



Aras Innovator*Edge AI*

Last Modified: 09/25/2026

Copyright Information

Copyright © 2026 Aras Corporation. All Rights Reserved.

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

Notice of Rights

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

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

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

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

Notice of Liability

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

Table of contents

Edge AI - Business Features and Benefits

Administrator Guide

User Guide

Appendix: Glossary

Frequently Asked Questions

Edge AI Release Notes

Task Agents

Aras InnovatorEdge AI Agents Overview

Requirements Ingestion Agent - User Guide

BOM Import Agent - User Guide

Smart Variants Agent - User Guide

Impact Analysis AI Agent - User Guide

Aras InnovatorEdge AI Agents - Release Notes

AI Search

AI Search - Business Features and Benefits

Get Started with AI Search

Installation of AI Search

Installation of AI Search on SaaS

AI Search Configurator - Admin Guide

AI Search - User Guide

Appendix: Basic Terminology

AI Search: Frequently Asked Questions

AI Search Release Notes



Edge AI - Business Features and Benefits

InnovatorEdge AI Platform Overview

InnovatorEdge AI is a governed service for AI-driven experiences connected to enterprise and PLM data. It allows organizations to instantiate AI agents that use natural language, work with authorized data through the Edge API, and support multiple interaction modes tailored to engineering workflows.

Agents and Templates

InnovatorEdge AI hosts multiple governed agents within a tenant. Every tenant receives the Aras AI Assistant, a production-ready agent that Aras builds, maintains, and improves with each release. Additional agents are created from templates covering direct question answering, the assistant baseline, and multi-agent orchestration. Each agent carries its own instructions, tools, skills, and data connections.



Administrator Guide

Introduction

InnovatorEdge AI is the agent platform for Aras Innovator. It hosts governed AI agents that reason over PLM data, call tools to retrieve and change it, delegate work to specialist agents, and reach systems beyond Innovator through MCP.

Every agent reaches your data through Edge API. You decide which item types, properties and actions are exposed, and every access is logged. For the agents your people talk to, requests run under the identity of the signed-in user, so each agent is bounded by the permissions of the person using it.

Conversation is the primary surface. A user asks a question in plain language and receives an answer drawn from live Innovator records, each with a link back to the record it came from.

Agents grow with your needs. Add skills to teach an agent domain-specific patterns, connect it to systems beyond Innovator through MCP, and compose agents so that one delegates specialist work to another.

Agents act on data as well as read it. They can create records, update properties, change lifecycle state, and delete items, and every operation that changes data requires explicit human approval before it is committed.

InnovatorEdge AI hosts as many agents as you need. Each one carries its own instructions, tools, skills, and data connections. The Aras AI Assistant ships as the default governed agent in every tenant, and you can create additional agents for specific tasks or domains.

Your work as an administrator happens in the Agent Management Platform, where you configure agents and the resources they draw on: Access Points to Innovator data, model configurations, skills, and the tool registry. Who can do what is controlled through Aras Centralized Identity and Access Management (CIAM).



Who This Guide is For

This guide is for the **Edge AI Administrator**, the person responsible for setting up, configuring, and operating InnovatorEdge AI for their organization: agents, models, Access Points, skills, MCP Servers, and user access within the tenant.

It assumes familiarity with Aras Innovator administration and with your organization's Edge API configuration.

What This Guide Does not Cover

Topic	Where to find it
Using the assistant as an end user	User's Guide
Configuring Edge API itself, including exposing item types, properties, and actions	Building your API schema
Connecting an Aras Innovator instance to Edge API	Connect your Aras Instances
Creating and managing Edge API keys and access tokens	Managing API keys and Access tokens
CIAM administration beyond assigning Edge AI roles	CIAM Documentation
Aras Innovator administration	Aras Innovator release-specific documentation

Related Documentation

- [Frequently Asked Questions](#) for common questions on setup, security, capability, and behavior.
- [Appendix: Glossary](#) for terms specific to InnovatorEdge AI.
- [Edge AI Release Notes](#) for what changed in the current release.

Getting Started

This chapter takes your tenant from newly provisioned to a working assistant that your users can reach. Work through the steps in order. Several of them happen in services outside InnovatorEdge AI, and each points to the guide that covers it in detail.

What You Need

Before you begin, make sure the following are in place:



- An Aras Innovator instance, and administrator access to it.
- An active Edge account for your organization, with both an Edge API entitlement and an Edge AI entitlement. Both are required. See [Get Started With Aras InnovatorEdge API](#) .
- At least one Innovator instance connected to Edge API with a valid trusted domain certificate.
- Tenant administrator access in Aras CIAM, so that you can assign Edge AI roles.

Important

The trusted domain certificate is what allows Edge API to identify the requester. Without it, an Access Point using End User authentication cannot verify who is asking, and identity-aware behavior is unavailable.

Setup Checklist

1. **Enable Edge API for your organization.** InnovatorEdge AI reaches your Innovator data through Edge API, so this establishes the layer everything else runs on. The designated tenant owner requests the Edge account; other people in the organization request user access from their tenant owner. See [Get Started With Aras InnovatorEdge API](#) .
2. **Confirm the CIAM tenant notification.** Aras shares the InnovatorEdge AI application with your tenant. Once shared, it appears in your CIAM Shared Applications list and the tenant administrator is notified automatically. Confirm the notification has arrived before continuing.
3. **Assign roles in CIAM.** Open CIAM user management and assign Edge AI User to the people who will use the assistant, and Edge AI Administrator to the people who will manage it. Users receive an email invitation when they are first granted access. If your organization uses Innovator user synchronization, configure it here. See **Roles and Access** on this page, and [CIAM user documentation / User Management](#) .

Important

A user with no assigned Edge AI role cannot reach the assistant or the Agent Management Platform. This step decides who has access, so complete it before you announce the service internally. We recommend managing access through dedicated groups rather than individual assignments; see CIAM Documentation for guidance on group and user assignment.



4. **Install the Aras Innovator AI Assistant.** The embedded assistant panel is delivered as a platform component application installed into your Aras Innovator instance. Download the package from your Edge account and install it following the Aras Innovator AI Assistant SaaS Installation Guide, which is provided in the Innovator AI Assistant folder inside the package. Once installed, the assistant is available to users as an embedded panel within Innovator.
5. **Connect your Innovator instance in Edge API.** Register your Aras Innovator instance as a data source, then expose the item types, properties and actions your agents will need. What you expose here sets the outer boundary of what any agent can reach, and it decides whether agents can change data at all: an agent can write only where the API exposes an Add, Edit or Delete action. See [Connect your Aras Instances](#) and [Building your API schema](#).
6. **Add an Access Point in the Agent Management Platform.** Open the Agent Management Platform and register the Edge API connection as an Access Point: the endpoint URL and the authentication mode. With End User authentication, the signed-in person's own identity is used for every request, so there is no credential to store. If you choose API Key authentication instead, create the key in Edge API first and enter it here. See **Access Points** on this page.
7. **Configure and publish the Aras AI Assistant.** Open the Aras AI Assistant in the Agent Library. Select the Access Point you created, choose the tools you want the assistant to use, and save. Test it in Simulation Chat if you want to see how it behaves first. Then publish it. See **Agents** on this page.

Important

Publishing is always explicit. Registering an Access Point makes an agent publishable, because the agent can then be configured against at least one API. It does not select tools for you and it does not publish anything. An agent stays in Draft, reachable only by you in Simulation Chat, until you publish it.

8. **Verify it as a user would.** Open Aras Innovator, open the assistant panel, and ask one real question about data you know exists. Confirm three things: an answer comes back, it is scoped to what that account should see, and the link on a result opens the record in Innovator. If you have both an administrator and an ordinary user account available, try the same question with each and compare what comes back.

What Happens Next



Your Edge AI Users can now open the assistant in the embedded panel inside Innovator, or select it from the agent list at `edge.aras.cloud/ai`.

As you extend what Edge API exposes, agents pick up the change on their next request.

There is nothing to refresh in InnovatorEdge AI.

From here, this guide covers configuring the assistant further and creating agents of your own.

Navigating the Agent Management Platform

InnovatorEdge AI has a single entry point, at `edge.aras.cloud/ai`.

Everyone with an Edge AI role arrives at a home page listing the agents published for them.

As an administrator you also have a route onward from there into the Agent Management Platform, which is where you configure everything and which is reachable only by Edge AI Administrators.

Keeping the platform in one browser tab and the assistant in another is the practical way to configure and test at the same time.

The platform is organized into two groups.

- **Agents.** The Agent Library, where you create, configure, publish, and manage agents.
- **Resources.** Access Points, Models, Skills, and MCP Servers. These are the building blocks an agent draws on, and each is shared across every agent in your tenant.

Aras provides some of these resources ready to use. Anything Aras manages is marked as such and presented read-only, so you can always tell what you inherit and what your organization owns. See **Skills**, and **Models** on this page.

Roles and Access

Access to InnovatorEdge AI is controlled through Aras CIAM, where Edge AI is registered as a single application. You assign each person one of two roles, and that role decides what they can reach and what they can do there.

Edge AI User

Converses with published agents, through the embedded panel in Innovator or the standalone chat. Cannot open the Agent Management Platform and cannot see or change any configuration.

Edge AI Administrator



Administers InnovatorEdge AI for one tenant. Creates, configures, publishes, archives and deletes agents, and manages Access Points, Models, Skills and MCP Servers. Can test draft agents in Simulation Chat, and can also converse with published agents.

