (use 2 in case you have a header as a first row).
 10. **Last Row** – index of the last row to be checked from the Excel file (use –1 to read till the end of file).
4. Press the **Save** button.

Following is the list of properties and their functions:

- **Import Number** – technical property, a sequence.
- **Name** – name of the Import item.
- **File** – excel file with data of items that will be imported.
- **Main Item Type** – type of imported items.
- **Configuration Name** – list with available configuration. It is mapped with the name from Available Import relation of Configuration Import.

Important

During the Import process, AIM will automatically get an import configuration based on parameters from the Import item. Import item will look for a configuration that matches the Configuration Name and criteria in Mode/Type/IT. You can create one generic import configuration for all Item Types if you need it with empty values for IT.

- **Import Mode**



- **With Change** – import using Change Item like ECO.
- **Ad Hoc** - is a direct change.
- **Import Type**
- **Flat** – import items without relationships.
- **Parent-Child** - import with connection parent-child
- **Change Item** – it is hidden by default. Available only if *Import Mode = With Change*. Users can select existing ECO items, otherwise it will be created automatically.
- **Worksheet** – name of sheet from uploaded Excel file. If the field is empty, the first sheet will be used from the file.
- **Header** – flag, meaning sheet has predefined labels in the first row of uploaded sheet.
- **Row Position** – row number of the header.
- **First Row**– the row number from which the import begins. It's "2" by default.
- **Last Row** – the row number where the import ends. It's "-1" by default to be not limited and read to the end of the sheet.
- **Quick LC View** – colored depending on the on states.
- **Configure**: User selects import configuration (type, mode, etc.) and data (file, sheet, etc.).
- **Mapping**: User map columns from Excel with properties from Innovator.
- **Staging**: Automatic step.
- **Validating**: Review validation results.
- **Imported**: Import succeeded, review results.
- **Validation Result** – validation summary.
- **Import Result** – Import Summary

Important

Next and **Back** buttons on the toolbar to move the Import LC without having to rely on the Promote button. Messages are not required for the promotion.

Prompt

×

Please, enter comments:

OK

Cancel



Important

Some states update in automatic mode and import need to be refreshed manually. Click the **Refresh** button to update the status from Staging to Validating.

Import has three relationships:

- **Columns Mapping** – map properties with Excel data.
- **Validation Results** – store validation results.
- **Import Log** - store import results.

Columns Mapping

This relationship allows you to set up a column mapping that you would like to import to Innovator from an uploaded Excel file:

Columns Mapping										
Index	Column	Header	Target	ItemType [...]	Mapped To [...]	Property Label	Property Type	Required	Metadata	From File
1	A1	Releated ID	Child	Part	item_number	Part Number	String	☒		☒
2	B1	Sequence	Main Item	Part BOM	sort_order	Sequence	Integer	☐		☒
3	C1	Quantity	Main Item	Part BOM	quantity	Quantity	Float	☒		☒
4	D1	Part Number	Parent	Part	item_number	Part Number	String	☒		☒

The list of properties and their description are as follows:

- **Index, Column** - extracted from the Excel file.
- **Header** - extracted from the Excel file.
- **Target** – choose which ItemType to map to. Not applicable for Main Item/Parent/Child.
- **Mapped To** – choose which Innovator property to map to.
- **Property Label** - actual label of property.
- **Property Type** - data type of the property
- **Required** - flag to indicate the property is required.
- **Metadata** - additional information for mapping and advanced control of the mapping.
- **From File** - flag to indicate which row comes from the Excel file.

Reusable/Shareable Mapping Configuration

It allows you to save a mapping in all states of importing and share it with other users. Note that saved Columns Mapping are **read-only**.

To find Saved Mapping navigate TOC to the *Import Manager > Mappings*. They are available **only** to **Import Managers** identity members by default.

