

AI Search - Administrator Guide

This guide covers configuring and monitoring AI Search indexing inside Innovator. It assumes AI Search is already enabled. See **Get Started with AI Search** for enablement steps, and **Installation of AI Search** or **Installation of AI Search on SaaS** for deployment-specific installation steps.

Who can use the configurator

The AI Search Configurator is available only to **Innovator administrators**.

On the dedicated AI Search page, administrators see a **gear icon** in the top-right corner that opens the configurator.

Non-administrators do not see this icon and cannot access the admin panel.

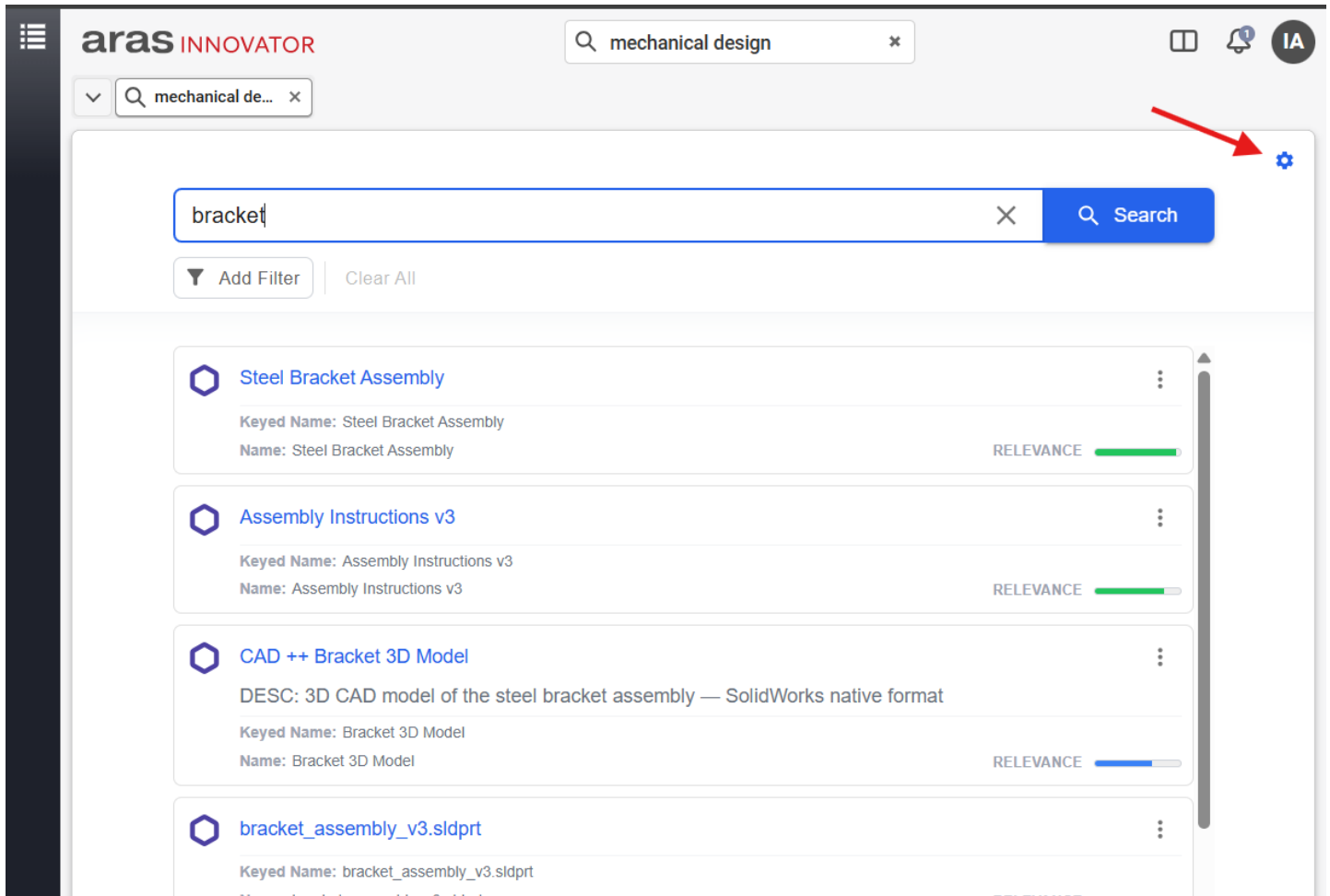
To return to search after configuring, use the **Back to Search** arrow in the Indexing Scope header.

This returns to the AI Search page and preserves the current URL context (such as an active search query).

Configuring Indexing Scope

Open the AI Search page in Innovator and select the gear icon in the top right corner to open the AI Search Configurator.





Selecting ItemTypes to index

The **Select an ItemType...** field is a searchable selector.

Type any part of an ItemType name to filter the list, then select the ItemType and click **Add**.

