

AI Search Release Notes

This page includes all AI Search Release Notes starting with the most recent.

2026 October 6 - AI Search - Minor Release - AI Search version 1.1.1

AI Search version 1.1.1 is now live!

6 October 2026 - Minor Release

Version 1.1.1 is a minor release that resolves an issue introduced in AI Search version 1.1.0. No configuration changes and no infrastructure changes are required.

Bug Fixes

Search Returned No Results on Instances with Large File Volumes [IH-25058]

Resolved an issue introduced in AI Search version 1.1.0 that caused AI Search to return no results on Aras Innovator instances with a large number of document-file relationships and file-related properties in the indexing scope. On affected instances, search queries exceeded the response time limit and returned no results.

Upgrading to AI Search version 1.1.1 resolves this issue. Search queries now complete within the response time limit on affected instances.



Known Behaviors and Limitations

Search Latency on Instances with Large File Property Tables

On Aras Innovator instances where file properties are configured for ItemTypes with a large number of records, search queries that retrieve file results may experience elevated latency. Adding a non-clustered index to the Innovator database for each configured file property column resolves this. See *SQL Server Index Requirements for File Properties* in the Administrator Guide for instructions.

2026 September 30 - AI Search - Incremental Release - AI Search version 1.1.0

AI Search version 1.1.0 is now live!

30 September 2026 - Incremental Release

Introducing Smart Configuration Changes Detection, indexing health visibility with in-place failure recovery, expanded search filters, document content visible directly in search results, a result detail drawer, improved search reliability on large deployments, and a new API for retrieving file chunk content.

What's new

Smart Configuration Changes Detection

Applying an index configuration previously triggered a full rebuild of every indexed ItemType in the active scope, regardless of what actually changed. Version 1.1.0 makes apply incremental: the system detects which ItemTypes and content scopes are affected by the change and re-indexes only those. ItemTypes that were not changed remain fully searchable throughout the apply.



Apply is now asynchronous. Selecting Apply returns the configurator to an interactive state promptly; the apply job continues in the background. Progress is visible in the Indexing Status section, which reports Running, Succeeded, or Failed. The configurator prevents additional configuration changes while an apply is in progress.

A full rebuild of item embeddings remains available as an explicit action. File embeddings are not cleared by any operation, including a full rebuild, because file parsing is resource-intensive.

Why it matters: Administrators can now iterate on the indexing scope without accepting the cost of a full rebuild each time. Adding an ItemType, adjusting properties, or narrowing file handling affects only what changed. Users searching other ItemTypes see no disruption.

Example scenario: An administrator adds a new ItemType to the indexing scope. Only that ItemType is indexed. Everything already in scope stays searchable and unchanged.

Monitor indexing health and resolve failed items from the Admin panel

The Indexing Status section of the AI Search Configurator now includes an **Indexing overview** table showing, per ItemType and for the File Indexer, how many items are in the index (Total, Indexed, and Failed). Rows with failures display a clickable **Failed** chip showing the error count.

Selecting the chip opens the **Failed Items** dialog for that row. The dialog lists each failed item or file by name and the reason the system reported. The administrator can retry a failed item to re-queue it for indexing immediately, or resolve it to dismiss it without indexing. Both actions are available per item and in bulk across a selection. Resolve requires confirmation.

Why it matters: A Failed or Partially Completed indexer status previously gave an administrator no information about which items failed or why, and no way to act without contacting support. Failures can now be investigated and cleared directly from the Admin panel, without a support request.

Example scenario: An indexer row shows 3 failures after an apply. The administrator opens the Failed Items dialog, corrects the underlying data issue for two items and retries them, then resolves



the third as a known bad record. The row returns to a healthy state.

Filter search results by state, date, and release status

The search filter panel now includes four filter properties in addition to Item Type: State, Modified After, Modified Before, and Is Released. Item Type filtering now supports selecting more than one type in a single search.

The State filter uses a text-contains match. Typing part of a state name is enough; exact text is not required and wildcards are not needed.

Why it matters: Narrowing search results by lifecycle state and date range are routine steps in review and change management work. These filters are now available directly in search results, without a separate navigation step.

Example scenario: A reviewer needs all documents in a Released state that were modified in the past 30 days. They set a State filter with "Released" and a Modified After date.

See why an item matched directly in the results

Expanding a result card now shows the text excerpts from indexed files that made the item relevant, grouped by source and annotated with the file name. Users no longer have to open the item to find out what matched.

When a search matches content at the file chunk level, the relevant chunk appears as its own result entry, with a clickable reference back to the file it belongs to.

Why it matters: Knowing why an item appeared in a result previously required opening the item and locating the relevant section. The matched content is now visible from the results list.

Example scenario: A user searches for a specific requirement phrase. The result card expands to show the matching excerpt from the attached specification file, with the file name shown, without



opening the document.

Inspect item details without leaving the results list

Selecting the detail control on a result card opens a side drawer with the item header, matched excerpts, and file references. The results list stays visible and in position behind the drawer. Full keyboard navigation is supported throughout.

Why it matters: Reviewing multiple results in sequence previously required opening and returning to the search view for each one. The drawer keeps the results list intact.

Example scenario: An engineer reviews several candidate parts from a single search by opening the drawer on each result in turn, without leaving the results view.

