Aras Office Connector 24

Administration Guide

Document #: 24.0.2022031501
Last Modified: 11/4/2022

Copyright Information

Copyright © 2022 Aras Corporation. All Rights Reserved.

Aras Corporation

100 Brickstone Square

Suite 100

Andover, MA 01810

Phone: 978-691-8900

Fax: 978-794-9826

E-mail: support@aras.com

Website: https://www.aras.com

Table of Contents

Send Us Your Comments

Aras Corporation welcomes your comments and suggestions on the quality and usefulness of this document. Your input is an important part of the information used for future revisions.

- o   Did you find any errors?
- o   Is the information clearly presented?
- o   Do you need more information? If so, where and what level of detail?
- o   Are the examples correct? Do you need more examples?
- o   What features did you like most?

If you find any errors or have any other suggestions for improvement, indicate the document title, and the chapter, section, and page number (if available).

You can send comments to us in the following ways:

Email:
TechDocs@aras.com
Subject: Aras Product Documentation

Or,

Postal service:
Aras Corporation
100 Brickstone Square
Suite 100
Andover, MA 01810
Attention: Aras Technical Documentation

If you would like a reply, provide your name, email address, address, and telephone number.

If you have usage issues with the software, visit https://www.aras.com/support/

4

Document Conventions

The following table highlights the document conventions used in the document:

Convention	Description
Bold	This shows the names of menu items, dialog boxes, dialog box elements, and commands. Example: Click **OK**.
`Code`	Code examples appear in `courier` font. It may represent text you type or data you read.
`Yellow highlight`	Code highlighted in yellow draws attention to the code that is being indicated in the content.
`Yellow highlight with red text`	Red text highlighted in yellow indicates the code parameter that needs to be changed or replaced.
Italics	Reference to other documents.
Note:	Notes contain additional useful information.
Warning	Warnings contain important information. Pay special attention to information highlighted this way.
Successive menu choices	Successive menu choices may appear with a greater than sign (-->) between the items that you select consecutively. Example: Navigate to **File** --> **Save** --> **OK**.

1 Overview

Microsoft Office applications are a big part of most manufacturing companies' product development process. For example, Microsoft Word is used to create product specifications, Microsoft Excel for product-related financial spreadsheets, Microsoft PowerPoint for product briefings, and Microsoft Visio for product development flowcharts. Microsoft Outlook is used for emails containing vital product and project information.

The Aras Office Connector provides a convenient way to manage Microsoft Office documents in Aras, directly from the user interface of the Office applications. The connector helps to standardize company documentation procedures by providing easy access to template documents, controlling numbering and naming of files, and automating creation of viewable files like PDF.

Administration of the Aras Office Connector is straightforward and contained completely within the Administration section of the Aras Innovator web client. The Connector is designed to be simple to use, and one aspect of this is that all options are controlled by the administrator on a site-wide basis. This helps to control corporate processes and keeps the operation simpler for the user.
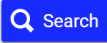
This *Administration Guide* assumes a basic knowledge of Aras configuration, including ItemTypes, Lists, and Methods. For more information on these topics, please see the Just Ask Innovator documentation which can be accessed from within Aras Innovator by clicking **Help → Just Ask Innovator**.

It is recommended that all the administrators of the Aras Office Connector also read the *Aras Office Connector 24 - User Guide*, as it explains how the administrator settings affect the user interface and has in-depth instructions on topics like template management and Office file property imbedding, which are both user and administrative related topics.

2 Aras Office Connector Preferences

Aras Office Connector preference settings are managed within Aras Innovator, in the Administration section of the TOC.

To access these settings, use the following procedure:

1. Login to Aras Innovator as an administrator.

2. Navigate to **Administration → Preferences**.

3. Click the **Search** icon [Q Search] to execute a search and display a list of Identities.

4. In the search grid, double click on the row containing the **World** identity. This opens the **Preferences** dialog for Aras Innovator.
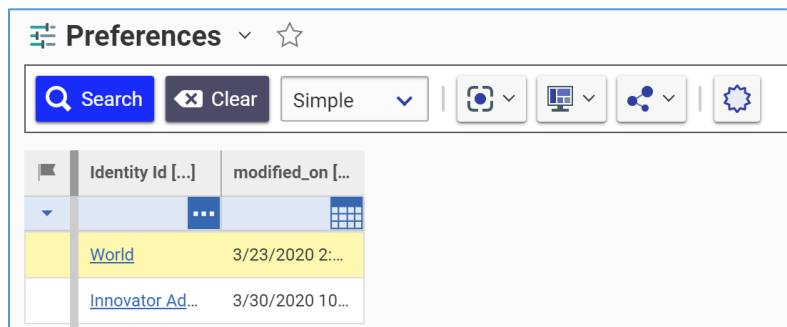
5. Click on the **Office Preferences** tab, and then within the tabular area at the bottom double click on **Office Connector Settings**. This opens the **Office Connector Settings** webpage dialog.
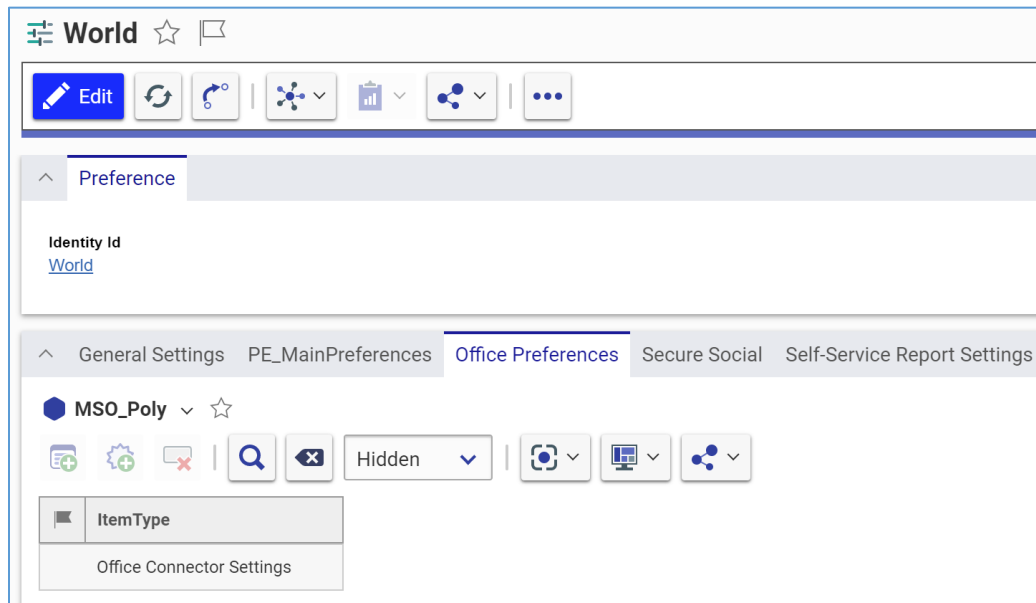
2.1 General Settings

General settings are in the top area of the dialog and control some general behaviors of the Aras Office Connector.
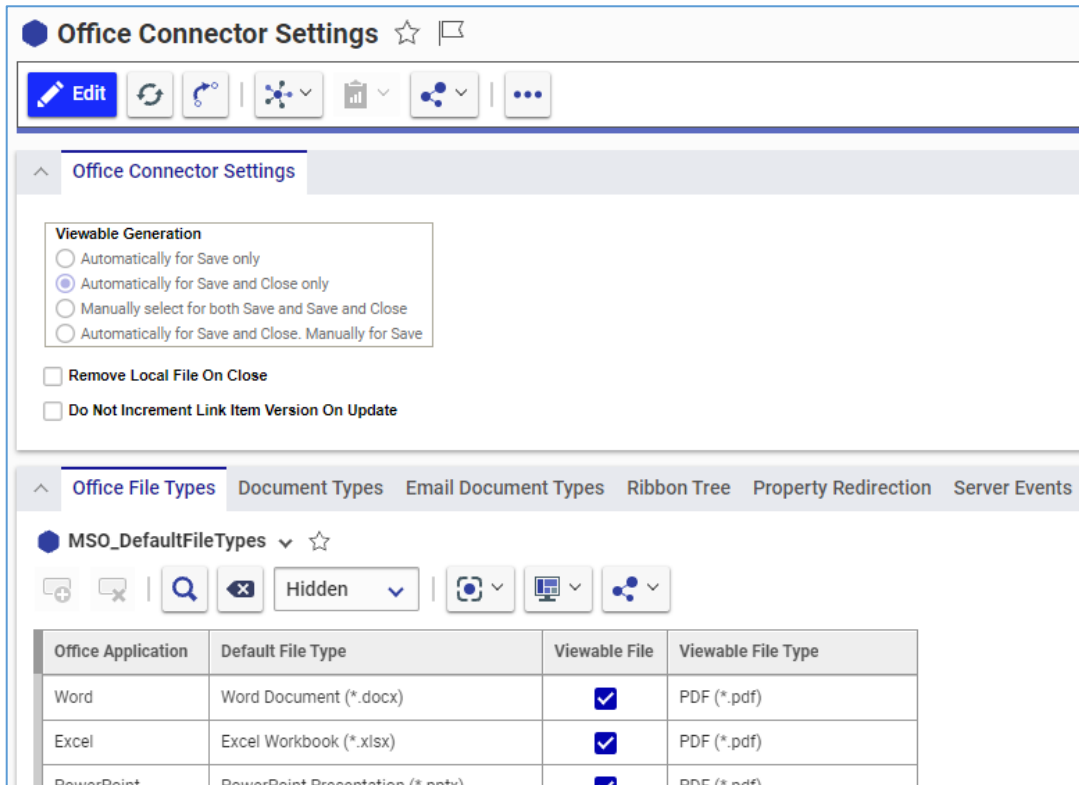
- **Viewable Generation** – Automating the creation of viewable files, like PDF and PDF/A, from Office applications is one part of the Aras Office Connector functionality. There are four options that enable you to determine when viewable files get generated. If you select the first or second option, the viewable generation is always handled automatically by the system. If you select the third or fourth option, a checkbox is added to the Aras ribbon that enables you to choose whether to generate the viewable file. All the options are summarized in the Behavior Table.