The list is sourced live from your connected Innovator instance and includes all ItemTypes registered in your database, including PLM data types, administrative types, and system types alike. Use the search field to find the data types relevant to your use case.



aras INNOVATOR

Q CAD x

Indexing Scope

Configure semantic indexing for an Innovator instance.

cui

Add

- cui_Control
 - cui_ControlCommandBarItem
 - cui_ControlEventHandler
 - cui_ControlFormField
 - cui_ControlFormFieldIdentities
 - cui_ControlRule
- cui_Dashboard
- cui_Event
- cui_FormField
- cui_Location
- cui_PresentConfigWinSection

Schema JSON Verify

Indexing Status

Monitor the current status and progress of semantic indexing for this instance.

The list includes both regular ItemTypes and Relationship ItemTypes.

Relationship ItemTypes are marked with a **Rel** badge and can be added in two ways:

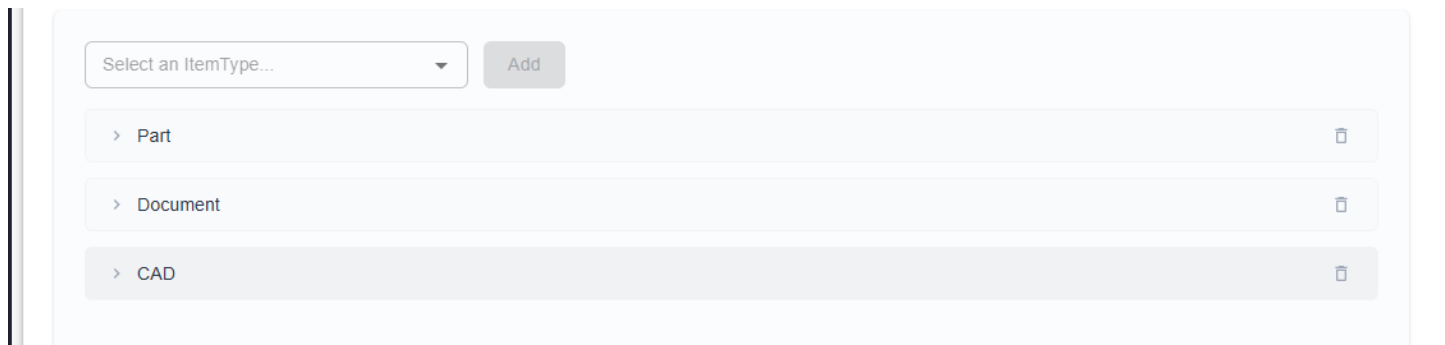
- **Directly to scope:** add the Relationship ItemType itself as an indexable type.
- **As file properties on another ItemType:** configure relationship-sourced file properties under a regular ItemType in the File Properties section (see below).

Each added ItemType appears as a **collapsible row**.

Select the row header to expand it and configure settings.

ItemType detail loads the first time you expand a row; a loading indicator is shown while detail is fetched. If detail fails to load, an inline error is shown with a **Retry** option.

Each row has a **delete** icon to remove the ItemType from scope. If a row fails to render, an error boundary offers to reset by removing that ItemType.



Selecting properties to include

Expand an ItemType row and use the **Select a property...** field to add properties one at a time.

The property selector is also searchable; type any part of a property name to filter the list.

Each added property appears in a table showing its field name and type, with a drag handle for reordering and a delete icon for removal.

Removing a property from the table **disables** it for indexing; it is not permanently deleted and can be re-added later from the property selector.

CAD

CAD Document Settings

Title Template

{{item_number}} - {{description}}

Use {{ fieldName }} for dynamic values.

Subtitle Template

Score: {{score}}

Use {{propertyName}} to reference item properties in templates.

File Handling Settings

Included File Types

Select file types...

File Handling is only available for ItemTypes that contain a property of type "Item".

Properties

Select which properties will be available for vectorization and searching.

FIELD NAME		
item_number		
name		
description		
state		

Important

The property selector shows a consistent set of Innovator properties regardless of which ItemType is selected. Properties available through the selector include system properties common to all items (created_on, css, current_state, generation, is_released, keyed_name, locked_by_id, major_rev, managed_by_id, minor_rev, modified_by_id, modified_on, new_version, not_lockable, owned_by_id, permission_id, state, team_id, and others), plus relationship-level properties for Relationship ItemTypes (quantity, source_id, related_id, sort_order, reference_designator).

Administrators who need to index a custom property that does not appear in the selector can add it by specifying its internal property name directly in the JSON schema and importing it using **Import Schema JSON**. The indexing engine reads properties from the actual Aras Innovator table for each ItemType, so any valid property name on that table can be indexed regardless of whether it appears in the selector UI.

Indexing Scope
Configure semantic indexing for an Innovator instance.