Permissions Reference

Action	Edge AI User	Edge AI Administrator
Converse with published agents	Yes	Yes
Test draft agents in Simulation Chat	No	Yes
Open the Agent Management Platform	No	Yes
Create and configure agents	No	Yes
Publish and unpublish agents	No	Yes
Archive, restore and delete agents	No	Yes
Manage Access Points	No	Yes
Manage Models	No	Yes
Manage Skills	No	Yes
Manage MCP Servers	No	Yes

A role carries the same rights across every agent and resource in the tenant, so an Edge AI Administrator administers the whole tenant rather than a subset of it. Where a person holds more than one role, their access is merged.

What a Role Does Not Change

An Edge AI role decides what a person can configure. It has no effect on what data they can reach. Data access is decided by the Access Point an agent uses. Where that Access Point uses End User authentication, every query and every write runs under the signed-in person's own Innovator identity, so Innovator permissions decide the data and two people asking the same agent the same question receive answers scoped to their own access. See **Access Points** on this page.

Important

Granting Edge AI Administrator gives someone full control of agent configuration. It grants no additional access to Innovator data. If an administrator needs to see more data through an agent, that is a change to their Innovator permissions, not to their Edge AI role.

Users Without a Role



A person with no assigned Edge AI role cannot reach the assistant or the Agent Management Platform. They see an access denied message telling them to contact an administrator.

Assign roles as part of onboarding, before you tell people the service is available. See **Getting Started**, and [CIAM user documentation / User Management](#) for how to assign roles and manage them through groups.

Agents

An agent is a configured assistant. It carries instructions that shape how it behaves, a model that does its reasoning, the tools it is allowed to call, the skills that teach it your domain, and the Access Points that connect it to your data. Configuring an agent is deciding what it should be able to do, then confirming it behaves that way before your users meet it.

Your tenant starts with one agent, the Aras AI Assistant, and you can create as many more as you need.

The Agent Library

Every agent in your tenant is listed together with its publish state and whether Aras manages it or your organization does. Open an agent from here to configure it, or create a new one.

The Aras AI Assistant

Every tenant starts with a complete, production-ready agent that Aras builds, tunes and maintains.

The Aras AI Assistant arrives as a Deep Agent at high reasoning effort, with the full set of Aras-managed skills selected, the built-in Innovator tools available, and agent-to-agent delegation ready to use. Connect it to an Access Point, choose the tools you want it to use, publish it, and your organization has a capable PLM assistant without designing one.

Its real value is that it keeps getting better on its own. Aras refines its instructions, extends its skills, and adds tools with every release, and those improvements arrive in your tenant automatically. You are not maintaining a prompt library or tracking which techniques have improved since you deployed. That is also why Aras owns its core: the name, agent type and instructions stay under Aras control so the agent can keep advancing without breaking what you have built on top of it, and it cannot be deleted.



You extend it freely. Add your own skills, connect MCP tools, nominate sub-agents, and assign a different model where your organization requires one.

When you want an agent that is entirely yours, start from a template instead. The AI Assistant template gives you the same baseline, which you then own outright and adapt however your processes demand. See **Creating an Agent**.

Important

The two paths trade differently. The Aras AI Assistant stays current automatically and is the right default for broad PLM capability. An agent you build from a template is yours to shape without limits, and does not receive Aras updates. Many organizations run both: the assistant for general use, and purpose-built agents for specific processes.

Creating an Agent

Creating an agent takes two decisions, in order. First the agent type, which determines how the agent reasons. Then the template, which gives you a starting configuration for that type.

Choosing an Agent Type

Both types are general reasoning frameworks rather than PLM-specific behavior. They differ in how the agent approaches a question.

A **ReAct** agent works in a single loop. It reads the question, decides on one action, takes it, looks at the result, and decides what to do next. It keeps going until it can answer. This is direct and quick, and it suits questions that resolve in a few steps.

A **Deep Agent** plans before it acts. It writes out the steps it intends to take, then works through them, adjusting the plan as it learns. This suits investigative questions where the answer has to be assembled from several retrievals, and where the right next step is not obvious until the previous one returns.

The trade-off is straightforward. A Deep Agent handles harder problems, and it does so by running more loops, which means users wait longer for an answer and each answer consumes more tokens. A ReAct agent answers faster and more cheaply on work that does not need a plan.



Choose ReAct for focused lookups and reporting. Choose Deep Agent for analysis that spans several objects or relationships, and for agents that coordinate sub-agents.

Choosing a Template

A template is a starting configuration: instructions, tool selections and skill selections already filled in, so you are adjusting rather than starting from nothing.

Template	Agent type	Use it for
General Aras Assistance	ReAct	General-purpose PLM questions answered directly
AI Assistant	Deep Agent	The same baseline as the Aras AI Assistant, as an agent you own and adapt
A2A Orchestrator	Deep Agent	An agent whose job is to route work to specialist sub-agents

Templates are scoped to an agent type, which is why you choose the type first.

A template is a one-time copy. From the moment the agent is created it is independent, and later changes to the template do not reach it. That independence is the point: the agent is yours.

Configuring an Agent

The agent editor is where you decide what the agent is and what it may do.

- **Identity.** The name and a description. Users see the description when choosing between agents, so write it for them rather than for yourself.
- **Instructions.** Solution instructions define the agent's behavior, scope and tone. Say what it should focus on and what it should leave alone, for example to concentrate on change history and avoid surfacing unreleased parts. Platform instructions are authored by Aras and apply across all interactions.
Every agent also carries system guardrails, internal instructions that enforce security and data integrity. Instructions shape behavior within those boundaries. An instruction that conflicts with a guardrail is ignored, so you can write instructions freely without risking the platform's guarantees.
- **Model.** Choose from the model configurations available in your tenant. See **Models** on this page.
- **Tools.** Every tool an agent can call comes from a registered MCP server, and the editor groups them that way. The Aras-managed server carries the Innovator tools, subgrouped by the



Access Point each one acts through; servers you register yourself carry everything else. Select the tools this agent needs. See **MCP Servers** on this page.

- **Skills.** Select the skills that teach this agent your domain patterns. Aras-managed skills are available to every agent; you can add your own alongside them, and disable an Aras skill on agents you own. See **Skills**.
- **Sub-agents.** Nominate the agents this one may delegate to. Where a sub-agent proposes a change to data, the approval surfaces in the user's own conversation rather than somewhere they cannot see it. See **Governed Data Changes** on this page.

Important

Tool selection is the boundary of what an agent can do. Instructions shape how an agent behaves; tools decide what it can touch at all. An agent can only call the tools it has been granted, and no instruction widens that. Grant changes take effect on the agent's next run.

Conversation Starters

Conversation starters are suggested prompts shown when a user opens a conversation with nothing in it. Selecting one places the text in the input, where the user can read and edit it before sending.

They are the cheapest way to teach a new user what an agent is good at. Write them as real questions someone in that role would actually ask.

Testing Before Publishing

Simulation Chat lets you use an agent while it is still in Draft, before any end user can reach it. It is available to Edge AI Administrators only.

Send prompts as a user would, then expand the reasoning steps on a response to see which data the agent queried and what it decided along the way. Simulation runs against live tenant data under your own identity and permissions, so what you see is what a user with your Innovator access would see.

Use it to confirm the agent is querying the right data and returning results in the shape you expect, before you expose it.



Publishing, Archiving and Deleting

An agent has two independent states, and it helps to keep them apart.

- **Publish state** is **Draft** or **Published**, and it controls whether end users can reach the agent. Only Published agents appear to Edge AI Users. Publishing is an explicit action you take once the agent has an Access Point and its tools selected and saved. Editing a published agent returns it to Draft automatically. You publish it again when you are satisfied, which means users never encounter a half-edited agent.
- **Activity status** is separate and describes the agent's availability: active, disabled, error, or archived. An agent is reachable by end users only when it is both active and published. Archive is a soft delete. An archived agent disappears from end users whatever its publish state, and Restore brings it back. Use Archive when an agent is out of service but you may want it again.
- Delete is permanent and cannot be undone.
- All of these actions are restricted to Edge AI Administrators. See **Roles and Access** on this page.

Governed Data Changes

Agents do not only answer questions. They create records, update properties, move items through lifecycle states, and delete items. They do none of it without a person saying yes.

The principle is simple: people act through AI within the rights they already have.

What an Agent Can Change

An agent with the right tools can:

- create a record
- update properties on an existing record
- change an item's lifecycle state
- delete an item

Two Layers of Governance

Before an agent can change anything, two decisions must have been made, in two different places.



- **In Edge API.** The API must expose a write action, Add, Edit or Delete, on at least one ItemType. An API that exposes only read actions cannot be used to change data by any agent, under any configuration. See [Building your API schema](#) .
- **On the agent.** When you activate an API for an agent, you select which of its tools the agent may use. A write action becomes available to that agent only once you assign the corresponding tool.

Both must be true. Either one on its own leaves the agent read-only.

The approval flow travels with the tools themselves. A tool that changes data carries the human-in-the-loop validation, so the approval step cannot be separated from the operation it protects.

Important

The two layers give you controls at different scopes. Restricting write actions in Edge API applies to every agent and every user at once. Withholding a write tool from an agent applies to that agent alone. Use the API layer for organization-wide policy, and the agent layer for per-agent scope.

Approval Is Not Optional

Human approval is a platform invariant, not a setting. The agent reasons, retrieves and proposes freely. Before any operation that changes data, it stops and waits for the person it is working with.

This holds for every agent, every operation, and every user. There is no configuration that commits a change automatically.

How Approval Happens

The request for approval arrives inline in the conversation, as a card showing three things: the action the agent intends to take, the item it will act on, and the data it will write. The user approves or rejects it there, in the conversation, with no separate screen and no approval queue to visit.

Rejecting makes no change and ends the action cleanly. The conversation carries on, so the user can correct the instruction and let the agent propose something different.

What Approval Authorizes



Approval authorizes the attempt, not the outcome.