- **Remove Local File On Close** – Setting this option to ON will cause the local file to be removed from disk after being saved to Aras, during the **Save and Close** command. This helps to keep the users' local disks clean and promotes the use of Aras as the source for their files.

- **Do Not Increment Link Item Version On Update** – Setting this option to ON will not increment the generation of an item when it is linked to the document.

Viewable Generation and **Remove Local File On Close** pertain to the **Save and Close** command. The following is a table containing a complete summary of the behaviors possible with **Save and Close**, as compared to a normal update save using **Save to Aras**.

Behavior Table

Behavior	Update Save	Save and Close
File saved to Aras	Yes	Yes
Reservation Claim	Claim is kept ON (claimed)	Claim is set to OFF (unclaimed)
File in session?	File is kept in active Office session	File is closed from active Office session
Viewable creation (PDF, PDF/A, etc.)	Option 1: Viewable created Option 2: No viewable created Option 3: Viewable created if selected by user Option 4: Viewable created if selected by user	Option 1: Viewable created Option 2: Viewable created Option 3: Viewable created if selected by user Option 4: Viewable created
Remove local file	No	Optional (set by administrator)
Version comment	No	Optional (only if enabled for the Document Type)
Promote to state	No	Optional (only if enabled for the Document Type)

2.2 Office File Types

The first tab within the Office Connector Settings dialog is **Office File Types**. This tab defines some behaviors for the five supported Microsoft Office Applications: Word, Excel, PowerPoint, Visio, and Outlook.
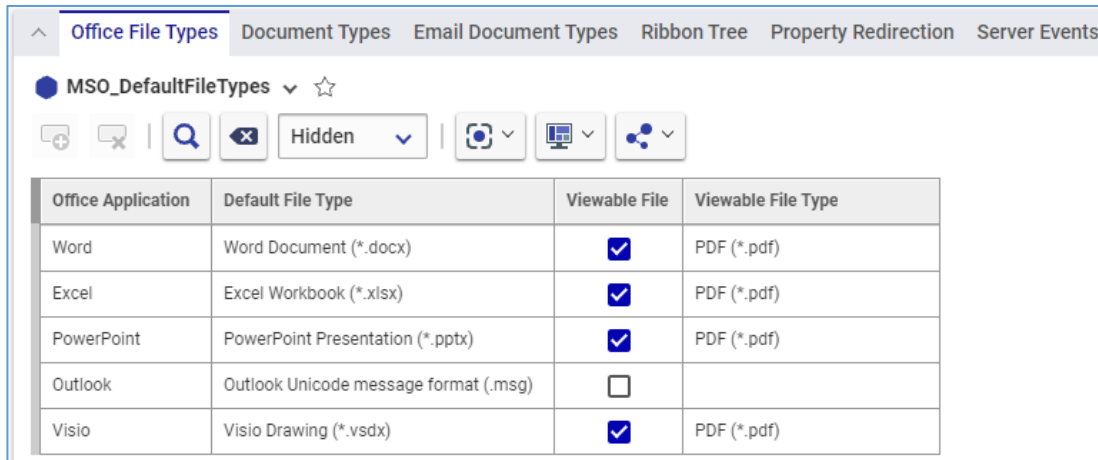
Office Application – Must be set to Word, Excel, PowerPoint, Visio, or Outlook. No other Office applications are currently supported.

Default File Type – This sets the file type that will be saved into Aras for each Office application. There is a pre-defined list of options for each application. For example, Word has these options:

- Word Document (*.docx)

- Word Macro-Enabled Document (*.docm)

- Word 97-2003 Document (*.doc)

By default, this is set to "Word Document (*.docx)", the standard format used by Word. You could set it to "Word 97-2003 Document (*.doc)", which forces all files stored in Aras to be saved in the backward-compatible format.

Viewable File – This controls whether viewable files are created for the application when saving to Aras. If this is enabled, then the times at which the viewable file is created is controlled by the Viewable Generation option described in section 2.1.

Note: When you enable the Viewable File option for Outlook, it means the corresponding viewable file will be created every time an Outlook email is saved to Aras. The **Viewable Generation** setting under General Settings has no effect.

Viewable File Type – This sets the file type of the viewable file created for the application. By default, this is set to PDF for all applications, but the file type can be changed to any of the pre-defined options.

2.3 Document Types

The second tab within the Office Connector Settings dialog is **Document Types**. A "Document Type" refers to the combination of an ItemType and Classification used to manage an Office file. Any Aras ItemType (not just Document) can be used to manage Office documents, if it is properly configured here in these settings. It is important to understand that the standard settings provided in this tab are examples only; it is expected that they will be configured to match requirements for your organization.

The standard settings refer to four Classification values of the Document ItemType as shown here. These Classification values (Ad-hoc, General, Product, and Project) are automatically included on the Document ItemType in the standard release.

Within the tabular view you can set the **Order** property to control the sort order for the Document Types. The sort order is the order in which users see the Document Types listed when they create new documents from the Aras Office Connector.

For the rest of the Document Type settings, it is more practical to edit them within a form rather than the tabular layout:

- To edit an existing Document Type, in the relationship grid, right-click to display a context menu. Select **Document Types > Open**

- To create a new Document Type and edit it, click the **Add Row** button, select the new row, and select **Document Types > Open**

You will then see the Office Document Type form.

Label – This is the name of the Document Type that users see. It can be any text string that makes sense for your company. By default, the value of Label will be automatically set as the combination of Item Type and Classification (as shown above), but you can change this by just typing something else in the Label field.

Item Type – The name of the base ItemType used to manage Office documents. This can be any ItemType in your system.

Classification – (Optional) A specific classification type if you want to use classification to differentiate different Document Types. If you leave this blank, the base ItemType is used without a specific class value.

Native File – Specifies the reference to the location of the native Office file (.docx, etc.) relative to the base ItemType. This can be either a relationship to File, a null relationship with an item property with a source File, or a property of type Item with source File. The value used to specify this is an Office Document Reference; see section 2.8 for details.

Viewable File – This specifies the reference to the location of the viewable Office file (.pdf, etc.) relative to the base ItemType.

Base Document Type – (Optional) Shows the name of another Office Document Type which is used as the basis for Property Mapping and Valid Linked Items settings. See section 2.3.2 for details.

Template Filter Method – This allows you to set the desired logic to determine which templates are available for selection by the user during the document creation process. The Template Filter Methods provided by default are intended to give administrators some examples of how to set this up. For example, the MSO_ReleasedTemplateFilter method only allows users to select Released templates during creation.

Generate Thumbnail– This controls whether the thumbnail will generate on the Document item form in Aras Innovator each time the item is saved.

Template Required? – This controls whether the user will be forced to select a template stored in Aras as the basis for the new document they are creating when using the **New Aras Document** command.

The idea is that certain Document Types should require the use of templates, thus standardizing the document creation process in the organization.

Linked Item Required? – This controls whether the user will be forced to select a linked item when creating a new document of this type. The idea is that certain Document Types should require the use of a linked item, such as a Part or Project, because the document should not exist without a relationship to this item.

Allow Promote – This controls whether the **Promote** function is available for the Document Type. This should only be enabled if the ItemType corresponding to the Document Type allows manual promotion and is not under the control of a change item like ECO or DCO. If enabled, the **Promote** function will be made available both in the Aras ribbon (for manual use) and as part of the **Save and Close** process, for promoting after work is complete.