Select an ItemType... Add

- > Part 🗑️
- > Document 🗑️
- > CAD 🗑️

📄 Import Schema JSON 📄 Export Schema JSON 🔍 Verify Cancel Apply

At least one property, file-indexable property, or file relationship is required per ItemType. Attempting to apply a configuration with none of these configured for a given ItemType returns the following error:

```
Apply failed
IndexConfiguration.IndexableItemTypes[N].IndexableProperties:
- At least one indexable property is required
```

Title and subtitle templates

Each ItemType supports an optional title template and subtitle template, which control how that ItemType's results are presented on result cards.

Field	Placeholder syntax	Example	Notes
Title Template	{{fieldName}}	{{item_number}} - {{description}}	Falls back to the item's keyed_name if left empty.



Subtitle Template	{{propertyName}} or {{score}}	Score: {{score}}	Optional. References indexed properties or the relevance score.
-------------------	----------------------------------	------------------	---

Part

Part Settings

Title Template

{{item_number}} - {{description}}

Use {{ fieldName }} for dynamic values.

Subtitle Template

Score: {{score}}

Use {{propertyName}} to reference item properties in templates.

Properties

Select which properties will be available for vectorization and searching.

FIELD NAME

item_number

name

description

File Handling Settings

Included File Types

3D PDF Document (*.pdf)

Select file types...

3D PDF Document (*.pdf)

Actify Model (*.3D)

Adobe Postscript (*.ps)

ASCII Text (*.txt)

AutoCAD Drawing (*.dwg)

AutoCAD Exchange (*.dxf)

Autodesk Model (*.dwf)

AVI Movie (*.avi)

string

Including file content

The **File Handling Settings** area lets an administrator select which file types attached to an ItemType's items should be included in search.

Important

File handling is only available for ItemTypes that contain a property of type Item.

The **Select file types...** field is searchable.
 Type any part of a file format name to filter the list.
 The available file types are sourced live from your Innovator instance's FileType table and reflect whatever formats are registered in your database.

A standard Innovator installation includes common engineering formats such as AutoCAD, CATIA, SolidWorks, STEP/IGES, as well as office formats (PDF, DOCX, XLSX) and text formats. Custom file types registered in your Innovator instance also appear.

Important

The file type list is loaded when the configurator opens and cached for one hour. File types added to Innovator after that point require reopening the configurator to appear in the selector.

Important

If an expected file type is missing from the list, check whether two registered file types in your Innovator instance share the same extension. The configurator shows only the first match when duplicates exist.

File Properties

Below the property table, each ItemType may show a **File Properties** section when file-related properties are available.

This section is separate from **File Handling Settings** (which selects file *types*).

File Properties controls which file-related fields are used for text extraction, vectorization, and search.

Each file property row shows the field name, data type, and an enable/disable switch. Properties sourced from a relationship show a Rel badge with the relationship name.

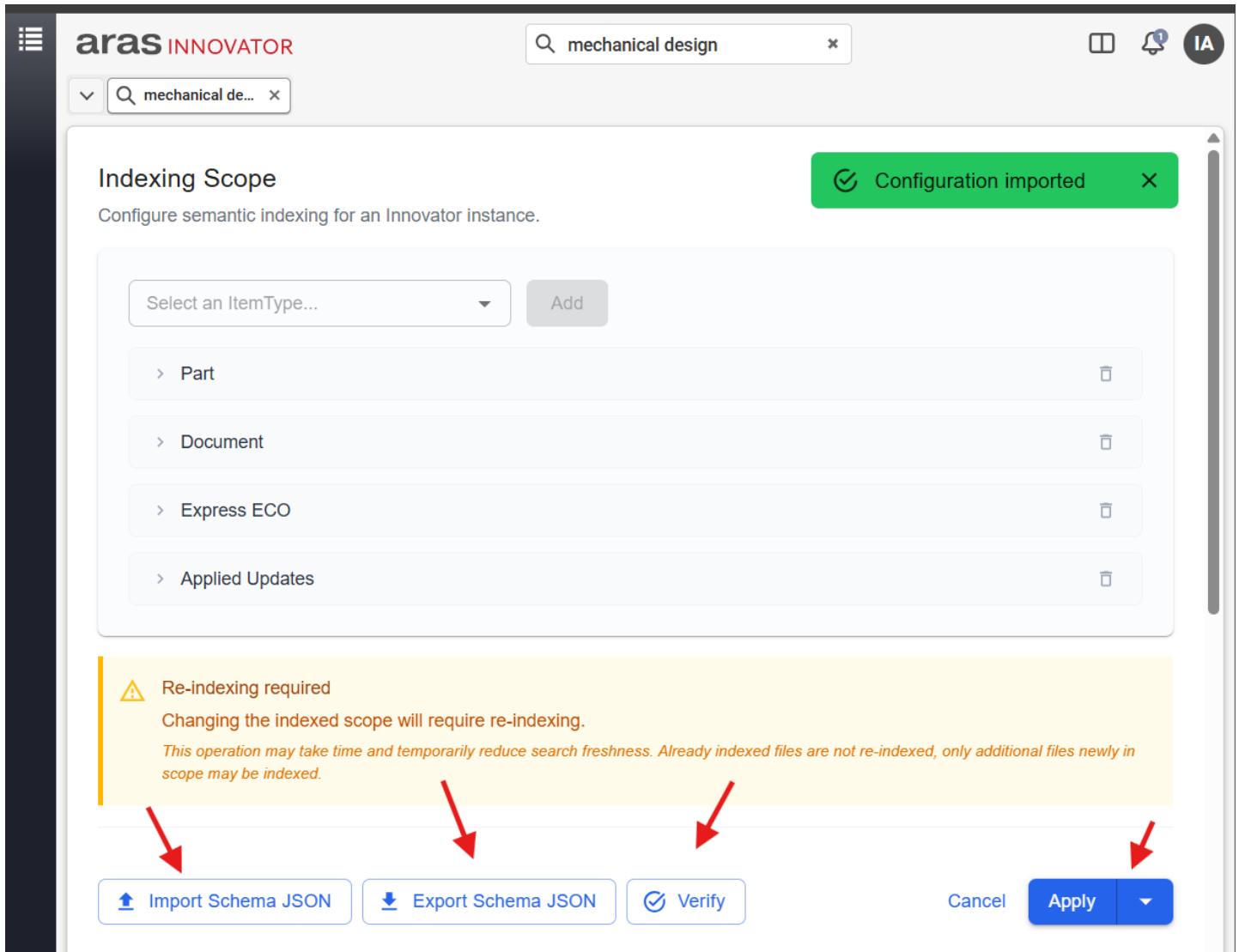
Enable at least one file property when file content indexing is required for an ItemType that has file handling configured.