The write executes through Edge API, and Innovator decides whether it succeeds. Permissions, lifecycle maps and workflow rules all still apply. If Innovator refuses the operation, nothing is written and the agent reports the reason back in the conversation.

Where the Access Point uses End User authentication, the write carries the approver's own Innovator identity. Approving an agent's proposal is then no more permissive than that person making the change by hand, and an agent cannot use approval to exceed the rights of whoever gave it.

Attribution and Audit

A change commits under the identity the Access Point authenticates with.

With End User authentication, the change is committed as the person who approved it and appears in Innovator's history as their change. With API Key authentication, it commits as the user the key belongs to, so Innovator records that account rather than the approver.

Write operations are logged with the same detail as reads: item type, item ID, user and timestamp. Your existing Innovator history and audit trail remain the record of what happened to your data, whether the change came through an agent or through the interface.

Important

Authentication mode decides attribution. Where changes must be traceable to a named person, and where a user should never be able to change more through an agent than by hand, bind write-capable agents to Access Points that use End User authentication. See Access Points.

Delegated Work

Where an agent hands part of a request to a sub-agent, and that sub-agent proposes a change, the approval surfaces in the user's own conversation.

Work moves between agents. Approval never moves away from the person.

Access Points



An Access Point is a registered Edge API connection. It is how agents reach your Innovator data, and what the connected API exposes is the outer boundary of what any agent bound to it can see or change.

Access Points are tenant-scoped and reusable. Register one and any number of agents can use it.

Registering an Access Point

You provide a name and description, the Edge API endpoint URL, and the authentication mode. Name and describe the connection for the colleague who inherits it, not for yourself: which Innovator instance it reaches and what it is for.

Choosing an Authentication Mode

This is the most consequential decision on an Access Point, and it is worth deciding deliberately rather than accepting a default.

- **End User** propagates the identity of the signed-in person through to Edge API. Every request, for schema and for data alike, runs under that person's own Innovator permissions, so nobody can retrieve or change data through an agent that they could not reach directly in Innovator. Changes are attributed to them. Nothing is stored on the Access Point, so there is no credential to rotate and nothing that could outlive a person's access.
- **API Key** authenticates with a static credential that you create in Edge API and enter here. Everything runs under the permissions of the user the key belongs to, and changes are recorded against that account rather than the person who approved them. Choose the key deliberately: it defines both what agents bound to this Access Point can reach and the account your audit trail will show. See [Managing API keys and Access tokens](#) for creating and rotating keys.

End User authentication requires the connected Innovator instance to have a valid trusted domain certificate, so that Edge API can identify the requester.

Important

Use End User authentication for any agent people talk to. It keeps Innovator's permission model intact end to end, it makes every change traceable to the person who approved it, and it leaves you no credential to manage. See Governed Data Changes.



Testing a Connection

Where an Access Point uses API Key authentication, you can test the connection and the platform reports the result. Use it after registering the connection, and again whenever you rotate the key, to confirm the endpoint is reachable and the credential is accepted.

Before you publish an agent on a new connection, also send a few prompts in Simulation Chat. That exercises the connection the way an agent will, against the item types you actually expect people to ask about. See **Agents** on this page.

What the Connected API Must Expose

Some capabilities depend on specific item types, properties and actions being exposed in Edge API. Where they are missing, the agent tells the user it could not retrieve the data rather than failing silently.

Capability	Required in Edge API
Changing data of any kind	An Add, Edit or Delete action on the relevant ItemTypes
Version and generation history	The <code>GetItemAllVersions</code> action and the generation property on the relevant ItemTypes, and the History Container and History ItemTypes exposed with <code>item_config_id</code> , <code>item_keyed_name</code> , <code>source_id</code> and <code>config_id</code>
Queries filtered by owner, creator or assignment	The User ItemType with <code>first_name</code> , <code>keyed_name</code> , <code>last_name</code> and <code>login_name</code> , plus the Alias relationship; and the Identity ItemType with <code>keyed_name</code> and <code>name</code> , plus the Member relationship

See [Building your API schema](#) for how to expose them, and **Configure Edge API for Item Version and History Queries** and **Configure Edge API for User and Identity Queries** in the [Get Started](#) guide for the detailed configuration.

Because agents read the schema at runtime, extending what an API exposes reaches your agents on their next request.

Removing an Access Point

An Access Point cannot be removed while an agent is using it, so retiring one is a sequence rather than a single action.

Deal with the dependent agents first. Either point each one at a replacement Access Point, or reconfigure it so it no longer needs this connection. Once no agent uses it, remove the Access Point.



Work in that order because your users are on the other end of those agents. Reconfiguring the agents first means nobody finds an agent unable to answer because the connection underneath it has gone.

Configuring the APIs Themselves

Access Points connect to APIs; they do not define them. The Access Points list links through to the API Management System, where the APIs and their schemas are configured. If you have not created an API yet, start there.

Models

A model configuration tells InnovatorEdge AI which language model to use for reasoning and how to reach it. Every agent uses one.

Aras-Managed and Tenant-Owned Models

Aras provides a model configuration to every tenant, maintained by Aras and available to you read-only. Agents use it unless you decide otherwise, so a new tenant has a working model from the first day with nothing to configure and nothing to procure.

Alongside it you can register your own. A tenant-owned configuration is yours to edit, test and remove.

Registering Your Own Model

Register a model from the Models screen. You provide a name, the provider, one or more deployment names, and the credential. Where the provider requires them, you also provide an endpoint and an API version. The first deployment you add is the default for that configuration.

Supported providers are Azure OpenAI and Microsoft AI Foundry, OpenAI, Anthropic, and Google. An OpenAI-compatible endpoint, such as a self-hosted deployment served through Ollama or vLLM, registers under the OpenAI provider type with a custom endpoint.

There are three common reasons to do this: your organization already has a model subscription it wants to use, a particular model suits your requirements for capability, cost or latency, or you want inference to happen on infrastructure you control.



Important

Registering your own provider is how you decide where inference happens. Prompts and responses are processed by the endpoint you nominate, inside the environment you run it in.

Testing a Model Configuration

Test the configuration to confirm the platform can reach the provider and that the credential is accepted. The result is reported back to you.

Before you publish an agent on a newly registered model, also send a few prompts through Simulation Chat. That exercises the model the way an agent actually uses it, including the agent's reasoning settings, which a connection test does not cover. See **Agents** on this page.

Assigning a Model

Set a tenant default so that every new agent starts on an approved model, and assign a different model to an individual agent in the agent editor where that agent has different needs. See **Agents**.

The Aras AI Assistant uses the Aras-provided configuration unless you assign it another one. If you do, it stops receiving Aras updates to its model configuration, while everything else it inherits from Aras continues to arrive. See **Agents**.

Skills

A skill is a reusable bundle of instructions that teaches an agent how to handle a particular kind of request: the query patterns your organization uses, the rules that apply to a domain, the conventions your data follows. Skills are how an agent learns your organization rather than PLM in general.

Skills are tenant-scoped. Write one and it is available to every agent in your tenant, so a pattern you get right once can be reused everywhere.

Aras-Managed Skills

Aras provides a set of skills covering common PLM work. They are available to every agent, read-only, and Aras extends and refines them with each release.



You can disable an Aras skill on an agent you own, and it stays disabled through later updates. Your decision is not undone the next time Aras ships an improvement.

Creating a Skill

A skill has three parts.

- **Name.** Required, and unique within your tenant. It is how you identify the skill when binding it to an agent, so name it for the situation it applies to rather than for the mechanism it uses.
- **Description.** A short summary of what the skill teaches. This is what you and your colleagues read when deciding whether an agent needs it, so be specific about when it applies.
- **Explanation.** The instruction text itself, up to 10,000 characters, and this is what the agent reads.

Writing a Skill That Works

The explanation is written for the agent, not for a person, and the difference matters.

Teach a pattern rather than a fact. A skill that says which property holds the approving engineer, and how your organization words that question, will keep working as your data grows. A skill that lists this quarter's programs will not.

Keep one skill to one concern. Small skills combine; a large one that covers three domains has to be bound to every agent that needs any of the three.

Use the room you have. Ten thousand characters is enough for worked examples, and examples teach an agent more reliably than abstract rules. Show a question your users actually ask and the shape of the answer you expect.

Say what to avoid as well as what to do. Naming a mistake you have seen an agent make is often the most valuable line in a skill.

Then test it. Bind the skill to a draft agent and work through real questions in Simulation Chat, expanding the reasoning steps to see whether the agent actually followed it. See **Agents**.

Important



A skill changes how an agent approaches a problem. It cannot widen what the agent can reach. Data access comes from Access Points and capability comes from tools, so a skill can never grant an agent access to something it was not given. Write skills freely; they cannot open a door.

Duplicating a Skill

Duplicate an existing skill when you need a variant. You get an independent copy, opened for editing, and changes to either one leave the other alone.

This is the practical way to adapt a skill for a second domain, or to try a rewrite while the original stays in service.

Removing a Skill

If a skill is bound to agents, you are shown which ones before the deletion goes ahead. Read that list. Removing a skill changes how those agents behave the next time someone talks to them, and the agents will not tell you what they lost.

MCP Servers

MCP Servers is the tool registry for InnovatorEdge AI. Every tool an agent can call is registered here, whether Aras provides it or your organization adds it.

The Model Context Protocol, MCP, is how an agent talks to a tool. InnovatorEdge AI applies it uniformly: the Aras tools that reach your Innovator data are deployed as an internal MCP server and speak the same protocol as anything you register alongside them. There is no separate mechanism for built-in tools and third-party tools.

That uniformity is the reason this screen matters more than its name suggests. It is the complete inventory of what agents in your tenant are capable of doing, in one place, under one governance model.

The Aras-Managed Server

Aras deploys its Innovator tools as an internal MCP server, present in every tenant and maintained by Aras. These are the tools that retrieve items, search, walk BOM and where-used relationships, read lifecycle state, and, where you have enabled them, change data.