MAP-00001 My First Saved Mapping ☆

Edit

Column Mapping

Item Number

MAP-00001

Name

My First Saved Mapping

Creator

[Innovator Admin](#)

Description

Shared With

[World](#)

Columns Mapping

Columns Mapping ☆

Hidden

Index	Column	Header	Target	ItemType [...]	Mapped To [...]	Property Label	Property Type	Metadata
1	A1	Related ID	Child	Part	item_number	Part Number	String	
2	B1	Sequence	Main Item	Part BOM	sort_order	Sequence	Integer	
3	C1	Quantity	Main Item	Part BOM	quantity	Quantity	Float	
4	D1	Part Number	Parent	Part	item_number	Part Number	String	

< Prev

Next >

Page: 1 of 1

4 Results

To **save or load** existing mapping, click on the **Mapping** icon on the Import Item/Column Mapping relationship:

Columns Mapping

Rows Validation

Import Log

Columns Mapping ☆

Hidden

Mapping

Index	Column	Header	Target	ItemType [...]	Mapped To [...]

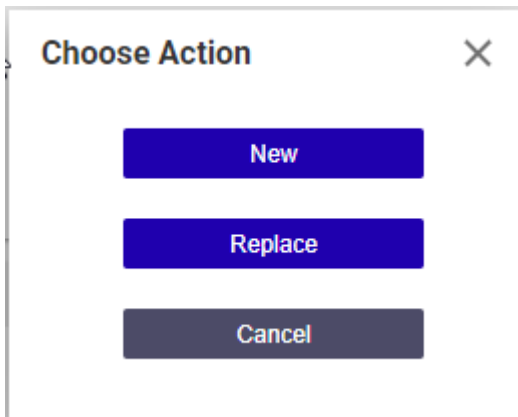
Load Mapping

Save Mapping



To create new **Mapping**:

1. Click on the **Mapping** button.
2. Click **Save Mapping**.
3. Choose a New Action.



4. Set any Name.
5. Add an Identity in *Shared With* field to share the mapping.
6. Click **Done**.

Important

After the mapping is saved, load it to Import item to avoid repeating steps. The mapping can only be loaded during the **Configure** step.

To **Load Mapping**:

7. Click on the Mapping action under the Columns Mapping tab of the Import item.
8. Click **Load Mapping**.
9. Find and select the necessary saved Mapping.
10. Press the **OK** button.
11. Save the import item to apply changes.

Rows Validation and Import Log

Validation Result and Import Logs inform about incorrect import configuration or mapping and help to fix the issues.



Columns Mapping Rows Validation Import Log			
Logs + × 🔍 🔍 Hidden 🔍 🖨 🔗			
Row ↑ 1	Severity ↑ 2	Message	Description
132	Fatal	Add to Change	Only Items in Released can be added to MCO.
133	Fatal	Add to Change	Only Items in Released can be added to MCO.
134	Fatal	Add to Change	Only Items in Released can be added to MCO.

The list of the properties and their description are as follows:

1. **Row** – Excel file row number.
2. **Severity** - level of error/log.
3. **Message** - either “Validation Rule” name or Import Manager step where the error occurred.
4. **Description** - short text about the cause of error.
5. **Details** (only on import log) – more information about the result/log.

Important

By default, the Log level is not processed.

Validation Levels

- **Fatal**: Unmanaged error, needs Administrator’s attention.
- **Error**: Validation rule with Error level failed, will block the full import if unsolved.
- **Warning**: Validation rule with Warning level failed, will skip the line if unsolved.
- **Info**: Validation Rule with Info level failed, only to inform the user, will not impact the import.
- **Log**: Validation Rule with Log level failed or additional log information from the system. No impact on the import.

Import Log Levels

- **Fatal**: Unmanaged error, either:
 - from the import logic, needs Administrator’s attention. Can cause unstable result.

- from an action on an item that caused an issue. The line is skipped, import proceeds with the rest of the data.
- **Error:** Pre-Validation Rule of Error level failed, no rows will be processed.
- **Warning:** Pre-Validation Rule of Warning level failed; the row will be skipped.
- **Info/Log:** No impact on the import.