Schema import, export, and Verify

Administrators can export the current configuration as a JSON schema file, edit it, and re-import it.

The action bar at the bottom of the page shows: **Import Schema JSON**, **Export Schema JSON**, and **Verify** on the left; **Cancel** and **Apply** on the right.





Import does not save automatically

Import Schema JSON loads a JSON file into the configurator editor and shows a success notification (Configuration imported).

The imported configuration is not saved until you select **Apply**.

- **Cancel** reverts all unsaved changes, including an import, back to the last configuration saved on the server.
- **Export Schema JSON** downloads the current editor state (including unsaved changes).
- **Verify** validates the current editor state without saving.



Verify

Verify checks that every ItemType referenced in the configuration exists in the current Innovator instance and that the property constraint described above is satisfied.

Outcome	Notification
Valid	<i>Configuration is valid</i>
Invalid	<i>Validation failed: field-level error details are shown</i>
Network or unexpected error	<i>Verification error or Unexpected response</i>

Verify does not save the configuration. Always select **Apply** after a successful **Verify** to persist changes.

Unsaved changes

The configurator tracks unsaved edits. While changes are pending:

- **Apply** is enabled; **Cancel** is enabled.
- With no unsaved changes, both **Apply** and **Cancel** are disabled.
- While the configurator is loading or a save/verify is in progress, action bar buttons are disabled.

If you navigate away using **Back to Search** with unsaved changes, a confirmation dialog appears:

Unsaved Changes
You have unsaved changes. Are you sure you want to leave? All unsaved changes will be lost.

Select **Leave Page** to discard changes, or **Cancel** to stay. Closing the browser tab with unsaved changes also triggers a warning.

Saving and Applying

Selecting Apply saves the configuration to the server and starts the apply job in the background. The configurator returns to an interactive state promptly; progress is visible in the Indexing Status section while the job runs.



While an apply is in progress, the configurator blocks additional configuration changes. **Apply** and **Cancel** are disabled until the current apply completes.

Outcome	Notification
Apply accepted	Configuration applied
Apply running	Indexing Status shows Running
Apply succeeded	Indexing Status shows Succeeded
Apply failed	Indexing Status shows Failed, with reason
Validation failure at submit	Apply failed — field-level error details are shown
Network or server error at submit	Apply failed — error message is shown

When you have unsaved changes, a re-indexing warning appears above the action bar:

Re-indexing required
Changing the indexed scope will require re-indexing.
This operation may take time and temporarily reduce search freshness.

The warning indicates that the change triggers indexing work. The scope of that work depends on what changed. See [What Apply Does for Each Change Type](#) below.

If the AI Search package is not installed on the Aras Innovator instance, the Indexing Scope section shows:

AI Search Not Installed
This Innovator instance does not have the AI Search package installed.
Please install it on the instance to configure indexing settings.

What Apply Does for Each Change Type

Apply detects the difference between the previously saved configuration and the new one and performs only the work that difference requires.

Change	What Apply does	What Apply does not do
New ItemType added	Starts initial indexing for that ItemType (items and files, per its configuration).	Re-index or disrupt any other ItemType.