Allow Select Existing Document – When this checkbox is selected, users can select an existing document item, create a new file, and link the two instead of creating a new document of a particular type. Users can then use the document number associated with the existing document item instead of generating a new document number.

Note: If this checkbox is not selected, users will not be able to use an existing document.

Available To – This specifies a single Identity that has access to the Document Type. Any user who belongs to this Identity will be able to create documents of this type, otherwise the user will not see the corresponding Document Type when creating a new document.

Document Numbering – This allows you to set one of three options for controlling the numbering of Aras Documents used to manage Office files – **Method**, **Sequence**, or **Manual**. Method means that you will define an Aras Method to define the logic for the Document Number. Sequence means you are going to directly use an Aras Sequence. Manual allows end users to enter the Document Number at the time of creation. If you select Sequence, you must enter the desired Sequence in the next field. If you select Method, you must enter the desired Method in the following field.

Document Numbering Sequence – If "Document Numbering" is set to Sequence, then this field must include the name of the Aras Sequence to be used to define Document Numbers.

Document Numbering Method – If "Document Numbering" is set to Method, then this field must include the name of the Aras Method to be used to define Document Numbers. This approach provides more flexibility than the Sequence approach since any type of desired logic can be used. The Document Numbering Methods provided by default are intended to give administrators some examples of how to set this up. For example, the Method **MSO_Product_Document_Numbering** includes the following logic:

- Determine if the Document is being linked to a Part (a valid Linked item).

- If YES, the Document Number will be of the format P12345-DOC-1

    o   Where "P12345" is the Number of the linked Part

    o   Where "-DOC-" is a fixed text string

    o   Where "1" is a sequential integer starting from 1. If a second Document is later linked to the same Part, then its Document Number will be P12345-DOC-2.

- If NO, the Document Number will be set using the Default Document sequence.

You can see that the Method approach provides additional logic that allows the Document Number to be based on the Part Number, if a linked Part exists.

Rename File? – This setting controls whether the Office file being saved into Aras (or being newly created) will be renamed, or just left as-is. Valid options are:

- **No** – If saving an existing file, the file will not be renamed, and if creating a new file, the user will be prompted to enter a file name.

- **Method** – An Aras Method will be used to define the logic for renaming (or naming) the file. If this setting is used, the name of the Aras Method must be defined in the File Naming Method field.

- **Match Document** – The filename will match the Number assigned to the Document, based on the Document Numbering approach described above.

File Naming Method – If **Rename File?** is set to Method, then this field must include the name of the Aras Method to be used to define the file name. Just like using "Method" for Document Numbering, this approach provides more flexibility. The File Naming Methods provided by default are intended to give administrators some examples of how to set this up. For example, the Method **MSO_Product_File_Naming** includes the following logic:

- Retrieve the value of the Document Number and append the dash character and the entered Document Title (if any) to it. Some examples:
    - Document Number = P12345-DOC-1 so File Name = P12345-DOC-1-MyTitle
    - Document Number = D54321 so File Name = D54321-MyTitle

So, you can see that the Method approach provides additional logic that allows the file name to be more descriptive.

2.3.1 Default Document Type

A **Default Document Type** has its Item Type value set, but the Classification value is not set. This is used to manage files that were previously stored manually in Aras. For example, an administrator uses the standard Aras "Document" ItemType and has ten different classification values that are already in use in existing Documents (not created by the Aras Office Connector). But in the future, the admin wants to use only specific Document Types in Aras Office Connector for 3 of those classifications. This means that if a user opens a Document that is set to any of the other 7 classification values, it will be opened as a "Default" Document Type. This means that it will follow the defined behaviors of the Default Document Type as far as property mappings, valid linked items, etc.

Essentially this saves time, so you don't have to explicitly define Document Types for every existing classification. Any document with a classification value that does not match a specific Document Type will be treated as "Default".

2.3.2 Base Document Type

Using a **Base Document Type** is a way to simplify the definition of Property Mappings and Valid Linked Items, in the case where most or all these definitions are shared with many Document Types. To set a Base Document Type, simply pick an existing Document Type from the Base Document Type property. In the following example, the Document Type "Document / Miscellaneous" is using a Base Document Type of "Document Base". All Property Mapping definitions and Valid Linked Items defined for "Document Base" are automatically used by the "Document / Miscellaneous" Document Type.
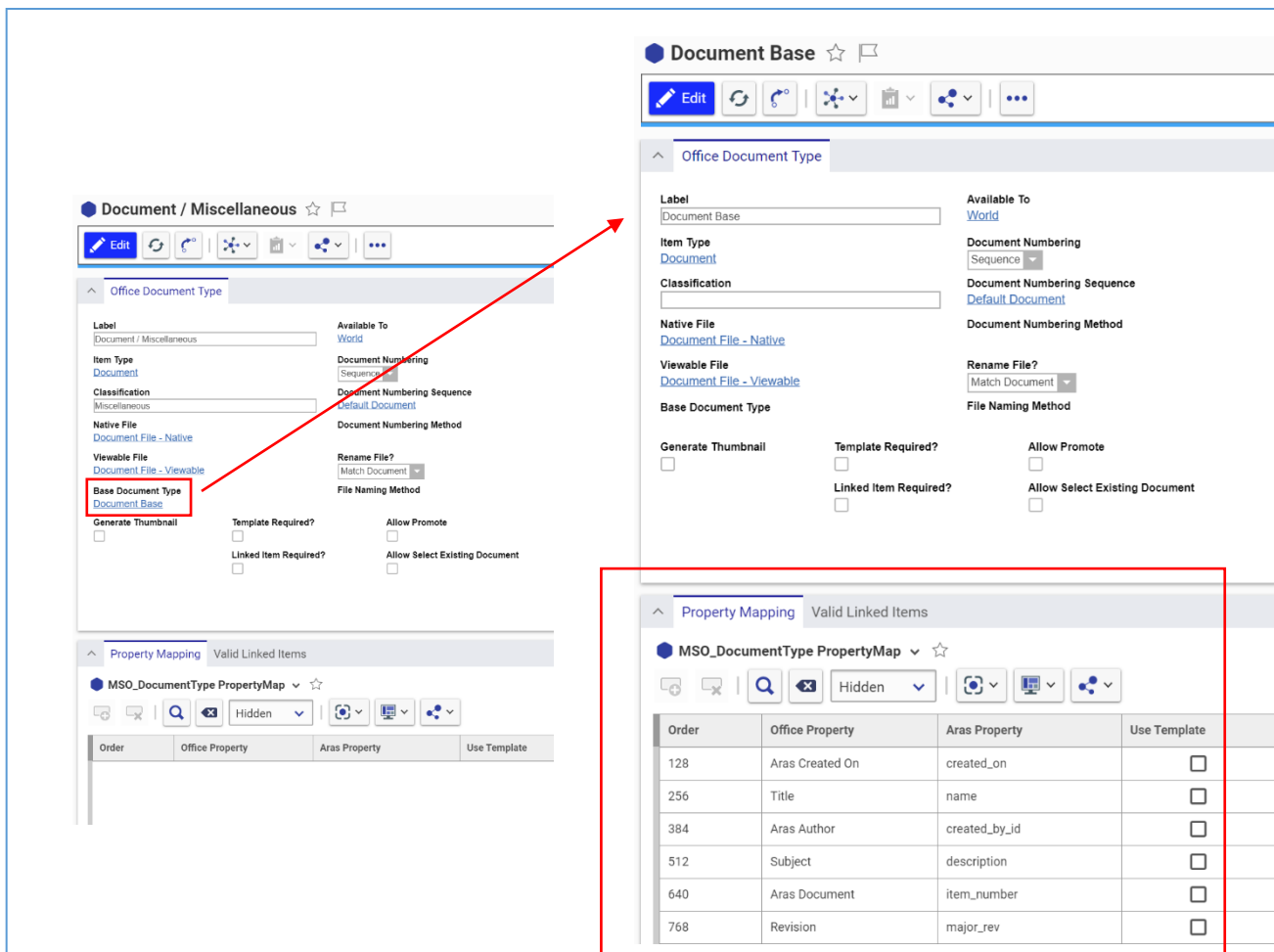
The following logic applies when using a Base Document Type:

- All settings in the Document Type header must be set directly and are not inherited from the Base Document Type.

- Any Property Mappings set on the current Document Type are underline additive to the Property Mappings set on the Base Document Type. In the case of duplicates, the settings on the current Document Type override those on the Base Document Type.

- Any Valid Linked Items set on the current Document Type are underline additive to the Valid Linked Items set on the Base Document Type. In the case of duplicates, the settings on the current Document Type override those on the Base Document Type.

- The Base Document Type must be the same ItemType as the referencing Document Type.