Within this server the tools are grouped by the Access Point each one acts through, so you can see which Innovator connection a given tool reaches. Aras extends this set with each release, and new tools appear in the registry without any work from you.

The Aras-managed server is also the only one that can act as the signed-in user. That is what allows Innovator's permission model to govern every query and every change an agent makes. See **Access Points**.

Registering an External MCP Server

Register a server to bring its tools into the registry. You provide a name, the server's URL, the authentication mode, and the credential where the server requires one.

Write a description. The next administrator needs to know which system this is and why an agent talks to it, and a URL alone will not tell them.

Authentication for External Servers

An external server is registered either with **no authentication**, for a public server that requires no credential, or with an **API key** that you supply.

Either way, the server is reached with what you configure on it rather than with the identity of the person using the agent. End-user identity is propagated to the Aras-managed server, and that is where Innovator's permissions apply.

Important

This asymmetry is the thing to understand before you register anything. Tools on the Aras-managed server run as the signed-in user, so Innovator's permissions govern them. Tools on an external server run with the credential you configured, so every person using that agent acts on the external system with those same rights. Grant external tools narrowly, and scope the credential to exactly what the agent legitimately needs.

Discovered Tools

Once a server is registered, the platform reads the set of tools it declares. That list belongs to the server and you do not edit it here. What you decide is which of those tools any given agent may use.



Testing an MCP Server

Test the server to confirm it is reachable, that the credential is accepted, and that tool discovery succeeds. Do this before binding any of its tools to an agent, so a discovery problem surfaces while you are configuring rather than while someone is asking a question.

Granting Tools to Agents

Tools are selected per agent, in the agent editor, where the registry appears grouped by server. An agent can call only the tools it has been granted, and a change to its grants takes effect on its next run. See **Agents**.

Reviewing the registry against your agents is worth doing periodically. It is the one view that answers what the agents in this tenant can actually do, which is a different question from what they were built to do.

Removing an MCP Server

Where agents use tools from the server, you are shown which agents are affected before the removal goes ahead.

Reconfigure those agents first, as you would when retiring an Access Point. An agent that loses a tool it relied on simply stops being able to do that part of its job, and it will not announce the loss.

Troubleshooting and Known Behaviors

Most reports that reach an administrator come down to configuration rather than to anything going wrong. This chapter starts with the checks that resolve the majority of them, then covers behaviors worth knowing about before a user reports them.

Start Here

Three questions settle most cases.

- **Is the agent published?** Only published agents are visible to Edge AI Users, and editing a published agent returns it to Draft. An agent that worked yesterday and is missing today has usually been edited and not published again. See **Agents**.



- **Does the user have an Edge AI role?** Without one they reach neither the assistant nor the platform. See **Roles and Access**.
- **Does the connected API expose what the question needs?** An agent can only work with the item types, properties and actions the Edge API exposes. This is the cause behind most "the agent cannot answer this" reports. See **Access Points**.

Common Symptoms

Symptom	Likely cause	What to do
A user cannot see an agent that exists	The agent is in Draft or archived, or the person has no Edge AI role	Publish or restore the agent, or assign the role. See Agents and Roles and Access .
An agent that was working is no longer available to users	It was edited, which returned it to Draft	Publish it again. See Agents .
The agent says it cannot find an item type you know exists	The item type is not exposed in the connected Edge API	Expose the item type and the properties the question needs. See Access Points .
Answers are missing records that should be included	On an Access Point using API Key authentication, the key's user cannot see those records	Use a key whose user can see the full schema, or switch the Access Point to End User authentication. See Access Points .
Two people get different answers to the same question	The Access Point uses End User authentication, so each answer is scoped to that person's Innovator permissions	This is working as intended. Compare their Innovator access. See Roles and Access .
Questions about who created, owns or modified an item do not resolve	The User and Identity item types are not fully exposed	Expose them with the properties and relationships listed in Access Points.
Version or history questions do not resolve	The history item types and the version action are not exposed	Expose them as listed in Access Points.
The agent will not change data even though the user approves	The API exposes no write action for that item type, or the agent has no write tool	Check both layers. See Governed Data Changes .
A connection test fails	The endpoint is unreachable or the credential is no longer valid	Correct the endpoint or rotate the key, then test again. See Access Points .
An agent cannot use an external tool	The tool is not granted to that agent, or the MCP server is unreachable	Grant the tool, or test the server. See MCP Servers .

Known Behaviors

- **Coverage varies by Aras application.** The assistant works against your schema without hardcoded item definitions, so any item type exposed through Edge API can in principle be queried. Item types belonging to some Aras applications, including Variant Management, Technical Documentation and Simulation Management, depend on schema exposure that is not always complete. Validate the questions you care about against those applications before you tell those teams the assistant covers them.
- **Results are bounded.** Retrieval is bounded per call, and large aggregations may be truncated. When a result is capped or truncated, the assistant says so and asks the user to narrow the request, so a bounded answer is visible rather than silent.
- **Calculations report their inputs.** For a calculation across a set, the assistant states how many records it included, because records with incomplete data are excluded. Treat that count



as part of the answer, and expect questions about it.

- **"History" is ambiguous in a short prompt.** Asking for the history of an item may return version and generation information rather than the full audit trail. Asking for the full history, or the action history, resolves it, and so does clarifying in the next message.
- **Some item types show an identifier in the context chip.** The chip above the chat input occasionally shows an item type and identifier instead of a readable name. The context itself is correct; only the label is affected.
- **Context follows the active item in Innovator.** When several item tabs are open, confirm the chip is pointing at the record the user means before relying on the answer. Removing the chip and asking again with the intended record open is the quickest way to be certain.
- **A manually stopped response ends that turn completely.** If a user stops the assistant mid-answer, any skill that had loaded during that turn is not carried into the next request in the same conversation. Starting a new conversation restores normal behavior.

Getting Support

If a problem persists, raise it through the Aras Subscriber Portal with:

- the release number, from the About panel
- your tenant ID, from the profile menu
- the name of the agent
- the name of the Access Point and the Edge API endpoint it uses
- the question the user asked, and what came back

Those details let support reproduce the case against your configuration rather than a generic one, which is usually the difference between one exchange and several.

Important

A support case leaves your organization. Review what you attach before you send it, and sanitize or anonymize anything covered by your intellectual property or confidentiality obligations. Conversation transcripts, error messages and screenshots often carry part numbers, program names, supplier names and specifications that support does not need. What support does need is the shape of the problem: the item type, the property, the action attempted and the response received. Replacing real identifiers with placeholders keeps the case diagnosable without exposing the design.



User Guide

Introduction

Most of what you need to know during a working day is already in Aras Innovator. Finding it has meant knowing where to look, knowing how to build the query, or asking the person who does.

The Aras AI Assistant answers the question instead. You ask in plain language and it answers from your own Innovator data, with a link to the record the answer came from.

It can also change data when you ask it to, and it always asks you to approve the change before anything is written. See **Changing Data** on this page.

Who This Guide is For

Anyone who works with Aras Innovator data. No knowledge of the data model or of any query language is assumed. If you can describe what you want in a sentence, you can use the assistant.

Check an Answer Before You Rely on It

The assistant works from your live Innovator data and shows you where every answer came from, which makes checking straightforward. Do check, whenever the answer is going into a decision.

AI-generated output must be reviewed and validated by a qualified person before it is used in any operational, engineering, manufacturing, compliance or business decision. Understanding an Answer describes what to look at, and Terms of Use gives the requirement in full.

Before You Start

The assistant becomes available once your organization's tenant is configured and you have been granted access. If you cannot see it at all, or you can sign in but no agents are listed, contact your Edge AI Administrator.

AI Where You Work



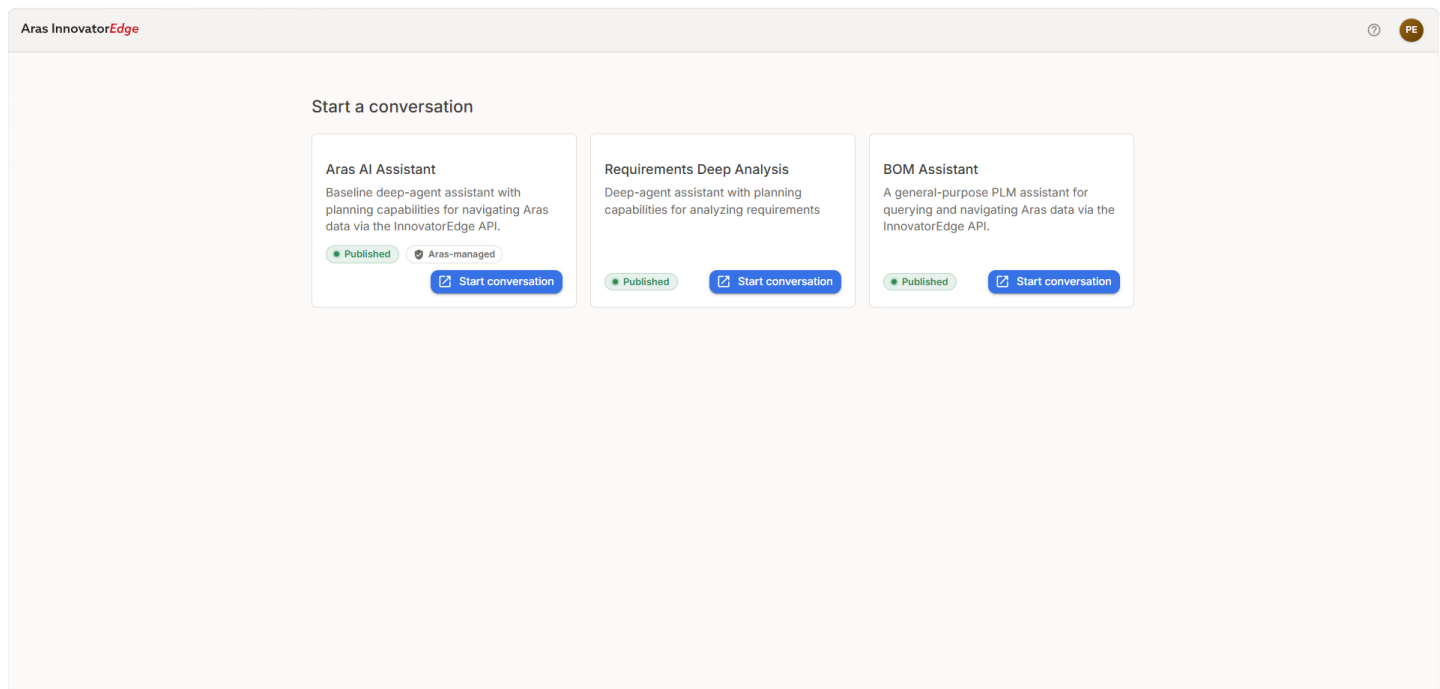
You can use the assistant in two places: inside Aras Innovator, alongside whatever you are working on, or on its own when you are not in Innovator at all.