Properties changed on an existing ItemType	Re-indexes items for that ItemType only.	Delete or retrigger file embeddings for that ItemType.
File handling changed on an existing ItemType	Restarts the file indexer for that ItemType. Existing file embeddings still within the new scope are preserved and reused without re-parsing.	Delete item embeddings or retrigger item indexing for that ItemType.
ItemType removed	Deletes item embeddings and initial-run history for that ItemType. Stops the file indexer only when no file-indexed ItemType remains active.	Delete file embeddings for that ItemType.
Configuration re-applied without changes	Does nothing. No deletion, no indexing run started.	—

Two invariants apply across all operations, including Apply and Reindex All:

- No operation deletes file embeddings. Files already parsed and embedded are preserved regardless of what changes. Existing file embeddings are reused without re-parsing when they come back into scope.
- ItemTypes not affected by a change remain fully searchable throughout any apply. Search returns only embeddings within the active scope; embeddings that are preserved in storage but outside the current scope are not returned.

scope may be indexed.

Import Schema JSON
Export Schema JSON
Verify
Cancel
Apply

Apply and Reindex All
Save this configuration and rebuild the entire index from scratch.

Indexing Status

Monitor the current status and progress of semantic indexing for this instance.

Last updated: 2026-09-24 Auto- ● sql-search ● sql-data ● innovator ● embeddings ● temporal

Apply and Reindex All

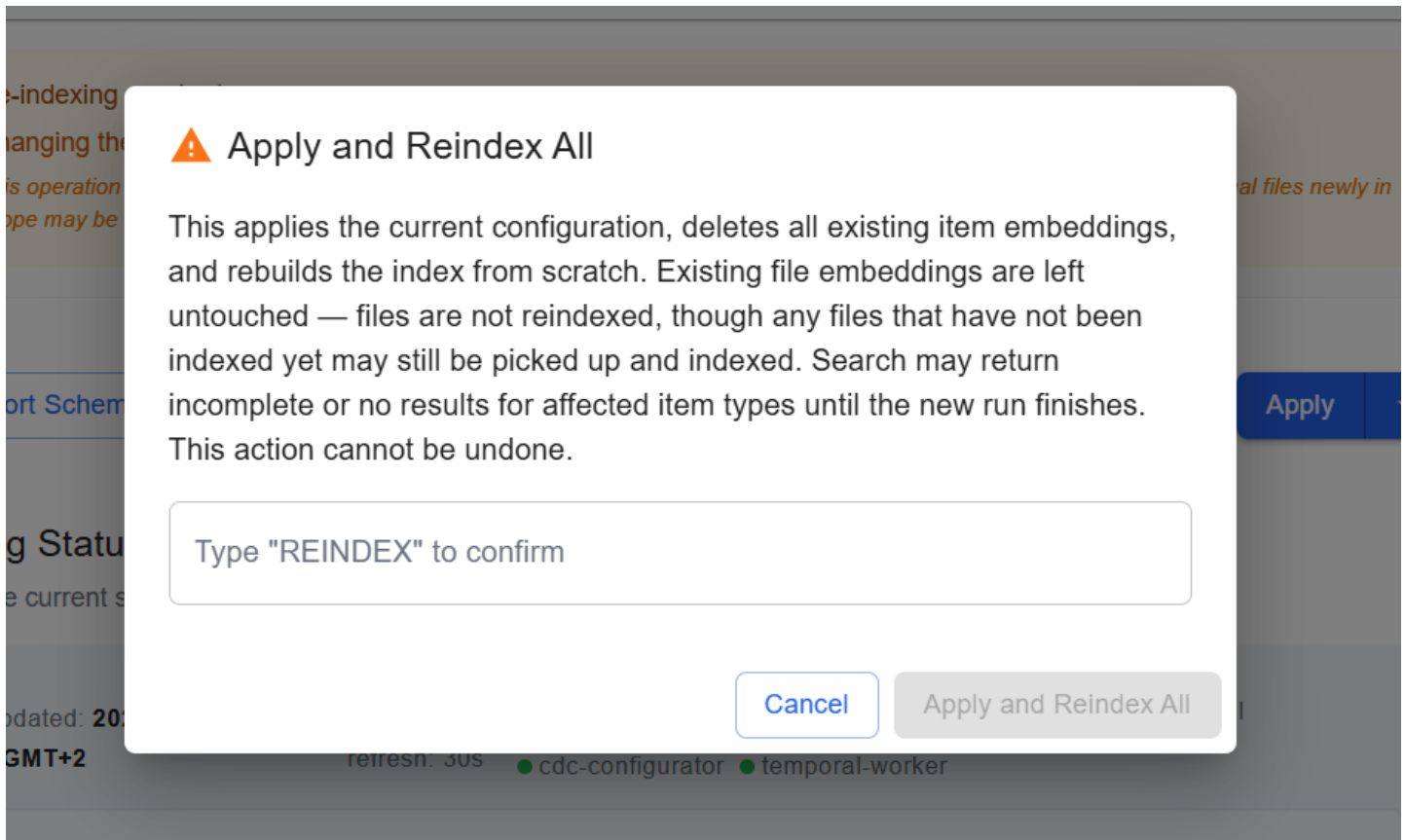
Apply and Reindex All is the secondary action on the Apply split button. Select the dropdown arrow to the right of Apply to access it.