- Only one level of reference is allowed (i.e., no recursive definition).

Note that the concepts of **Base Document Type** and Default Document Type can be combined. This is achieved simply by defining a Base Document Type to have no value for Classification.

2.3.3 Attaching an Office File to an Existing Aras Innovator Document

Note: You must check the Allow Select Existing Document checkbox on the appropriate Office Document Type before you can use this feature.

Use the following procedure to attach a file to an existing Aras Innovator document:

1. Click **Aras> New Aras Document** in the ribbon. The New Aras Document dialog appears:

2. Select the document type from the dropdown list in the **Document Type** field.

3. Click the ellipsis [...] next to the **Document Number** field. The Search dialog box appears.
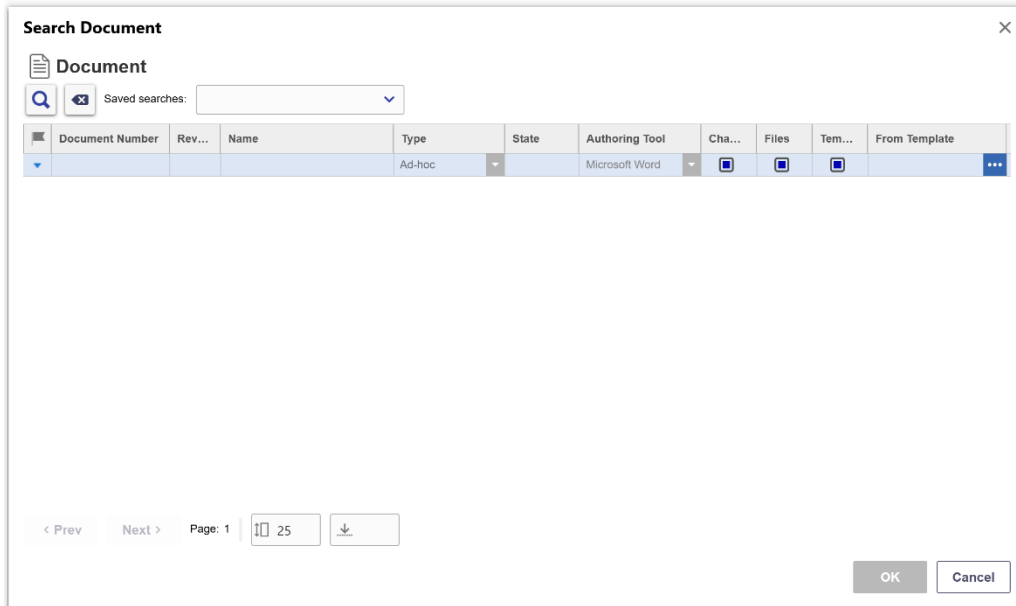
4. Click the **Search** icon to see a list of existing document items that do not have any related files.
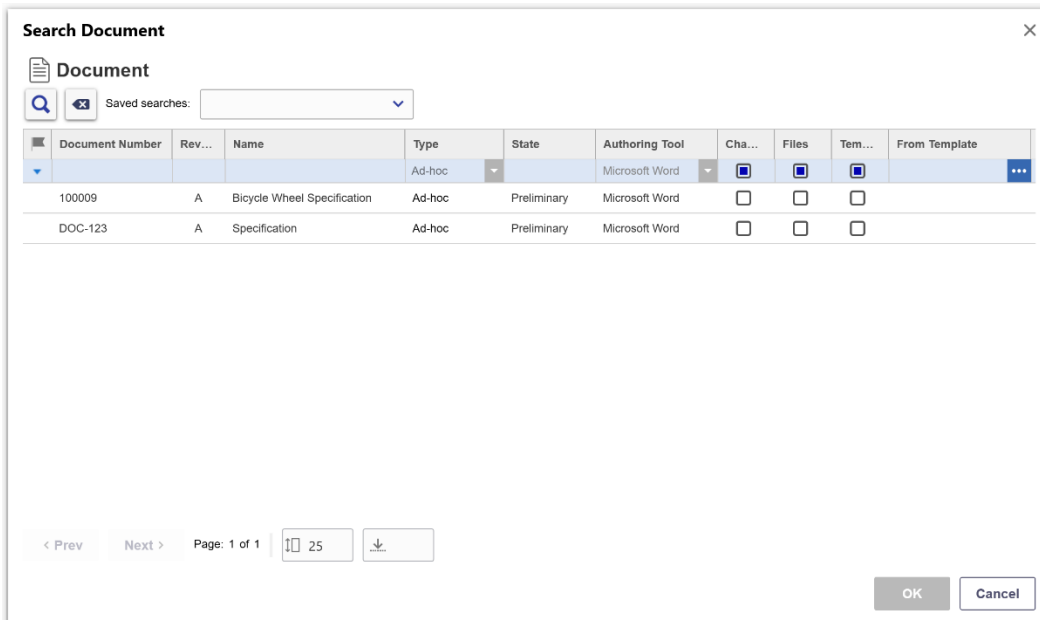
5. Select the existing document and click **OK**. The document number is applied to the new document.

2.4 Email Document Types

The third tab within the Office Connector Settings dialog is Email Document Types. This tab controls the behaviors of the Aras Office Connector based on defined Document types, specifically for saving email messages from Outlook. It is important to understand that the standard settings provided in this tab are examples only as with Document Types you should configure them to match the requirements for your organization.

Note: Even though the standard settings only include one "Email" class, you can define multiple classes for saving emails from Outlook, if desired. If only one class is defined, it will be automatically pre-selected in the Save dialog, which does save the user some mouse clicks.

As with Document Types, you can right-click on an entry, then pick **Email Document Types > Open** to open the form:

Settings for Email Document Types are largely the same as for Document Types, with these three differences:

Message File – This specifies the reference to the location where the email message file will be saved, relative to the base ItemType. This can be either a relationship to File, a null relationship with an item property with source File, or a property of type Item with source File. The value used to specify this is an Office Document Reference; see section 2.8 for details.

Viewable File – This specifies the reference to the location where a viewable (e.g., PDF, PDF/A) version of the email message file will be saved, relative to the base ItemType. This can be either a relationship to File, a null relationship with an item property with source File, or a property of type Item with source File. The value used to specify this is an Office Document Reference; see section 2.8 for details.

Attachment Files – This specifies the reference to the location where any email attachment files will be stored, relative to the base ItemType. This can be either a relationship to File or a null relationship with an item property with source File. It cannot be a property of type Item with source File, since potentially multiple files can be stored. The value used to specify this is an Office Document Reference; see section 2.8 for details.

Property for Message – In addition to saving the email message body as a Message File or Viewable File as mentioned above, the Aras Office Connector can also optionally save the message as a formatted text property on the base ItemType. If this is desired, the name of the property to be used must be entered here, and if blank it is disabled. There are two reasons for wanting to include the message body in a property: directly reading the message in the form, and searchability of the message content. For the message to be searchable in Aras, the property does not need to be made visible on the Document form. So, there are two possible steps to configure the Property for Message, depending on your end goal:

1. **Step 1 – Required**: Add the property to the base ItemType. For example, edit the Document ItemType and add a property with Name = email_message and Data Type = Text. Add the name "email_message" into the dialog. The Aras Office Connector will now store the message into this property, and it can be searched in Aras. However, the property is not visible on the Document form.

2. **Step 2 – Optional**: if desired to show the message on the Form: Add the email_message property onto the Form, with a label such as "Message". The field must be defined as Formatted Text to display the content without formatting characters.

Note: The Formatted Text field type on Aras forms cannot fully display all content which may be embedded in email messages, such as images. For this reason, it may be desirable to use only step 1 above to support searchability, and have the user open the attached message or viewable file (which supports all message content) when they want to read the message.

2.5 Ribbon Tree

The fourth tab in the Office Connector Settings dialog is Ribbon Tree. It enables you to customize the Aras Office Connector commands display for the identities defined in the selected Preference. If you configure the Ribbon Tree for the World identity, the ribbon is customized for all users. Use the Ribbon Tree to configure the display for Microsoft Word, PowerPoint, and Excel.

All the Ribbon Command Names are listed in the tree view and are visible by default. If you unselect Visible, it does not appear in the Aras Office Connector ribbon for users belonging to the specified Preference. Users will notice changes when they log into Aras Office Connector and the ribbon reads the preferences set by the administrator. The ribbon refreshes using the specified preferences, customizing which commands the user can interact with. If a parent command is not visible, the associated child commands will also not be visible.

You can still change command labels within XML for additional customization to the display. For more information, refer to section 3.9 Changing Label Text in the Aras Ribbon.

2.6 Property Redirection

The fifth tab in the Office Connector Settings dialog is Property Redirection. Property Redirection can be used to define redirection for specific properties, such as "authoring_tool", which are required by the Office Connector. Property Redirection can be used in cases where a property required by the Office Connector is already being leveraged for another purpose and cannot be used simultaneously with the Office Connector.