Search performance on larger indexing volumes

On instances with large volumes of indexed content, unfiltered search queries now return results within the expected response time. Earlier releases performed an exhaustive scan of all indexed embeddings on every query, producing extended delays or timeouts at scale. Version 1.1.0 routes queries through an index-aware strategy on SaaS deployments and a streaming strategy on on-premises deployments, both of which avoid the full-scan behavior for unfiltered queries.

Queries that run for longer than 30 seconds now return a message prompting the user to apply filters or narrow the search terms rather than displaying an indefinite loading state.

Filtering by item type on instances with large indexing volumes can produce extended search times in this release. Users whose Innovator permissions restrict access to a small portion of the indexed content may also experience longer response times on large deployments. Both are known behaviors in version 1.1.0.

Why it matters: Production deployments with large indexing scope can now run unfiltered searches without encountering timeouts, and users receive guidance when a query needs to be narrowed



rather than waiting indefinitely.

Retrieve file chunk content through the AI Search API

The AI Search API now exposes the indexed file chunks that drive search results. Chunks can be retrieved inline with a standard search response or through a dedicated endpoint for a specific item. Both are access-controlled: only chunks for items the caller is authorized to access are returned.

Why it matters: Developers and LLM agents consuming AI Search results now have access to the underlying file content behind each result, enabling grounded answer generation and retrieval workflows that need source text rather than item references.

Example scenario: An LLM agent sends a search query and receives a ranked list of items, each including the highest-ranked file excerpts. The agent uses those excerpts as context when generating its response.

Bug fixes

Filters could not be added after the filter panel was opened [IH-21708]

Resolved an issue where the Add Filter control was not functional after the filter panel was opened. Filters could not be applied without reloading the page.

Date filter returned a service error when date range was invalid [IH-23601]

Setting Modified Before to a date earlier than Modified After now returns a validation message in the filter panel rather than a service error.

File type selector showed types the parser does not support [IH-19039]



The file type selector in the AI Search Configurator now shows only file types that the file parser supports. Previously, administrators could select formats that would produce no indexed content.

Removed files continued to appear as indexed [IH-22290]

File indexing status no longer reports files as part of the active index after they have been removed from the indexing scope.

Apply button remained in a loading state after configuration was saved [IH-19640]

The Apply button in the AI Search Configurator no longer stays in a loading state after a configuration is successfully applied.

Previous configuration appeared after navigating back to the Admin panel [IH-21755]

Navigating from the search view back to the Admin panel now shows the current saved configuration rather than a previously applied one.

Scroll was not accessible when the filter panel was open [IH-20447]

The scroll control is now available on result cards when the filter panel is expanded.

ItemType with no items to index remained in a Pending state indefinitely [IH-17843]

An ItemType with no items or files in scope now reaches a Completed state with a count of zero rather than staying in a Pending state.

2026 July 14 - AI Search - Release Announcement - AI Search version 1.0.0



Aras InnovatorEdge AI Search version 1.0.0 is now live!

June 14, 2026 - Foundational Release

Aras is pleased to announce the general availability of **AI Search**, the first semantic search capability built into **Aras Innovator**.

AI Search enables natural language search across Innovator items and the files attached to them, governed by the same permissions, policies, and data model your organization already owns and controls.

What Is Included in This Release

Natural Language Search in Innovator

Any Innovator user can now search across the configured item and file scope using plain language, with no knowledge of ItemTypes, classification paths, or filter syntax required. Results are ranked by relevance, filterable by item type and version status, and each result links directly back to the source record in Innovator.

Administrator-Controlled Indexing

Administrators define what is searchable entirely within Innovator. No engineering involvement is required to configure the indexing scope, select properties for vectorization, enable file content handling, or customize how results are presented. The configuration can be exported, imported, and validated before it is applied, and administrators can monitor indexing progress and system health in real time from within the same interface.

Semantic Search API

AI Search results are available to external developers and integrations through the Semantic Search API endpoint, including through Edge API Query-on-the-fly. This enables AI Search to serve as a retrieval layer for agents, external applications, and custom workflows that operate alongside



Innovator.

On-Premises Deployment

AI Search is the first Edge AI platform component to support full on-premises deployment using Docker containers, in addition to SaaS. Organizations with strict data residency or network isolation requirements can self-host the complete AI Search stack alongside their Innovator instance.

Why This Matters

Innovator contains the most complete, authoritative record of a product's lifecycle. The barrier to accessing that knowledge has not been its absence; it has been that retrieving it required knowing exact item numbers, classification paths, or the specific structure of the data model.

Engineers who know what they are looking for in plain language have had to learn PLM vocabulary, run manual searches, or depend on someone who has already navigated the structure. AI Search removes that barrier. The search experience is the same for a first-day user and a ten-year PLM veteran.

Every query executes under the user's own Innovator session. Every result is drawn from data the user is already authorized to see. No customer data is used to train or improve AI models.

Deployment

AI Search is available for Full SaaS and Full On-Premises deployments. This release requires an active Aras InnovatorEdge AI Subscription (AS-1242) and Aras Innovator 14.0.15 or later. On-premises deployment additionally requires SQL Server 2025 or later on the same SQL Server instance that hosts Innovator; Azure SQL Database with vector support is also supported.

Known Behaviors and Limitations

- The property selector in the AI Search Configurator shows Innovator system properties in this release. ItemType-specific custom properties such as item_number and description can be



indexed by adding them directly through JSON schema import; the indexing engine reads from the actual Innovator table and supports any valid property name.

- Queries must be at least 3 characters long.
-