This action saves the current configuration and triggers a full rebuild of all item embeddings. All item embeddings for the active configuration are deleted before rebuilding. Search results for affected ItemTypes are incomplete during the rebuild.



Important

File embeddings are not deleted or reprocessed by Apply and Reindex All. File parsing is resource-intensive and already-indexed file content is preserved and reused. Only item embeddings are cleared and rebuilt.



Use Apply and Reindex All when item embeddings need to be rebuilt from scratch: for example, after an embedding model change or to recover from a corrupted item index state. For routine configuration changes, the standard Apply handles re-indexing automatically and with less disruption.

Important

Apply and Reindex All deletes all item embeddings before rebuilding. Search results for affected ItemTypes are incomplete during the rebuild. On large deployments, this operation may take an extended time to complete. Use it deliberately.

SQL Server Index Requirements for File Properties



On Aras Innovator instances where file properties are enabled for ItemTypes with a large number of records, adding a non-clustered index to the Innovator database for each configured file property column improves search query time.

This applies when both of the following are true:

- One or more ItemTypes in the indexing scope have file properties enabled in the **File Properties** section of the AI Search Configurator.
- The source table for that ItemType in the Innovator database contains a large number of records.

Important

The AI Search service account does not have permission to create indexes on Innovator database tables. These statements must be run by a database administrator with sufficient privileges on the Innovator database.

Creating the Index

For each qualifying ItemType, create a non-clustered index on each configured file property column. Run the following on the SQL Server instance that hosts the Innovator database, substituting the actual ItemType name and file property name:

```
CREATE INDEX [IX_{ItemType}_{filePropertyName}]
ON [innovator].[{ItemType}] ({filePropertyName});
```

Example for a Part ItemType with a `drawing_file` file property:

```
CREATE INDEX [IX_Part_drawing_file]
ON [innovator].[Part] ([drawing_file]);
```

Repeat for each file property configured for that ItemType. If file properties are configured on a Relationship ItemType, create the index on the relationship table instead:

```
CREATE INDEX [IX_{RelationshipName}_{filePropertyName}]
ON [innovator].[{RelationshipName}] ({filePropertyName});
```

Verifying the Index



Run the following after index creation to confirm the index is in place, substituting the actual table name and column name:

```
SELECT
    i.name AS IndexName,
    c.name AS ColumnName,
    i.type_desc AS IndexType
FROM sys.indexes i
INNER JOIN sys.index_columns ic
    ON i.object_id = ic.object_id AND i.index_id = ic.index_id
INNER JOIN sys.columns c
    ON ic.object_id = c.object_id AND ic.column_id = c.column_id
INNER JOIN sys.tables t ON i.object_id = t.object_id
INNER JOIN sys.schemas s ON t.schema_id = s.schema_id
WHERE s.name = N'innovator'
    AND t.name = N'Part' -- substitute ItemType table name
    AND c.name = N'drawing_file'; -- substitute file property column name
```

A row in the result confirms the index is in place. If the query returns no rows, the index does not exist and must be created.

Indexing Status



Indexing Status

Monitor the current status and progress of semantic indexing for this instance.

Last updated: **2026-09-24 17:24**
GMT+2

 Auto-refresh:
30s

● sql-search

● sql-data

● innovator

● embeddings

● temporal

● cdc-configurator

● temporal-worker

Item Type ↑	Total	Indexed	Failed
ITEM INDEXERS			
CAD	5,000	5,000	0
Document	5,000	4,991	3 Errors 
Drawing	0	0	0
ECO	1,200	1,130	2 Errors 
PR	640	0	1 Error 
Part	12,480	12,480	0
FILE INDEXER			
Files	—	8,288	1 Error 

The Indexing Status section, below Indexing Scope on the same page, shows the live state of indexing for the active configuration.

The section auto-refreshes every 30 seconds and displays a **Last updated** timestamp. Select the **refresh control** next to the timestamp to refresh data.

Last updated: **2026-09-24 17:25**
GMT+2

 Auto-refresh:
30s

● sql-search

● sql-data

● innovator

● embeddings

● temporal

● cdc-configurator

● temporal-worker

Item Type ↑	Total	Indexed	Failed
-------------	-------	---------	--------

If a refresh fails but previously loaded data is available, a warning is shown: *Could not refresh indexing status. Showing last known data.*

If the initial load fails entirely, an error is shown with **Retry**.