Item Type – This is the name of the ItemType within Aras Innovator that requires property redirection.

Original Property – This is the Name of the property within the Aras ItemType definition which is the property to be redirected. This must be set to "authoring_tool". No other Aras properties are currently supported.

Redirect Property – This is the name of the property within the Aras ItemType definition which the original property will be redirected to.

2.7 Valid Linked Items

One of the capabilities of the Aras Office Connector is that it allows easy linking of Documents to other Aras items like Parts and Projects. The term "link" refers to the Aras concept of a relationship between two items. When "links" are used, the base item that contains the files is always the related (i.e., subordinate) item, related to a parent item such as a Part. **Valid Linked Items** are defined on the second tab of either the Office Document Type or Office Email Document Type form. This setting defines which other Aras ItemTypes are allowed to be used as parent items of the relationship, for the given Document Type. For example, in the standard settings only Parts can be linked items for the "Document / Product" Document Type, because this type is designed to manage documents related to Part definition (specs, test plans, etc.).
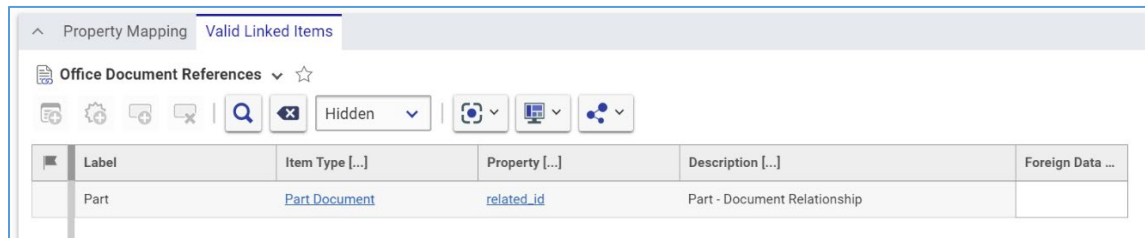
Valid Linked Items are defined separately for each Document Type, and you can have as many valid items as you want. Entries for Valid Linked Items are based on Office Document References as described in the following section.

2.8 Office Document References

Office Document Reference is an ItemType used by the Aras Office Connector to define Document Types in a flexible way. It defines the "links" between the base ItemType used by the Document Type and:

- The files managed by the Connector (both native and viewable)

- Any valid linked items (such as Part, Project, etc.)

In each of these cases, the "links" can either be:

- Item relationships (e.g., Document File, Part Document)

- Properties of type Item (e.g., the "native_file" property on CAD Document)

Office Document References are accessed from within the Administration section of the TOC.

The standard references shown here are provided to support the standard Document Types, but customers are expected to create their own to fit with the ItemTypes they are using to manage Office documents.

Office Document References contain four properties as shown on the form here:

Label – The name of the reference.

Note: When the reference is used for Valid Linked Items, this Label will be visible to users in the "Link to" selector in the New Aras Document dialog. The Label should typically be set equal to the Singular Label of the parent ItemType.

Item Type – For a relationship, this is the name of the Relationship ItemType. For a property of type Item, this is the ItemType containing the property.

Property – For a relationship, this will always be the "related_id" property. For a property of type Item, this is the name of the property.

Description – A description for the reference.

Here are some typical examples to illustrate the use of Office Document References, using standard Aras ItemTypes like "Document" and "CAD Document" to explain the common data model options:

Use Case Table

Use Case	Label	Item Type	Property
Managing the native file on the Files tab of a Document item	Document File – Native (label is not critical)	Document File	related_id
Managing the native file using the native_file property on a CAD Document item	CAD Native File (label is not critical)	CAD	native_file
Managing the native file using the CADFiles null relationship to CAD Document	CAD File (label is not critical)	CADFiles	attached_file
Linking a Part as a valid linked item to a Document	Part	Part Document	related_id
Linking a Part as a valid linked item to a CAD Document	Part	Part CAD	related_id
Linking a "CustomItem1" item as a valid linked item to a Document, where its item property "document_link" is configured to show a single linked Document	CustomItem1	CustomItem1	document_link

Once Office Document References are created, they are used in two places in the definition of Office Document Types: The links to the Native and Viewable Files, and the Valid Linked Items.

2.9 Property Mapping

Property Mapping is defined on the first tab of either the Office Document Type or Office Email Document Type form. This tab controls the property mappings that occur between the Office file and the Aras Document. Properties can be mapped in either direction and can be set to be mapped either during the initial document creation (Create) or upon subsequent saves (Update). In addition to those two mapping events, any properties mapped "From Aras to Office File" are also updated when the user clicks the Refresh button in the Aras ribbon.

Property Mappings must be defined for each Document Type separately, although by using Base Document Types you can simplify the process. A set of standard property mappings are provided with the Aras Office Connector, as shown here.

xProperties can be used in property mappings. Any Extended Classifications (xClasses) assigned to template documents will be automatically assigned to documents created from the templates, and property mappings can be configured for xProperties within the assigned xClasses.

If a property mapping includes an xProperty configuration but the document item does not include these xClasses, the property mapping will be ignored.

If a user creates a new document in Office Connector, and then selects an existing document and document template, any xClasses contained in the documents are merged.

Standard Mappings for Document Types:

Standard Mappings for Email Document Types:

Office Property – This is the name of the property within the Office file. It can be either a standard property (such as Title or Subject), or a custom property (such as Aras Author). When mapping from Office File to Aras, this property must exist in the file. When mapping from Aras to Office File, if the property does not exist it will be automatically created as a custom property.

For mapping to and from Excel Cells, this value is the name of a Named Cell.

Aras Property – This is the Name of the property within the Aras ItemType definition. It must exist for either mapping direction or an error will occur.

Use Template – If this is enabled, it means the corresponding property value from the Template Document will be set into the newly created Document. This only applies to regular Document Types, not Email Document Types. See section 0 for details.

Direction – The following table shows all valid mapping directions and which Office applications they work in:

Direction	Word	Excel	PowerPoint	Visio	Outlook
From Office File to Aras	X	X	X	X	
From Aras to Office File	X	X	X	X	
From Outlook Email to Aras					X
From Aras to Word	X				
From Word to Aras	X				
From Aras to Excel		X			
From Excel to Aras		X			
From Aras to Excel Cell		X			
From Excel Cell to Aras		X			
From Aras to PowerPoint			X		
From PowerPoint to Aras			X		
From Aras to Visio				X	
From Visio to Aras				X	

For example, to set up a mapping that maps from all MS Office applications to Aras, use "From Office File to Aras", whereas if you want the mapping to apply only for Word, use "From Word to Aras".

Create and Update – Checkboxes in these columns define on which events the mappings take place. "Create" means the initial save to Aras, and "Update" means subsequent saves to Aras.

Some important points about property mapping:

- Mapping for Outlook Emails is only in one direction since emails are saved to Aras but never opened from Aras.

- The same Office Property name can be used both for "From Office File to Aras" and "From Outlook Email to Aras." The default mapping for Subject is an example.

- Mappings from Outlook Emails are limited to properties contained within the email message header. The most useful of these are: Subject, From, To, CC, BCC, and Date.

- Mappings can be one way on Create and the other way on Update. The default mapping for Title is an example. This means that the initial Title is established from the Office file during the first save to Aras, but after that Aras is the source of any changes to the Title.

- The same property must not be mapped in both directions on the same event.

- Valid Aras property types for mapping include String, Text, Integer, Decimal, Float, List, Filter List, Multi Value List, Boolean, Date, and Item. Foreign and Federated property data types are also supported, but only in the direction from Aras to Office.

- If properties are defined as "Required" on the Aras Item Type, then the property entry field will be presented to the user with a blue background inside Aras Office Connector. The user will not be able to create or save a new document until all required properties are entered.

- If properties are defined as having a default value on the Aras Item Type, this default value will be present to the user inside Aras Office Connector. They may then change the value if they like.

> **Note:** **Classification** and **authoring_tool** property mappings are only supported from Aras to Office. They cannot be mapped **From Office File to Aras**, **From Word to Aras**, **From Excel to Aras**, **From PowerPoint to Aras**, **From Visio to Aras**, and **From Excel cell to Aras**.

2.9.1   Mapping to and from Excel Cells

The mapping directions **From Aras to Excel Cell** and **From Excel Cell to Aras** allow you to tie worksheet cell values to properties in Aras. This can be used, for example, to drive a calculation in Excel from values coming from Aras. When using these options, the value specified in the "Office Property" column is the name of a named cell in Excel. To name a cell in Excel, select the cell, right click, and select "Define Name…".

The mapping directions **From Aras to Excel** and **From Excel to Aras** map to and from Custom Properties in the Excel file, rather than cells.