Features

Import Summary

After finishing Validation with any result (Success, Error) you will get a section with information about processed lines with count of items with Success, Error, Warning or Note. *The Validation Result* section is shown on screen below:

Validation Result	
Started On	10/9/2023 8:58:38 AM
Finished On	10/9/2023 8:58:50 AM
Lines in File	10
# Correct	10
# with Error	0
# with Warning	0
# with Note	0

After finishing Import with any result (Success, Error) you will get section with information about processed lines with count of items used for importing successfully or not. Also, you will get information about the count of items that were created or updated:

Import Result	
Started On	10/9/2023 8:59:51 AM
Finished On	10/9/2023 8:59:55 AM
Lines	
# with Success	0
# with Error	0
Items	
# Created	0
# Updated	0

Import Types and Modes



AIM allows Import Items into Innovator from Excel files. It also allows users to import two **Import Types** of data:

- **Flat.**
- **Parent-Child.**

User items can be imported through two **Import Modes**:

- **AdHoc.**
- **With Change.**

AIM provides two Configurations by default:

- **Part.**
- **Part BOM.**

These two configurations allow AIM to work with different Import Types and represent Methods for Work with Change Management. Every import should be created using a combination of:

- **Import Type** (Flat, Parent-Child), and
- **Import Mode** (AdHoc, With Change).

Description of Import Mode/Type combinations:

Type		
	Flat	Parent Child
Mode		
AdHoc	• Imports Items as-is into Innovator, as users can add them manually via UI. • Main ItemType can be any Item Type. • AIM creates/updates items based on the data from the file. • If an item already exists in Innovator, then AIM updates the existing item.	• Imports relations between items. • Main ItemType for this import should be an ItemType representing Relationship. • AIM creates/updates/deletes items based on the data from the file. • Imported item will be used as relationship and connect parent and child items. (e.g. Part BOM will connect parent Parts to child Parts).



With Change

- Extends *Flat* behavior but item will be added to Change Item as Affected Item with Add or Change (when item is in *Release* state) action.
- If item already exists in Innovator with properties, then Import Manager updates existing item.
- Extends *Parent Child* behavior but item will be added to *Change Item* as Affected Item with Add or Change (when item is in *Release* state) action.
- Every relationship item will be added to the item before assignment.

Important

By default, the represented Part BOM configuration for *With Change* works with *Simple ECO*.

When creating import, and promoting it to the *Mapping* state, required item type properties should be associated with columns from the Excel file.

For example, after importing an Excel file with data as in the image below:

	A	B	C	D	E
1	Releated ID	Sequence	Quantity	Part Number	
2	AIM-Part-004	1	1	AIM-Part-001	
3	AIM-Part-005	2	2	AIM-Part-001	
4	AIM-Part-006	3	3	AIM-Part-001	
5	AIM-Part-007	1	5	AIM-Part-002	
6	AIM-Part-008	2	6	AIM-Part-002	
7	AIM-Part-009	3	7	AIM-Part-002	
8	AIM-Part-010	1	8	AIM-Part-003	
9	AIM-Part-011	2	9	AIM-Part-003	
10	AIM-Part-012	3	10	AIM-Part-003	
11	AIM-Part-003	4	11	AIM-Part-003	
12					

AIM in importing state reads headers and creates Column Mappings:



Columns Mapping ☆

Hidden

Mapping

Index	Column	Header	Target	ItemType [...]	Mapped To [...]	Property Label	Property Type	Required	Metadata	From File
1	A1	Related ID	Child	Part	item_number	Part Number	String	☒		☒
2	B1	Sequence	Main Item	Part BOM	sort_order	Sequence	Integer	☐		☒
3	C1	Quantity	Main Item	Part BOM	quantity	Quantity	Float	☒		☒
4	D1	Part Number	Parent	Part	item_number	Part Number	String	☒		☒

< PrevNext >Page: 1 of 14 Results

On this image Header values were taken from excel file from row specified in Header Row Position, highlighted on the image below:

Import

Configure

→

Mapping

→

Staging

→

Validating

→

Imported

Import Number

IMP-0010

Name

Test

Status

Mapping

File

AIM_WithChange_Import.xlsx

Worksheet

Main Item Type

Part BOM

Configuration Name

BOM

Header Row Position

☒ 1

Import Mode

Ad Hoc

Import Type

Parent-Child

First Row

2

Last Row

-1

After starting mapping, the headers can be associated with Innovator properties. To select property, firstly choose an item type which contains property. It is allowed by choosing *Target* on relationship. *Target* represents an item that will be updated at the end of the import.

Flat

Main Item

- Properties from this Target will be used in create/update process with type taken from **Main Item Type**.

Parent Child

- Properties from this Target will be used in create/update process with type taken from **Main Item Type** and will use *source_id* and *related_id* to create association between items.