Indexing Status reflects the **currently saved** configuration on the server. Unsaved changes in the Indexing Scope editor are not reflected here until you select **Apply**.



Service health indicators

A row of status indicators at the top right of the Indexing Status section shows the health of the six backend services that AI Search depends on. Each indicator shows a colored dot and service name:

Dot color	Meaning
Green	Service is healthy
Red	Service is degraded or unhealthy
Indicator	Service
sql-search	SQL search data store
sql-data	SQL data store
innovator	Connected Innovator instance
embeddings	Embedding model service
temporal	Temporal workflow engine
cdc-configurator	Change Data Capture configurator

These indicators are read-only monitoring. They do not trigger repairs or configuration changes. If health data cannot be loaded, **Health unavailable** is shown with a **Retry** link.

Indexing overview table

Below the health indicators, the **Indexing overview table** shows one row per ItemType in the active scope, plus a **Files** row when file handling is enabled for at least one ItemType. Each row shows:

- **Total:** the discovered item count, or an em dash if the total is not yet known.
- **Indexed:** the number of items successfully indexed.
- **Failed:** **0** as plain text when there are no failures, or a clickable chip labeled with the error count (for example, *3 Errors*) when items failed.

Important

The Files row always shows an em dash (—) in the **Total** column instead of a count. Reliably counting the total number of source files is not yet supported end-to-end, so this cell is intentionally left blank rather than showing a misleading number. The Files row's **Indexed** and **Failed** columns are unaffected and populate normally.



Item Type ↑	Total	Indexed	Failed
ITEM INDEXERS			
CAD	5,000	5,000	0
Document	5,000	4,991	3 Errors ▲
Drawing	0	0	0

Select a Failed chip to open the Failed Items dialog for that row. See Handling errors and failed items below.

Showing and hiding the initial indexer details table

Show Initial Indexer details

▼

The initial indexer details table is hidden by default. Select **Show Initial Indexer details** below the Indexing Status panel to expand it. Select **Hide Initial Indexer details** to collapse it.



[Hide Initial Indexer details](#)
^

Last updated: 2026-09-24 17:39 GMT+2
 ↻ Auto-refresh: 30s

Item Type ↑	Status	Started	Completed	Progress
ITEM INDEXERS				
CAD	🔄 Running	2026-05-14 15:15 GMT+2	—	1,234 / 5,000
Document	✅ Completed	2026-05-14 12:10 GMT+2	2026-05-14 12:42 GMT+2	5,000 / 5,000
Drawing	—	—	—	—
ECO	⚠️ Partially Completed	2026-05-14 11:00 GMT+2	2026-05-14 11:35 GMT+2	780 / 1,200
PR	❌ Failed	2026-05-14 11:00 GMT+2	2026-05-14 11:00 GMT+2	Discovery did not complete
Part	⌛ Pending	—	—	Awaiting discovery...
Rows per page: 10 ▾ 1–6 of 6 < >				Total count: 11,200
FILE INDEXER				
Files	🔄 Running	2026-05-14 15:20 GMT+2	—	412 / 8,300

The table is organized into two sections:

- ITEM INDEXERS: one row per ItemType in the active scope. A total item count is shown at the bottom of this section.
- FILE INDEXER: a single row covering all file indexing activity. This row appears when file handling is enabled for at least one ItemType.

Monitoring progress

Each row in the indexer table shows one of five states:

State	Meaning
Pending	Indexing is queued but has not started
Running	Indexing is currently in progress



Completed	Indexing completed successfully			
Partially Completed	Indexing finished but some items failed			
Failed	Indexing did not complete successfully			

Item Type ↑	Status	Started	Completed	Progress
ITEM INDEXERS				
CAD	 Running	2026-05-14 15:15 GMT+2	—	1,234 / 5,000
Document	 Completed	2026-05-14 12:10 GMT+2	2026-05-14 12:42 GMT+2	5,000 / 5,000
ECO	 Partially Completed	2026-05-14 11:00 GMT+2	2026-05-14 11:35 GMT+2	780 / 1,200
PR	 Failed	2026-05-14 11:00 GMT+2	2026-05-14 11:00 GMT+2	Discovery did not complete
Part	 Pending	—	—	Awaiting discovery...
Rows per page: 10 ▾ 1–5 of 5 < >				Total count: 11,200

Each row also shows the **Started** and **Completed** timestamps for the most recent run, a **Progress** counter, and a progress bar.

Progress messages include:

Message	Meaning
<i>Awaiting discovery...</i>	Indexer is pending and the total item count is not yet known
<i>Discovery did not complete</i>	The run failed before item discovery finished
Processed / total count (e.g. 150 / 500)	Items or files processed out of the discovered total

For the **File Indexer**, an info icon may appear when some files reused existing embeddings and did not require reprocessing.