2.9.2   Special Mapping Properties

Some special mapping properties are supported that perform specific functions in the connector. They are summarized in the following table:

Special Property Name	Description	Property Data Type in Aras	Valid Mapping Directions
$MSO_VersionComment	Captures the Comment text entered in the Save and Close dialog (see Note 1 below)	String or Text	From Office File to Aras From Word to Aras From Excel to Aras From PowerPoint to Aras From Visio to Aras
$MSO_EmailDescription	Captures the Description text entered in the Save to Aras dialog from Outlook	String or Text	From Outlook Email to Aras
$MSO_Participants(To)	Captures the To address(es) from the Outlook email when using Save to Aras	String or Text	From Outlook Email to Aras
$MSO_Participants(From)	Captures the From address(es) from the Outlook email when using Save to Aras	String or Text	From Outlook Email to Aras

Special Property Name	Description	Property Data Type in Aras	Valid Mapping Directions
$MSO_Participants(CC)	Captures the CC address(es) from the Outlook email when using Save to Aras	String or Text	From Outlook Email to Aras
$MSO_Participants(BCC)	Captures the BCC address(es) from the Outlook email when using Save to Aras	String or Text	From Outlook Email to Aras
$MSO_HasComments	Set ON if the current Office file has active comments	Boolean	From Office File to Aras From Word to Aras From Excel to Aras From PowerPoint to Aras From Visio to Aras
$MSO_HasChanges	Set ON if the current Office file has active tracked changes	Boolean	From Office File to Aras From Word to Aras From Excel to Aras *(not supported in PowerPoint and Visio)*
$MSO_User	Resolves to the ID of the current user's alias Identity Useful for setting owned_by_id or managed_by_id	Item (User)	From Office File to Aras From Word to Aras From Excel to Aras From PowerPoint to Aras From Visio to Aras
$MSO_String(String)	Used to set a constant text string All text inside the parentheses is take as the constant string.	String or Text	From Office File to Aras From Word to Aras From Excel to Aras From PowerPoint to Aras From Visio to Aras

Special Property Name	Description	Property Data Type in Aras	Valid Mapping Directions
$MSO_Item(ItemType, KeyedName)	Used to map the ID of a specific item **ItemType** = ItemType name **KeyedName** = KeyedName property of the desired item	Item	From Office File to Aras From Word to Aras From Excel to Aras From PowerPoint to Aras From Visio to Aras

To use any of these special properties, you must define a property with the required data type on the Item Type corresponding to the Document Type, then create a row in the Property Mapping where the Special Property Name is set in the Office Property column. For example, the following mapping sets the value of the Description text from the Outlook **Save to Aras** dialog into the "description" property on the document in Aras.

Note: If $MSO_VersionComment is mapped to an Aras property for a specific Document Type, then the Save and Close dialog will contain a Comment box to capture the user's version comment. If the property is not mapped, no such box will appear.

2.9.3 Mapping Properties from Template Documents

By checking the "Use Template" flag on a property mapping, you enable the ability to map that value from the Template Document to the newly created Document. This means that Template Documents can be used for driving property (metadata) content, not just the attached Office template file. This capability is valid only for the "Create" action, and only for property mappings in the direction of Aras to Office, including:

- From Aras to Office File
- From Aras to Word
- From Aras to Excel
- From Aras to PowerPoint
- From Aras to Visio
- From Aras to Excel cell

For example, if you wanted to map the value of **project_code** from the <u>Template</u> Document into the newly created Document and into its Office file, you would define a mapping like this:

The use of the "Use Template" flag also extends to the **Save As to Aras** command. When that command is used, the existing source Document is treated as the "Template" in regard to the "Use Template" mapping flag. Any properties that are mapped from Template Document to new Document during the **New Aras Document** command will also be mapped from "source" Document to new Document during the **Save As to Aras** command.

2.9.4 Mapping Properties from Linked Items

Documents are often linked to other Aras items, like Parts. It is useful to be able to map properties from these linked items. This capability is available but uses a very specific technical approach known as Foreign properties in Aras, which are essentially properties that are inherited from one item to another. Using this approach, it is possible to map and display properties from linked items, but not to set properties on those items. The following example describes the step-by-step process to set this up.

EXAMPLE

Company ACME has certain product specifications in Word that require displaying the official Part and Project numbers from their Aras PLM system, in addition to the Document number itself.

The header of the Word document looks like this:
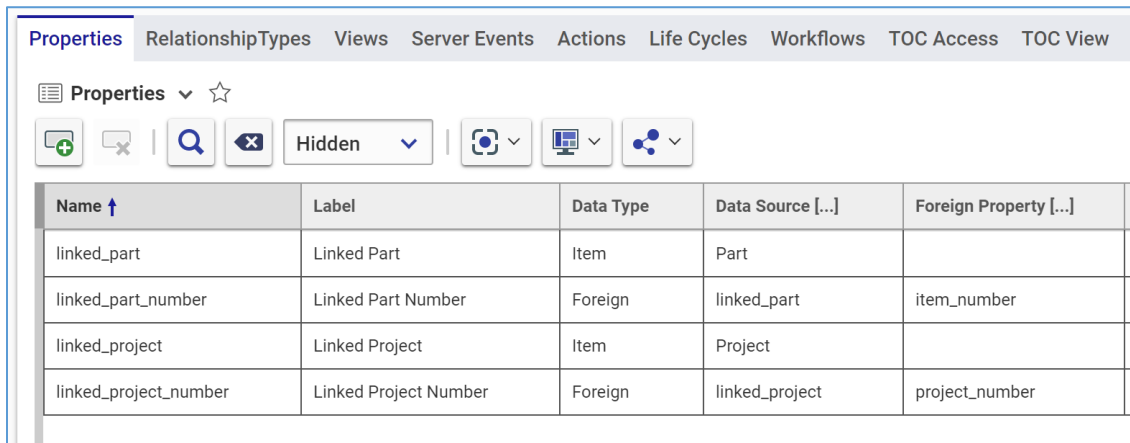
ACME PRODUCT SPECIFICATION

Document Number	DOC-22412
Part Number	P14332-001
Project Number	PROJ-2213

While the value of Document Number is a standard case, getting the values for Part Number and Project Number mapped into the Word document requires establishing appropriate Foreign properties on the Document Item.

STEP 1: Set up Properties on the ItemType

Properties necessary to support our example look like this:

The **linked_part** and **linked_project** properties are regular Item properties that point to the linked Part and Project, respectively. The **linked_part_number** and **linked_project_number** properties are Foreign properties which refer back to the Item properties.

Note: It is possible to access more than one property from the same linked item; you just need to define multiple Foreign properties with the same Data Source but different Foreign Properties.

STEP 2: Set up Foreign Data Source for Linked Items

The Aras Office Connector needs to know the names of which item properties to set for each linked item. When Foreign properties are used, you need to add the name of the Item property into the Foreign Data Source column of the Valid Linked Items settings, as shown here:

Foreign Data Source is an optional setting. Notice that "Activity" and "WBS Element" are Valid Linked Items for this Document Type, but they don't have any Foreign Data Source set. This simply means that no property mappings can be supported from any Activity or WBS Elements that may be linked to a Document of this type.

Note: When a Foreign Data Source is selected, the Aras Office Connector will automatically set the ID of the user's selected linked item into the Foreign Data Source property for the current Document. Then based on the standard behavior of Aras Foreign properties, property values from the linked item are made available.

STEP 3: Set up Property Mappings

To map the actual Foreign properties into Office, just use Property Mappings as usual. Only directions going from Aras to Office are supported.

3 General Administration Topics

This section contains several topics pertinent to configuring Aras Innovator for best use of the Aras Office Connector.

3.1 ItemType Configuration

The Aras Office Connector is designed to work with any ItemType as the base Item, but certain configuration steps must be followed to prepare this ItemType for use by the Connector. All this configuration is already included for the standard Document ItemType within Aras Innovator, so if you wish to configure the Aras Office Connector to work with an ItemType other than Document, it is a good idea to follow the Document ItemType as an example. Configuration of an ItemType for the Aras Office Connector includes the following topics:

Properties Necessary for Aras Office Connector

Name	Label	Data Type	Length	Hidden	Hidden2
authoring_tool	Authoring Tool	List (Authoring Tools)			
external_id	External Id	String	256	X	X
external_owner	External Owner	String	128	X	X
external_type	External Type	String	128	X	X
from_template	From Template	Item (base item)			
is_template	Template	Boolean			
Item_number	*Optional*	String	*Optional*		

Note: There is no requirement that any of these properties be visible on the Form of the Base ItemType, but typically they are useful in showing Authoring Tool, Template, and From Template.