Starting from InnovatorEdge AI

Go to **edge.aras.cloud/ai**. You arrive at a home page listing the agents published for you. Choose one and the conversation opens.

Most organizations publish the Aras AI Assistant, which answers general questions about PLM data. Some publish additional agents for particular kinds of work. Where you see more than one, the description under each name tells you what it is for.

If you are an Edge AI Administrator, the same entry point also leads to the Agent Management Platform. Configuration is covered in the [Administrator Guide](#).



Inside Aras Innovator

Open the assistant panel and it appears beside your work, resizable and dismissible at any time.

Everything you can do in the standalone chat you can do here, and one thing more: the panel knows which record you have open and uses it as context, so you can ask about the item in front of you



without naming it. See **Working in Context** on this page.

For a longer session, expand the panel to full screen. Your conversation and your context come with you, and collapsing it returns you to the panel with everything intact.

The screenshot displays the Aras Innovator application interface. The main window shows a part record for 'MP2941' with various fields like Name, Body Panels, Type, Unit, Make / Buy, Cost, and Effective Date. Below this, there's a 'BOM' (Bill of Materials) section with a table listing components and their quantities.

Seq...	Part Number	Rev...	Name	Type	Quantity	created_by_id...	State	owned_by_id...	Unit	Reference Designator
5	MP2194	B	Thing-O-Matic 2 Front Panel	Component	1	Mike Miller	Released		EA	
10	MP2196	A	Thing-O-Matic 2 Back Panel	Component	1	Mike Miller	Released		EA	
15	MP2199-2	A	New Thing-O-Matic 2 Right Panel	Component	1	Terry Adams	Released		EA	
20	MP2193	C	Thing-O-Matic 2 Top Panel	Component	1	Mike Miller	Released		EA	
25	MP2198	A	Thing-O-Matic 2 Left Panel	Component	1	Mike Miller	Released		EA	
30	MP2195	C	Thing-O-Matic 2 Bottom Panel	Component	1	Mike Miller	Released		EA	

On the right side, there's an 'Aras AI Assistant' chat panel. It shows a list of actions like 'Call eal_query_fetch', 'Reasoning', 'Call write_todo_list', etc. Below the list, there's a text input field with the placeholder 'Let me know what you're looking for...' and a '0/500' character count. At the bottom, there's a link to 'Legal Terms and Privacy Policies'.

Which One to Use

- Use the **embedded panel** when your question is about the record in front of you, or when you want to move between answers and records without leaving Innovator.
- Use the **standalone chat** when you are not working in Innovator, or when your question is not about anything you have open.

The one real difference is item context. Only the embedded panel can see what you are looking at.

Switching Agents

Where more than one agent is published, the agent name in the header lets you change to another one.

Switching starts a fresh conversation with the agent you choose, so if your current conversation already has messages you are asked to confirm before it is cleared.



Your First Question

Open the assistant. With nothing in the conversation yet you see its name, a short description of what it does, and a few suggested questions your administrator has set up. Those suggestions are worth reading, because they tell you what this particular agent is good at.

Select one, or type your own. Something similar to:

Show all released Parts changed in the last 30 days

An answer comes back as a list, and each row links to the record in Innovator. That is the whole loop: ask, read, follow the link to the source.

Asking Questions

Ask in the words you would use with a colleague. You do not need to know which item type holds the answer, which properties are involved, or how the data is related. The assistant works that out and tells you what it did.

What You Can Ask

Finding and filtering items. Retrieve items by state, date, owner, attribute, or any combination.

- List all open ECOs assigned to me
- Show released Documents changed in the last 30 days
- Find Parts with no preferred manufacturer

Product structure. Explore assemblies and trace where a component is used.

- Show the BOM for ASM-7700 at rev C
- Where is Part P-4420 used?
- Which assemblies contain capacitor C-100?

Change activity. Review what a change affects, who raised it, and what is in flight.

- What is affected by ECO-2024-0451?
- Who submitted CR-2025-0033?
- Show change requests opened this quarter



Counts and totals. Get a number, or a breakdown.

- How many Parts are in this assembly?
- How many open changes do we have, by type?
- Count Documents per classification

Status and ownership. Check what is waiting on you or on someone else.

- What is assigned to me right now?
- Show items waiting on my approval
- List Parts created by Riley Miller

History. Retrieve what happened to an item, or how it has been revised.

- Show the full history of Part MP0101
- How many revisions does Part MP2954 have, and what state is each in?

Use `full history` or `action history` when you want the record of what was done.

A shorter phrasing such as "history of Part MP0101" may return revision information instead. If that happens, ask again for the action history.

Diagrams and Charts

The assistant can draw what it found as well as list it.

- Diagram the relationships for ASM-7700
- Show me the data model around Part and Document
- Show a pie chart of Parts by classification

Ask for the form you want in your question, or ask afterwards about a result you already have on screen.

Writing a Good Prompt

A prompt that works usually carries three things:

- **the object** you are asking about



- **the action** you want performed on it
- **the scope** or **format** of the answer you want back

Show released Documents changed in the last 30 days

That is action, object, scope. Drop the scope and you get every document. Drop the object and the assistant has to guess what you meant.

You do not need all three every time, and a follow-up question often needs none of them because the conversation already supplies the context. But when an answer is not what you expected, the missing piece is usually one of the three.

- **Name the item type when your question could apply to several.**
 - Show all open ECOs is clearer than show all open changes because the second could mean several things.
- **Narrow with what you already know.** A date range, a state, an owner or a classification will get you a shorter and more useful answer than a broad question and a long list.
- **Refine rather than restate.** Once you have an answer, build on it with a follow-up instead of writing a new question. See **Working in Context** on this page.
- **Watch what the assistant says it resolved.** If you use a term the system does not recognise, the assistant tells you what it matched, what it could not confirm, and answers with what it was able to establish. That message is the fastest route to a better question, because it tells you exactly which word to change.

Prompts and What They Return

The shape of your question decides the shape of the answer. These examples cover the common ones.

What you ask	What comes back
How many Parts are in 'Released' state?	A single number
List all open ECOs assigned to me	A table of matching ECOs, each row linked to its record
Count Documents per classification	A breakdown, one row per classification
Show a pie chart of Parts by classification	The same breakdown, drawn as a pie chart
Show the BOM for ASM-7700 at rev C	The assembly structure, indented by level
Where is Part P-4420 used?	The assemblies that contain it
Diagram the relationships for ASM-7700	A diagram of the items connected to it



Show the full history of Part MP0101	The changes made to it, with dates and what changed
List Parts created by Riley Miller	Parts filtered by creator. The same works for owned by, modified by and managed by
What is the average number of days for the Design Review activity?	A calculated average, together with how many records it was based on

Asking in Your Own Language

Ask in whichever language you work in. The assistant understands the languages the underlying model supports and answers in the language you used. You can also ask it to reply in a different language at any point in the conversation.

Your PLM data comes back as it is stored. Item names, descriptions and property values are not translated; only the assistant's own explanations are.

Understanding an Answer

An answer is not only the result. It arrives with the record links behind it and with the assistant's own working, and both are there so that you can check it.

Reading the Response

Responses come back as formatted text: paragraphs, lists and tables, with a diagram or a pie chart where you asked for one.

Tables are part of the response rather than a separate grid, so they are for reading rather than for sorting or filtering. If you want the same data ordered differently, or narrowed, ask for it. That is usually quicker than rearranging it yourself would have been.

Where the Answer Came From

Every answer carries links to the Innovator records behind it.

In the embedded panel, following a link opens the record in Innovator and your conversation stays exactly where it was. In the standalone chat, the record opens in a new tab.

Following one link is the fastest check you can make on any answer.

Seeing How the Assistant Worked



Every response carries its reasoning with it, collapsed beneath the answer. Expand it and you see the whole working: the query that was run, what each tool was asked for and what it returned, and the assistant's own description of what it was doing at each stage.

This is not diagnostic detail meant for support. It is the answer's evidence. Open it whenever a result surprises you, because it almost always shows one of two things: that your question was read differently than you meant it, or that the data really does say what the answer says.

Where the agent plans before it acts, the plan appears in a side panel: the steps it intends to take, and the state of each one as it works through them. You can watch it progress and see exactly where it has reached.



Aras InnovatorEdge

Aras AI Assistant

Clear chat

What is assigned on me right now?