A total item count is shown at the bottom of the table. The table supports pagination; use the rows-per-page selector and navigation controls to view all indexers when many ItemTypes are configured.



The status of the most recently completed run persists between configuration changes and page reloads, so administrators see the last known outcome rather than a blank state.

Empty and not-installed states

If no ItemType has been added to the indexing scope, this section shows:

No Indexers Configured
No item types have been configured for indexing yet.

If the AI Search package is not installed, this section shows:

AI Search Not Installed
Install the AI Search package to view indexing status.

Handling errors and failed items

A **Failed** or **Partially Completed** status on an indexer row indicates that the corresponding indexing run did not complete without errors.

Document — Failed Items

3 Errors

	Name	Reason		
☒	8073D8E98C764EC7873103B49B9CEF9E	HTTP 429 Too Many Requests: embedding service rate limit exceeded		
☒	2B4C8F1A9D764EC7873103B49B9CEF01	Failed to parse item: required property "description" is empty		
☐	STALE_INFLIGHT_PROCESSING_ROW_001	Stale processing claim - ready for retry		

Retry 2 selected

Resolve 2 selected

Close

Select the Failed chip in the Indexing overview table (see above) to open the Failed Items dialog for that row. The dialog title reads *{ItemType or Files} — Failed Items*, and it lists each failed item or file

with its **Name** and failure **Reason**.

For a single row, select **Retry** to re-queue that item or file for indexing immediately, or **Resolve** to dismiss it from the failed items list without indexing it. Select multiple rows with their checkboxes to enable bulk actions in the dialog's action bar: **Retry N selected** and **Resolve N selected**.

Resolve requires confirmation. The **Resolve Failed Items** dialog reads: "*N item(s) will be dismissed from the failed items list without being indexed. This does not delete the underlying item -- it stops the platform from retrying it automatically. You can still retry it later if it reappears.*" For files, an additional sentence clarifies that resolving dismisses only file-indexing reconciliation, and that a separate File ItemType reconciliation failure, if present, must be resolved from the item-type view. Select **Resolve** to confirm, or **Close** to cancel. **Retry** has no confirmation step and queues immediately.

Select **Close** to dismiss the Failed Items dialog. It also closes automatically once its failed items list becomes empty.

All ItemType versions are indexed.

Troubleshooting and Common Issues

Symptom	Likely Cause	Resolution
Apply fails with "At least one indexable property is required"	No property, file-indexable property, or file relationship has been added to the ItemType	Expand the ItemType row and add at least one property, enable file handling with file types, and/or enable file properties
An expected property does not appear in the property selector	The property is specific to that ItemType and is not surfaced by the selector UI	Add the property by its internal name using Import Schema JSON ; the indexing engine reads it from the actual Aras Innovator table
An expected file type does not appear in the file type selector	The file type is not supported by the AI Search file parser; or two file types share the same extension (only the first is shown); or the file type was added to Aras Innovator within the one-hour cache window	Confirm the file type is in the supported format list (PDF, DOCX, PPTX, HTML, Markdown, PNG, JPG, GIF, TIFF, BMP, WEBP). If it is supported and still missing, reopen the configurator to refresh the cache, or check Aras Innovator FileType configuration for duplicate extensions.
Configuration imported appears but search behavior is unchanged	Import loads the editor only; changes are not saved until Apply	Select Apply after import, or Cancel to discard the import
Unsaved changes were lost after leaving the page	Back to Search or browser navigation discarded pending edits	Save with Apply before leaving; respond to the unsaved-changes dialog if prompted
The Index Is Not Ready Yet or Is Currently Unavailable message appears on search	The index is rebuilding, temporarily locked, or has not completed initial indexing	Wait for indexing to complete, or check Indexing Status for the current state of each indexer
Indexing Status shows "No Indexers Configured"	No ItemType has been added to the indexing scope yet	Add at least one ItemType in the Indexing Scope section and select Apply
Indexing Status shows Could not refresh indexing status	A background refresh failed but cached data is still shown	Select refresh to retry; check service health indicators



Schema import succeeds but Apply still fails	The imported schema references an ItemType or property that does not exist in this Aras Innovator instance	Use Verify before Apply to catch validation errors early
Indexer row shows Failed or Partially Completed	One or more items in the run encountered errors	Select the Failed chip in the Indexing overview table to open the Failed Items dialog and retry or resolve the affected items
A Relationship ItemType does not appear as expected in search results	Properties or file properties may not be configured correctly	Confirm the relationship ItemType has indexable properties enabled, or that relationship file properties are enabled under the parent ItemType
Service health indicators show red	A backend dependency is degraded or unhealthy	Wait and retry; contact your administrator or Aras support if the condition persists

If issues persist after following the steps above, contact Aras Technical Support via the Subscriber Portal with your tenant ID, the ItemType configuration in question, and any error details copied from the Failed Items dialog.