Table 1:   On any Relationship between the Base ItemType and File (example: Document File)

Name	Label	Data Type	Length	Hidden	Hidden2
external_id	External Id	String	256	X	X
external_owner	External Owner	String	128	X	X
external_type	External Type	String	128	X	X

Table 2:   On any Relationship between a Valid Linked ItemType and the Base ItemType (example: Part Document)

Name	Label	Data Type	Length	Hidden	Hidden2
external_id	External Id	String	256	X	X
external_owner	External Owner	String	128	X	X
external_type	External Type	String	128	X	X

Property for Email Message

As described in section 2.3.3, the message body from the Outlook email can be stored in a text property as well as in the attached Message file. It can optionally be displayed on the Document form. The property can have any name you like, since you provide the name within the Email Document Types settings.

Classification

Document Types can make use of ItemType classification if you wish. Classification values defined on the ItemType will be available when defining Document Types. This requires you to configure the Class Structure of the Document ItemType. You can use Document classes already existing at your site or add new ones specifically for use with the Aras Office Connector.

Search Dialog Properties

Search dialogs used by the **Open** and **Links** functions of the Connector are driven directly from the ItemType definition, the same as the search grid in the Aras web client. The display of a given property in these search dialogs is driven by the "Hidden" checkbox in the ItemType definition, and the order is driven by Sort Order.

3.2 Managing Office Files already in Aras Innovator

Office files previously stored manually in Aras Innovator can be easily managed going forward using the Aras Office Connector, with a minimal amount of configuration. The necessary steps are:

1. Make sure all properties listed in section 3.1 are included on the ItemTypes and relationships.
2. The Authoring Tool property of each existing document must be set equal to the corresponding Office application. For example, if the document contains an Excel .xlsx file, then the Authoring Tool must be set to "Microsoft Excel". This is necessary so that the "Open" command can find the valid documents. If you need assistance in setting this value for existing documents in the system, please contact Aras Support.
3. Configure the necessary Document Types in Office Connector Settings that correspond to the ItemTypes and classifications currently used in your system to manage Office files. See section 0 for details.

Note:	The Aras Office Connector only manages one native file and one viewable file per item. Additional files can be attached to the item but will be effectively ignored by the connector.

3.3 Template Document Management

A Template Document in Aras is just a regular Document with the "Template" checkbox checked. It contains an Office file that can be used as the basis of any new file created from the template. The Office file you use for the template should be a regular file format (like .docx, .xlsx, or .pptx). There are two methods for creating Template Documents, either manually in the Aras web client, or by marking a Document created by the Aras Office Connector as a template. Either method requires some work in the Aras web client; it is not possible to create a template completely using the Aras Office Connector.

3.3.1 Creating a Template Document manually

1. Create your template file in the Office application and save it locally.
2. Log into the Aras web client.
3. Create a new Document (or similar item defined as an Office Document Type).
4. Set the following properties:
   a. Number
   b. Type
   c. Authoring Tool
   d. Template checkbox ON
5. Attach your template file on the Files tab.
6. Save and close the Document.

3.3.2 Creating a Template Document using the Aras Office Connector

1. Create a Document using the connector, using either of these two approaches:
   a. Use **New Aras Document** to create a new file, create the file contents, and then **Save to Aras**.
   b. Create the file contents locally, then **Save to Aras**.
2. Log in to the Aras web client.
3. Search for the Document you just created and open the form.
4. Set the Template checkbox ON.
5. Remove any viewable file if it exists.
6. Save and close the Document.

At this point, the template is ready to be used by the Aras Office Connector. Regardless of the method used, you will end up with a Document that has the following key properties set:


These properties are used for the following purposes:

- **Document Number** – Main identifier for the template. Will appear in the Aras ribbon.

- **Type** – Sets which Document Type the template is for. The template will only be listed when that Document Type is selected in the **New Aras Document** dialog.

- **Authoring Tool** – Controls which Office application the template is to be used for. The template will only appear in the **New Aras Document** dialog when you are in that application.

- **Template** – Marks this Document as a template.

- **File** – The file that will be used when the user creates a New Aras Document using this template.
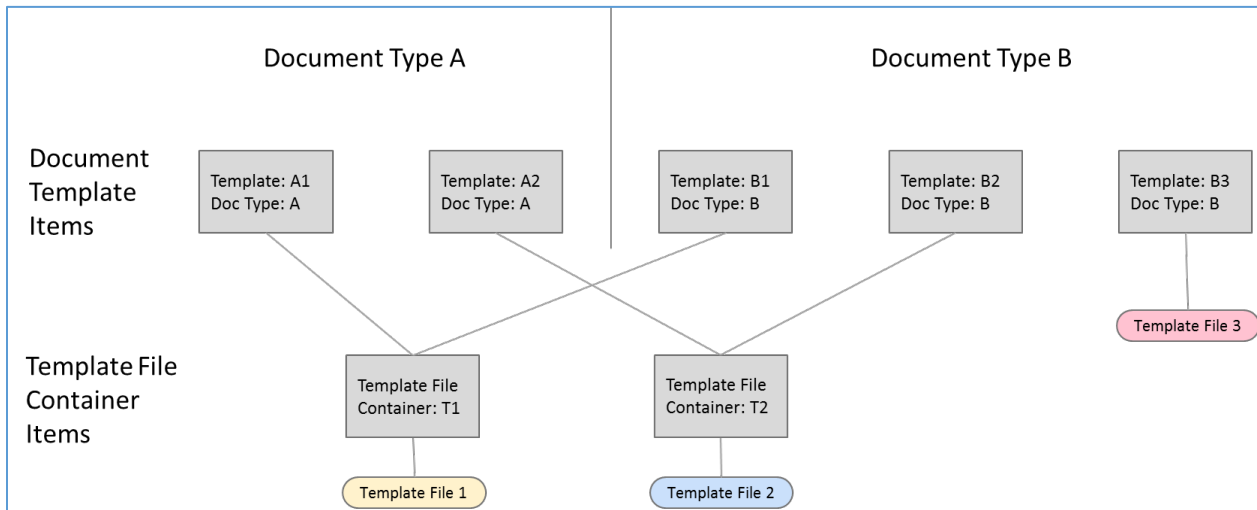
3.3.3 Using Template File Containers

Using Template File Containers is an advanced approach to template management that allows sharing a common template *file* across multiple template *Documents*.

In the standard approach to template management described previously, if the same template file is to be used in multiple Template Documents, it must be copied, like this:
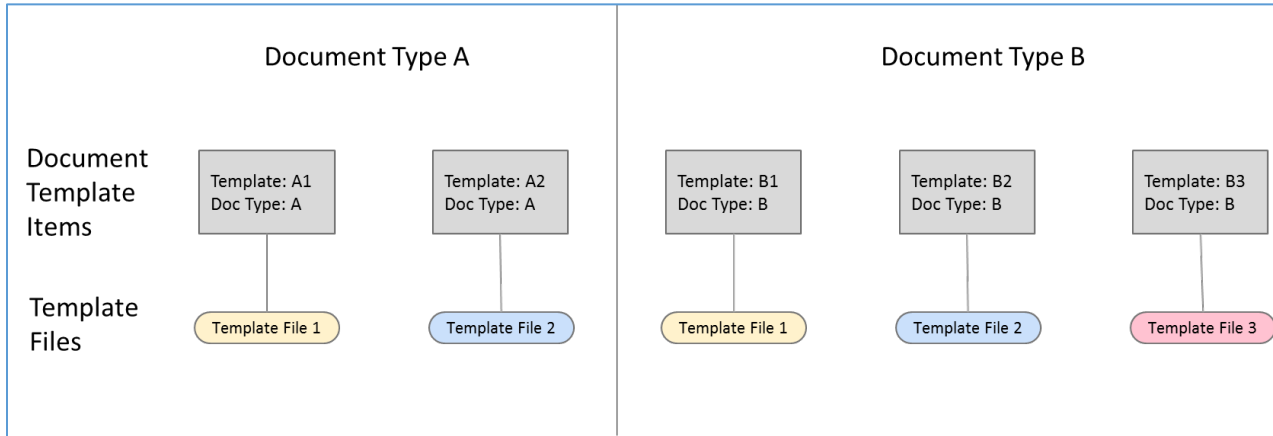
By using Template File Containers, template files can be shared by multiple Template Documents. The existing approach can still be used, as shown with Template B3.

To use Template File Containers, you must create a new Item Property with the name **template_file_container**, and source equal to your "Document" ItemType. Here is an example showing the property defined for the standard Document ItemType:

To create a Template File Container, simply create a Document and attach the template file to it. Then create Template Documents and set the value of "Template File Container" to point to the container item. You can do this as many times as you want.

Note that the Template File Container does not <u>itself</u> need to be a Template Document, as shown in the previous example, although either way is valid. This is controlled by whether the Template flag is set on the Template File Container or not.