Call load_skill

Reasoning

Call who_am_i

Reasoning

Call odata_schema_fetch

ARGUMENTS

entity_set_name: InBasket_Task

RESULT

InBasket_Task: {

label: InBasket,

label_plural: My InBasket,

actions: {

get,

getlist

}

properties: {

id: {

type: String

}

}

Reasoning

Sorting tasks by due date

I'm thinking about sorting by due date in ascending order for urgency, but I wonder if that's too complicated—keeping it simple might be better. I should finalize my query with specific fields like id, keyed_name, name, state, status, due_date, start_date, and created_on. I'm considering if I need the instructions field, since it might be lengthy. Including id seems necessary for the table format, even though the user didn't specifically ask for it.

Call odata_query_fetch

Reasoning

Call odata_query_fetch

Let me know what you're looking for...

0/500

By using this chat, you agree to Aras [Legal Terms](#) and [Privacy Policies](#).

Aras InnovatorEdge

Aras AI Assistant

Clear chat

which open tasks are assigned to me through an alias identity? And which of them require immediate action?

Call load_skill

Call write_todo_list

Reasoning

Call who_am_i

Reasoning

Call odata_schema_fetch

Reasoning

Call write_todo_list

Thinking...

inbasket, with pending tasks waiting for user input while active tasks are already in progress. It seems logical that "immediate action" means 'Pending' tasks need user attention.

I might need two separate filters: one for open tasks assigned to me, considering both 'Pending' and 'Active' states, and another for tasks needing immediate action based on due dates or being in the 'Pending' state. I should check if my workflow skill instructions are fully loaded.**Determining immediate action tasks**

I'm considering how to interpret "immediate action" in relation to pending and active tasks. It seems that "immediate action" might refer to active tasks needing action now. However, open tasks include both types. I think there could be a way to classify tasks as requiring immediate action if they're overdue. I wonder if I should ask the user for

Let me know what you're looking for...

0/500

By using this chat, you agree to Aras [Legal Terms](#) and [Privacy Policies](#).

Plan (4)

Resolve current user identity

Done

Current user is admin; alias identity id is DBA5D86402BF43D5976854B8B48F CDD1.

Fetch InBasket_Task schema and relevant assignment/state fields

Done

InBasket_Task exposes my_assignment, state, status, due_date, name, keyed_name, and assigned_to navigation to Identity.

Query open tasks assigned through alias identity and identify those needing immediate action

In progress

When a Result is Bounded

Retrieval is bounded, so a very broad question can come back with part of the answer rather than all of it. When that happens the assistant says so in the answer, and asks you to narrow the request.

©2026 Aras Corporation All Copyrights Reserved

34

Take it at its word. A narrower question does not just avoid the limit, it usually gives you a more useful answer.

Judging an Answer You Will Act On

Three things, and they take about a minute between them.

- **Open the reasoning and read the filters.** Confirm the assistant narrowed the data the way you intended. This catches most misreadings.
- **Follow one link.** Open a single record from the result and confirm it says what the answer says.
- **Read the count.** Where the assistant has calculated across a set, it tells you how many records it used, because records missing the data it needed are left out of the calculation. If that number is lower than you expected, that is the reason, and the answer is describing a smaller set than you had in mind.

Working in Context

The assistant keeps track of two different things: the record you have open, and the conversation you are having. They behave differently and it is worth knowing which is which.

The Record You Have Open

In the embedded panel, the assistant sees the record you are looking at in Innovator. A chip above the input shows which one, so you always know what "this" refers to.

That means you can ask about the item in front of you without naming it.

With ECO-2025-0100 open in Innovator

You: Who submitted this?

Open a different record and the chip follows: your next question uses the new one.

If your question is not about the record on screen, remove the chip and ask without it.

The Conversation So Far

Follow-up questions build on the answer before them, so you do not have to restate anything.

You: Show all open ECOs assigned to me



Twelve ECOs come back.

You: Which of these were created in the last 7 days?

The list narrows to three.

You: Show the affected items for the first one

Each question does less work than the one before it, and gets you further. This is usually faster than writing one long question, and it lets you change direction when an answer is not what you expected.

When the Assistant Asks You Something

If a question could mean several things, the assistant asks before it goes looking. Ask for **the latest change** without saying what kind, and it may ask whether you mean an ECO, an ECR or an ECN.

Answer it and the assistant continues with the narrower question.

Stopping a Response

If an answer is going somewhere you do not need, stop it with the **Stop** button in the input area. The assistant halts, and you can ask something else straight away.

Starting a New Conversation

New Chat clears the current session: the conversation and everything the assistant was holding from it.

It does not change what you have open in Innovator. If you are working in the embedded panel with a record on screen, that record is still your context, because the panel reads it from Innovator rather than from the conversation.

Changing Data

The assistant can change Innovator data, not only read it. Nothing is written without your explicit approval, every time.

What It Can Change



With the right permissions and configuration, the assistant can create a record, update properties on an existing one, move an item to a different lifecycle state, or delete an item.

If your assistant cannot do these things, your organization has not enabled them. Ask your Edge AI Administrator.

Nothing Happens Until You Approve

When the assistant is ready to change something, it stops and shows you an approval card before anything is written.

The card tells you what it intends to do, which item it will act on, and what will be written. For an update it shows the current value alongside the proposed one, so you can see exactly what changes.

The screenshot shows the Aras InnovatorEdge AI chat interface. At the top, there's a header with the Aras InnovatorEdge logo and a user profile icon. Below the header, there's a chat area with a message from the AI Assistant: "Plan step 2 of 3: Update the Part name to Arm Assembly 20-1000-01". Below this message, there's an approval card titled "Odata Item Update". The card contains the following information:

- Execute odata_item_update**
- entity_set_name**: Part
- id**: B03F51ADF40846C8A0E372DCC6992A77
- data**: {"name": "Arm Assembly 20-1000-01"}

At the bottom of the approval card, there are two buttons: "Reject" and "Approve". Below the approval card, there's a text input field with the placeholder "Let me know what you're looking for..." and a character count "0/500". At the bottom of the chat area, there's a footer with the text "By using this chat, you agree to Aras Legal Terms and Privacy Policies."

On the right side of the chat area, there's a "Plan" section with a list of steps:

- Find the latest Part item for AM-1 and verify its current name (Done)
- Update the Part name to Arm Assembly 20-1000-01 (In progress)
- Confirm the update result (Pending)

The "In progress" step shows the following details:

- Using verified id: B03F51ADF40846C8A0E372DCC6992A77.
- Will report the returned item data after patch.

Approving

Approve and the change is submitted to Innovator under your own identity.

- **If Innovator accepts it**, the change is applied and the assistant confirms it, with the updated record.
- **If Innovator refuses it**, because your permissions do not allow that operation or because a validation rule rejects it, nothing is written and the assistant tells you why.

Rejecting

Reject and nothing is written. The action ends there and the conversation carries on, so you can correct what you asked for and let the assistant propose something different.

What Your Approval Means

Approving authorizes the attempt, not the result.

The change runs under your own Innovator identity, so it can do exactly what you could do by hand and no more. Your permissions, the lifecycle rules and any workflow constraints all still apply.

A change made this way appears in Innovator as your change, in the record's own history.

Important

Read the card before you approve it. It is showing you the change as the assistant understood your request, and understanding is the part worth checking. The permissions behind it are already yours.

Changing Something Back

There is no undo. A change committed through the assistant is a change in Innovator like any other. You can ask for the reverse as a new change, giving the previous value, and it goes through the same approval. The previous value is in the record's history if you no longer have it.

Known Behaviors

A few things behave in ways worth knowing before you meet them.

- **Asking for history may give you revisions.** A short phrasing such as "history of Part MP0101" can return revision and generation information rather than the record of what was done. Ask for the full history or the action history, or just say so in your next message.
- **Some items show an identifier rather than a name in the context chip.** The chip may show an item type and a number instead of a readable name. The context is still correct; only the label is affected.
- **Stopping a response ends that turn completely.** If you stop the assistant mid-answer, the next question in the same conversation starts clean. If the assistant then behaves differently than it did before you stopped it, start a new conversation.



Getting Help

Ask Your Administrator First

Your Edge AI Administrator can resolve most things faster than support can, because they are configuration rather than faults.

Contact them when the assistant is not there at all, when you sign in and no agents are listed, or when the assistant cannot answer about data you know exists. That last one usually means the data has not been made available to the assistant yet, which is something they can change.

Documentation and Support

Documentation and the Aras subscriber portal are both reachable from the help menu.

You can raise a problem in the subscriber portal yourself. Include the agent you were using, the question you asked, what came back, and what you expected instead.

Important

A support case leaves your organization. Before you attach a conversation, an error message or a screenshot, check it for anything covered by your organization's confidentiality or intellectual property obligations, and replace real part numbers, program names and supplier names with placeholders. What support needs is the shape of the problem, not your design data.

Terms of Use

AI-generated output requires human review. AI-generated outputs must be reviewed and validated by a qualified person before use in any operational, engineering, manufacturing, compliance, or business decision. See the [Aras InnovatorEdge Product Specific Terms](#) for full details.

Customer data is not used for model training. Aras does not use customer data, user prompts, inputs, or AI interaction data submitted through InnovatorEdge services to train, retrain, or improve any general-purpose or shared AI or machine learning models. See the [Aras InnovatorEdge Product Specific Terms](#) for full details.

Session data is not retained. The assistant maintains short-term context within an active conversation so that follow-up questions work. That context is not stored or retained after the session ends.



Appendix: Glossary

Essential terms and their meaning in the context of Aras InnovatorEdge AI.

Term	Definition
Access Point	A registered Edge API connection as it appears in the Agent Management Platform. Access Points are tenant-scoped and reusable across multiple agents. At least one is required before an agent can be published. See also Access Points in the Administrator Guide .
Activity status	The health and availability state of an agent, separate from its publish state. An agent's activity status is active, disabled, error, or archived.. End users can reach an agent only when it is both active and published.
Agent	A configured AI instance that interprets user questions, selects the appropriate tools, and returns results. Agents are created from templates and governed by solution instructions, platform instructions, bound skills and tools, and bound Access Points.
Agent Library	The central screen in the Agent Management Platform, where administrators create, configure, publish, and manage agents.
Agent Management Platform	The management application for InnovatorEdge AI, where administrators configure agents, models, skills, MCP Servers, Access Points, and access. Reached at its own route, separate from the chat application. Available only to Edge AI Administrators.
Agent Overview	A read-only summary available after an agent is first saved. Displays the agent's identity and configuration state.
Agent type	The reasoning approach an agent uses. Two agent types are available: Deep Agent (plan-and-execute) and ReAct Agent (iterative reason-and-act).
Approval card	The inline card presented when the Aras AI assistant proposes a data change. Shows the action type, target item, and data payload. The user approves or rejects before any change is executed.
Aras AI Assistant	The default governed agent that ships with every tenant. Pre-configured by Aras with locked core settings and editable extension points.
Archived	An agent that has been soft-deleted. An archived agent is hidden from end users regardless of its publish state, and can be returned to service with Restore.
Authentication mode	The authentication method used by an Access Point or an MCP Server. End User propagates the signed-in user's identity, so calls execute under that user's permissions and no credential is stored. API Key uses a static credential held in the secret store, and calls execute under the permissions of the user the key belongs to. MCP Servers additionally support no authentication; End User is not selectable for MCP Servers.
Built-in tools	The Innovator data tools Aras provides with the platform, covering item retrieval, search, BOM retrieval, relationship traversal, and lifecycle state queries. Made available to an agent by selecting them on the agent, from an Access Point the agent is bound to.



Centralized Identity Access Management (CIAM)	Centralized Identity Access Management, the Aras identity service that authenticates users and carries their Edge AI role assignment. Edge AI is registered as a single CIAM application.
Context awareness	The embedded panel's ability to detect the Innovator item the user is viewing and use it as implicit context for questions.
Context chip	The indicator above the chat input showing which Innovator record is currently in context. The user can remove it to ask an unrelated question.
Conversation starters	Suggested prompts configured by the administrator that appear on the start screen when no conversation is active. Selecting one places the text in the input for review before it is sent.
Deep Agent	An agent type that plans before it acts: it writes out the steps it intends to take, then works through them, adjusting as it learns. Suited to questions whose answer must be assembled from several retrievals. The Aras AI Assistant is a Deep Agent.
Delete	Permanent removal of an agent. Unlike Archive, a deleted agent cannot be restored.
Draft	The publish state of an agent that end users cannot reach. Draft agents are visible only to Edge AI Administrators, who can test them in Simulation Chat.
Edge AI	The InnovatorEdge service that delivers conversational intelligence and governed action on top of Aras Innovator data. Depends on Edge API for all data access.
Edge AI Administrator	Tenant-level administrator role. Manages agents, models, skills, Access Points, and MCP Servers within one tenant. Can simulate draft agents and converse with published agents.
Edge AI User	End-user role. Can converse with published agents through the embedded panel or the standalone application. Cannot access the Agent Management Platform.
Edge API	The InnovatorEdge service that exposes Aras Innovator data and actions as governed endpoints. All Edge AI data interactions flow through Edge API. Agents read the Edge API schema at runtime, so changes to an API definition take effect on the next request.
Edge API connection	A registered, tenant-scoped link between Edge AI and a configured Edge API endpoint. Stored once and reusable across multiple agents. Displayed in the Agent Management Platform as an Access Point; the two terms refer to the same object.
Embedded Panel	The in-Innovator access mode for the Aras AI Assistant: a resizable conversational panel that operates alongside the Innovator workspace without interrupting the user's active context.
Ephemeral Interaction	The data handling model for LLM interactions within Edge AI. Data passed to the language model for reasoning is not stored, retained, or used for model training after the interaction completes.
Governed agent	An agent whose core configuration is locked by Aras and cannot be changed by the tenant. For the Aras AI Assistant the locked settings are name, agent type, solution instructions, and platform instructions, and the agent cannot be deleted. The LLM configuration is not locked and can be changed by an Edge AI Administrator.
Human-in-the-loop (HITL)	The requirement that a person explicitly approves any operation that changes Innovator data before it is committed. In InnovatorEdge AI this is a platform invariant rather than a configurable option: there is no setting that enables automatic commit. The validation flow is carried by the tools that change data.

LLM configuration	A registered model provider and deployment that an agent uses for reasoning. Aras provides a default configuration to every tenant, and administrators can register their own. Overriding the Aras default stops the tenant from receiving Aras updates to it.
MCP Server	An external tool server that provides additional tools to agents. The platform discovers the tools a server offers and makes them available for binding to agents.
Metadata Refresh	The process of forcing Edge AI to re-read and re-cache an Edge API schema after the underlying API definition has been updated. Triggered by editing and re-saving the connection in Access Points.
Models	The registry of AI model providers available for agent configuration. Model selectors distinguish between Reasoning Models (used for language understanding and response generation) and Embedding Models (used for semantic search and knowledge retrieval).
Platform instructions	Aras-authored instructions that govern platform-level agent behavior. Locked for the Aras AI Assistant.
Polyitem	An Aras Innovator data concept representing an item type that can resolve to multiple concrete item types depending on context. The Aras AI Assistant can query relationships involving Polyitem types.
Published	The publish state of an agent that end users can reach. An administrator publishes an agent explicitly, once it has an Access Point and the tools it needs selected and saved. Only published agents appear to Edge AI Users.
Role-Based Access Control (RBAC)	The Aras Innovator permission model applied to all data queries executed by Edge AI agents. Users can only retrieve data they are already authorized to see in Innovator.
ReAct Agent	An agent type that works in a single loop: it reads the question, takes one action, looks at the result, and decides the next action. Suited to questions that resolve in a few steps.
Reasoning effort	The depth of reasoning an agent applies to a request. The Aras AI Assistant ships at High.
Restore	The action that returns an archived agent to service.
Schema Caching	The platform's local storage of Edge API metadata. Keeps agent responses fast and ensures the agent works from a consistent view of the data structure. Refreshed by editing and re-saving the relevant connection.
Session context	Short-term conversational context retained within a single user session. Enables follow-up questions to refine previous results. Not retained after the session ends.
Simulation Chat	An administrator-only environment for testing an agent against live data before publishing. Each response includes an expandable reasoning trace. Simulation works on draft agents, so an administrator can test an agent before end users can reach it.
Skill	A reusable instruction bundle that teaches an agent domain-specific query patterns, rules, or behaviors. Skills are bound to agents through the agent editor. Aras-managed skills are read-only; an administrator can disable one on their own agents, and it stays disabled through later updates.
Solution instructions	Instructions that define an agent's behavior, scope, and personality. Locked for the Aras AI Assistant.
Standalone application	The out-of-Innovator access mode for the Aras AI assistant, reached at its own URL, separate from the Agent Management Platform. Offers the same conversational capabilities as the embedded panel.



System guardrails	Internal instructions built into every agent that enforce security, data integrity, and platform governance. Custom instructions that conflict with system guardrails are automatically ignored.
Template	A pre-built agent definition that provides a starting configuration (solution instructions, tool bindings and skill bindings) for a new agent. A template is a one-time copy source: an agent created from it does not inherit later changes to the template and does not receive Aras updates. The Aras AI Assistant is not delivered through a template.
Tenant	The organizational account unit within InnovatorEdge. Access Points, agents, models, skills, MCP Servers, and user access are all scoped to a tenant.
Test Connection	A check available on an Access Point that uses API Key authentication, confirming that the endpoint is reachable and the credential is accepted. The platform reports the result.
Tool	A single callable capability available to an agent, such as retrieving an item or traversing relationships. Tools come from Access Points, for Innovator data, or from MCP Servers, for external systems, and are selected per agent. Tools that change data carry the human-in-the-loop approval flow.

Frequently Asked Questions

Setup and Prerequisites

Do I need Edge API to use Edge AI?

Yes. Edge API is a prerequisite. All data interactions between agents and Aras Innovator flow through configured Edge API connections. Edge API must be active and connected to your Innovator instance before enabling Edge AI.

Can Edge AI connect directly to the Aras Innovator database?

No. All queries are executed through Edge API endpoints under the user's own session and permissions.

Which Aras Innovator versions are supported?

Edge AI is compatible with all Aras Innovator versions supported by Edge API. See the [Edge API documentation](#) for the supported version matrix.

Is Edge AI available on-premises?

Edge AI is delivered as a cloud service. Your Aras Innovator instance can remain on-premises or in a hybrid deployment, and Edge AI reaches it through the Edge API layer. You can also point Edge AI at your own model provider, including a self-hosted, OpenAI-compatible endpoint, so that inference runs on infrastructure you control. On-premises deployment of Edge AI itself is not part of version 2.0.

Do I need to install anything in Innovator?

Yes. The Aras Innovator AI Assistant Package must be installed on your Innovator instance to enable the embedded assistant panel. This package is not included in the Edge AI service and must be applied independently.

Does the embedded assistant consume a license?



Yes. Opening the assistant from the Innovator command bar consumes an Aras.AIAssistant license.

Roles and Access

What roles does Edge AI use?

Two. Edge AI User can chat with published agents and has no access to configuration. Edge AI Administrator has full control over agents, skills, tools, Access Points, and model configuration within the tenant. Where a user holds more than one role, access is merged.

What happens if a user has no Edge AI role?

They cannot reach the chat or the Agent Management Platform. They see an access denied message with instructions to contact an administrator. Assigning roles is a required onboarding step, not an optional one.