It is not necessary for the Template File Container to be of the same ItemType or classification as the template itself, which provides a lot of flexibility for the re-use of template files. However, the container ItemType and classification does need to match <u>some</u> valid Aras Office Connector Document Type.

3.3.4 Administrator topics for Template Documents

The following are a few topics about Template Documents that apply to administrators only:

- The template capability in the Aras Office Connector is designed to be used by anyone in the organization; it is not inherently an administrator-only function. However, as an administrator you can control the ability for users to create templates by restricting their ability to set the "is_template" property ON. This property is how the Connector determines which Documents are templates. You can restrict access to this property in several ways, such as removing it from the form, setting it to disabled on the form, or defining a client-side method to control the ability to set the checkbox on.

- Users can be prevented from modifying Template Documents by using standard Aras's permissions and life cycle states. For example, typically if the Template Document is in the Released state, users will not be able to modify it.

3.4 Office Property Creation and Embedding

Refer to the **Property Mapping and Embedding** section of the *Aras Office Connector 24 - User Guide* for a complete description of dealing with properties inside of Office files. Embedding mapped properties inside Office files, such as Document Number and Revision from Aras, helps to standardize documentation practices within an organization.

3.5 Enabling the "Open in Office" Action

The "Open in Office" action enables users to open files from the Aras web client, under the control of the Aras Office Connector. To set up this action, simply go to the Actions tab on the ItemType used to manage your Documents and add the action "MSO_OpenInOffice". This will enable this action from the main search grid, the main Actions menu, and the item's Actions menu.

Note that if users click on the hyperlinks of Office files in the Aras web client (the standard View action), the system will open the file in the Office application, but not under the control of the Aras Office Connector. Users must use the Open in Office action to properly use the Aras Office Connector.

3.6 Local File Management

Files managed by the Aras Office Connector are kept locally in a working directory which is established at the time that the Connector is installed on each desktop. The default location is **C:\Users\<username>\Documents\OfficeConnector** but can be configured differently.

The Aras Office Connector does not affect the ability to use the normal Office **Save** and **Open** commands. So even when users are managing their files using the Aras Office Connector, they can still save locally and open locally.

3.7 Using Aras Office Connector with Visual Collaboration

The automatic PDF generation capability of the Aras Office Connector is designed to work with Aras Visual Collaboration, enabling all the users in the organization to easily view and markup the document content. For details on configuring Aras Visual Collaboration, please see the *Aras Innovator 24 - Visual Collaboration Administrator Guide*.

3.8 Saving Thumbnail Images

The Aras Office Connector has the optional ability to generate thumbnail images and store them with the Document. To enable the capability, you must create a property with the name **thumbnail** and type Image on the ItemType used to manage your Office files.

If this property exists, a thumbnail image will be stored on each Document, using one of two options. The two options are controlled by the Windows setting "Always show icons, never thumbnails", which is located here in Folder Options:
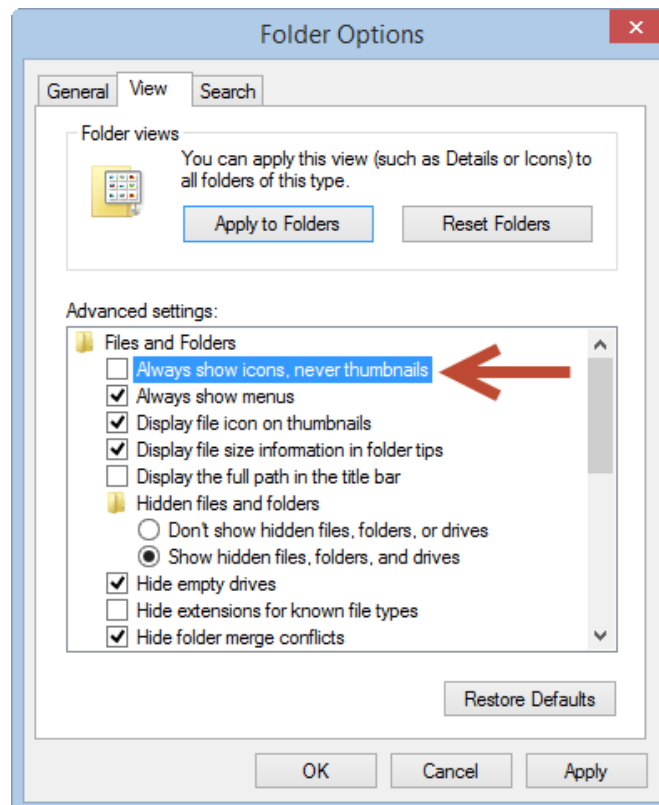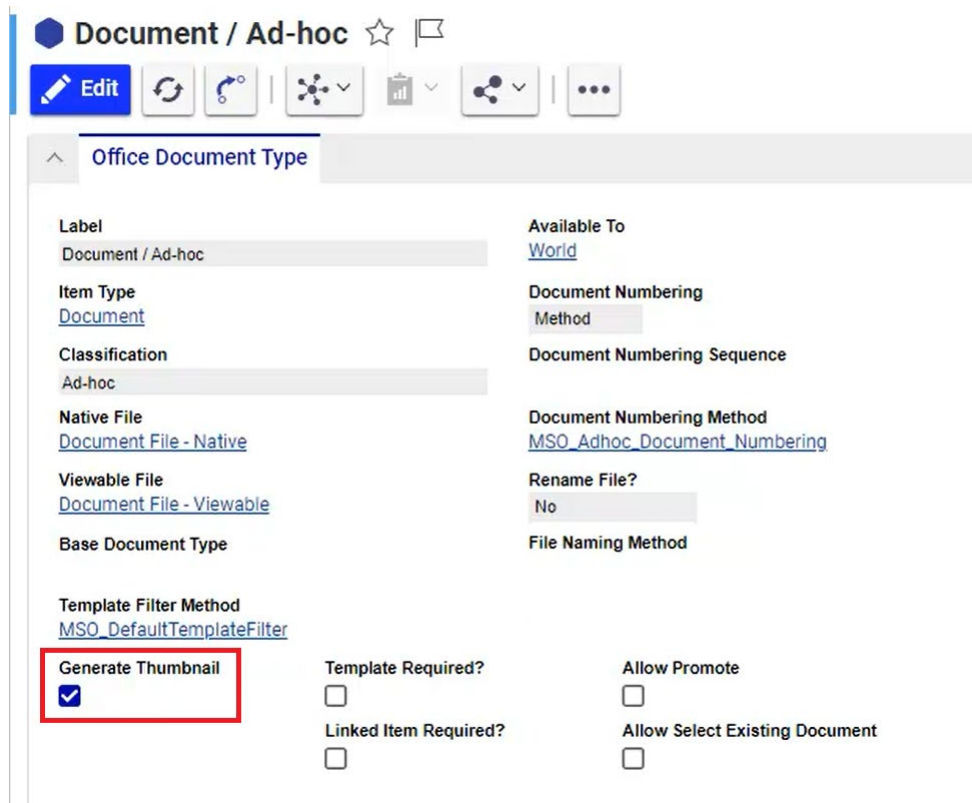
If this setting is OFF, you will get thumbnail images which are pictures of the first page of the corresponding Office file, like this:

If this setting is ON, you will get thumbnail images which are just standard Office icons, like this:

The generation of thumbnail images can still be disabled although the **thumbnail** property exists on the ItemType. Thumbnails are not generated unless **Generate Thumbnail** is enabled for the Office Document Type.

3.9 Changing Label Text in the Aras Ribbon

Aras Office Connector supports the ability to configure all labels and tooltip strings used inside the Aras ribbon, including the name of the ribbon itself. This is useful if you refer to the Aras system by some other name. For example, you could change the ribbon name from "Aras" to "PLM", the "Save to Aras" command to "Save to PLM", etc.

All strings which can be configured are maintained inside an XML resource file, located at **C:\Users\<username>\AppData\Local\Aras Innovator Microsoft Office Connector\Languages**

The file is named **resource.neutral.xml** and will be used as the "localization" source by default if the Languages folder does not contain a source file for the current culture.

The resource file is structured as follows:

```xml
<?xml version="1.0" encoding="utf-8" ?>
<ui_resources>
   <resource_file name="ResourceSetName">
      <resource key="label.Text" value="some text" />
      <resource key="label1.Text" value="some other text" />
   </resource_file>
</ui_resources>
```

The resource keys are named in a straightforward way, corresponding to the strings used inside the ribbon. Simply change the text strings you like, and then restart the application.

4 Troubleshooting

4.1 Error When Running Open In Office

When a user uninstalls one version of MS Office and installs another, an error message might appear when clicking **Open in Office** saying that the application is not supported or that it cannot connect to the Microsoft Office Application.

Contact Aras Support to resolve this error.