Is access controlled per agent or per resource?

Access is role-based at application level. Per-resource permissions, ownership-scoped access, and a separate agent builder role are not part of version 2.0.

Agents and Configuration

What is the Aras AI Assistant?

The default agent that ships with every tenant, pre-configured by Aras. An administrator connects it to an Access Point, selects the tools it should use, saves, and publishes it, after which end users can reach it. Administrators can extend it further with skills, MCP tools, and sub-agents, and can change the model it uses. Its name, agent type, solution instructions, and platform instructions are locked, and it cannot be deleted.

Can I change the model the Aras AI Assistant uses?

Yes. An Edge AI Administrator can assign a different LLM configuration and deployment. Once you override the model configuration, the tenant no longer receives Aras updates to it. Tenants that keep the default continue to receive them.

How is the Aras AI Assistant configured out of the box?



As a Deep Agent with High reasoning effort, with all Aras-managed skills selected and agent-to-agent delegation enabled with no sub-agents pre-selected.

Why can I not publish the Aras AI Assistant?

Publishing requires at least one Access Point in the tenant. Add one in the Agent Management Platform, then open the assistant, select that Access Point and the tools you want it to use, and save. You can then publish it.

Does adding an Access Point publish my agents?

No. Publishing is always explicit. An Access Point makes publishing possible, but it is not attached to any agent until you select it there, and no agent becomes available to end users until you publish it.

What is the difference between a template and an agent?

A template defines the core capability and behavior model. An agent is a configured instance of a template: it has a name, instructions, and specific Edge API connections bound to it. The Natural Language Analytics (NLA) template is the starting point for all structured PLM data querying agents.

Can I create additional agents beyond the Aras AI Assistant?

Yes. Administrators can create custom agents from templates. Each agent can have different solution instructions, tools, skills, and Edge API connections. Two agent types are available: Deep Agent and ReAct Agent.

Can I edit the skills and models that Aras provides?

No. Aras-managed skills, tools, and model configurations are shared to your tenant as read-only and are marked as Aras-managed in the interface. You can disable an Aras skill on your own agents, and a skill you disable stays disabled through later updates. You can also add your own skills alongside them.

How do I receive updates to Aras-managed content?

Automatically, at agent load time for skills, and by data migration for new tools, announced in the release notes. Only the latest version is available; there is no version pinning or rollback. An update never interrupts a conversation in progress, and the next conversation uses the updated



configuration. Agents you built yourself do not receive Aras updates, even if they started from the same template.

Can multiple agents share the same Edge API connection?

Yes. Edge API connections are tenant-scoped and reusable. A single connection can be linked to any number of agents.

Can I remove an Access Point that an agent is using?

No. An Access Point cannot be removed while it is attached to an agent. Repoint or reconfigure each agent that uses it first, then remove the Access Point.

Can I use my own AI model provider?

Yes. Administrators can register external model providers from the Models screen in the Agent Management Platform. Supported providers are Azure OpenAI and Microsoft AI Foundry, OpenAI, Anthropic, and Google. OpenAI-compatible endpoints, such as Ollama or vLLM, can also be registered using the OpenAI provider type with a custom endpoint.

What happens if I update an Edge API after assigning it to an agent?

Nothing is needed from you. Agents read the Edge API schema at runtime, so a newly exposed item type, property, or action is available on the next request.

Can I restrict what an agent is allowed to do?

Yes. The scope of data and actions available to an agent is determined by the Edge API connections bound to it and the governance policies defined in those APIs. Agents can only access data and perform operations explicitly exposed through their configured connections.

Can I test an agent before publishing it?

Yes. Use the Simulation Chat, accessible from the agent configuration or Agent Overview. It is visible only to administrators. Each response includes expandable reasoning steps showing what the agent queried and why. Only published agents are visible to end users.

What happens if an Edge API connection used by a published agent becomes unreachable?



Connections are exercised when they are used, so the agent stays published and the failure appears during a conversation: the affected tool returns a connection error and the assistant reports that it could not retrieve the data. Correct the endpoint or credentials on the Access Point in the Agent Management Platform. Where the Access Point uses API Key authentication, you can test the connection and see the result.

Data Changes

Can agents update, create, or delete data in Aras Innovator?

Yes. In version 2.0, agents can create records, update properties, change lifecycle state, and delete items. Every operation that changes data requires explicit human approval before it is executed.

Does the assistant ever change data without asking me?

No. Human approval is a platform invariant, not a configurable option. The agent can reason, plan, and propose freely, but it pauses before any operation that writes to Innovator and waits for explicit approval. There is no setting that enables automatic commit.

Where does the approval happen?

Inline in the conversation. The assistant presents an approval card in the chat stream showing the intended action, the target item, and the data payload, with approve and reject controls. There is no separate screen, queue, or application to switch to.

What happens if I reject a proposed change?

Nothing is written and the action ends cleanly. The conversation continues, so you can correct the instruction and let the assistant propose a revised action.

Does approval bypass my Innovator permissions or workflow rules?

No. Approval authorizes the attempt, not the outcome. The write is executed through Edge API under your own identity, so Innovator permissions, lifecycle rules, and workflow constraints still apply. If Innovator rejects the operation, no data is changed and the assistant reports the reason.

When the assistant plans several steps, do I approve each one?



Each operation that changes data is confirmed individually before it is committed. Read and analysis steps do not require approval. Review the full proposed plan before approving the first step, because approving one step does not authorize the rest.

Is there a record of what the assistant did?

Read operations are logged with item type, item ID, user, and timestamp. Changes committed through the assistant appear in Innovator as changes made by you, since the assistant acts under your identity and not a service account.

Data Access and Security**Can a user see data they are not authorized to see in Innovator?**

No. All queries execute under the user's own Innovator session via Edge API. The same permission model that governs access in Innovator applies to every query made through the assistant. This applies to search and relationship traversal as well: items you cannot see are excluded from Bill of Materials (BOM) and where-used results rather than hidden from the display.

Does the agent use a service account to reach my data?

No. Every Edge API call made by an agent carries the signed-in user's own authentication token. Attempts to reach items outside your permission scope return a governed error.

Is my data used to train AI models?

No. Aras does not use customer data, user prompts, inputs, or AI interaction data submitted through InnovatorEdge services to train, retrain, or improve any general-purpose or shared AI or machine learning models. See the Aras InnovatorEdge [Product Specific Terms](#) for full details.

How long does the assistant retain conversation data?

The assistant maintains short-term session context within an active conversation to support follow-up questions. This context is not retained after the session ends and is separate from configuration and audit records, which are persisted by design.

Are my agents and connections visible to other tenants?

No. Agents, Access Points, MCP servers, and model configurations you create are scoped to your tenant. Aras-managed resources are shared into your tenant as read-only and cannot be modified or removed by you.

How are credentials stored?

Credentials for Access Points and external MCP servers are held in a managed secret store rather than in the application database, and are never returned through the interface or the API once saved.

Using the Assistant

What types of questions can the assistant answer?

The assistant handles structured PLM data queries: listing and filtering items, BOM structure and where-used, change analysis, version and generation history, counts and aggregations, and status and ownership queries. It maps plain-language questions to the Innovator schema and retrieves results through configured Edge API connections.

Where can I use the assistant?

In two places, with the same capabilities in both. Embedded as a side panel inside Aras Innovator, which is resizable and can be opened alongside your work, and as a standalone chat reached from edge.aras.cloud/ai. Clicking **Full Screen** from the embedded panel opens the standalone view in a new browser tab.

Does the assistant know which item I have open in Innovator?

Yes. When the panel is open in Innovator, the item on the active tab is offered as context and shown as a chip above the input. You can remove the chip if you want to ask an unrelated question. If no item is open, the assistant works normally with no item context.

How do I get from an answer back to the data?

Every result links to its Innovator record. Clicking a link in the embedded panel navigates Innovator to that item.

What if the assistant misunderstands my question?



The assistant asks a short clarifying question when intent is ambiguous before executing a query. If the result is not what you expected, rephrase the question with more detail. Include the item type, property name, or lifecycle state you are interested in.

Can I refine results without starting a new query?

Yes. The assistant retains session context, so you can follow up with refinements such as "filter to items updated this week" or "sort by lifecycle state" without repeating the original question.

Can the assistant visualize results?

Yes. It can diagram the relationships between items and the structure of the data model, and it can produce a pie chart for a simple breakdown. Ask for the form you want in your question, or ask for it afterwards about a result already on screen.

How do I start over?

Click **New Chat**. This clears the session context, so the assistant no longer refers to anything from the previous conversation.

How do I switch to a different agent?

Use the agent switcher in the header. Switching agents starts a fresh conversation, and you are asked to confirm if the current conversation already has messages.

What are the suggested questions on an empty conversation?

Conversation starters, defined by your administrator. Selecting one places the text in the input so you can review and edit it before running it.

Can I stop a response mid-generation?

Yes. Click **Stop**. The assistant halts its current execution. You can then ask a new question or refine your previous one.

Can I export results from the assistant?

Yes. Tables can be exported to Excel and opened in the Innovator Search Grid. Charts can be exported as images.



Can the agent query items created or modified by a specific person?

Yes. The assistant correctly handles queries that filter items by the person who created, modified, owns, or manages a record. For example: "List all Parts created by Mike Miller" or "Parts modified by Jane Smith."

Can the agent show the history of a specific item?

Yes. Ask the agent to retrieve the change history of an item using natural language. For example: "Show me the history of Part MP0101." The agent returns a list of historical changes for that item.

Do AI-generated answers require human review?

Yes. AI-generated outputs must be reviewed and validated by a qualified person before use in any operational, engineering, manufacturing, compliance, or business decision. See the Aras InnovatorEdge Product Specific Terms for full details.

External Tools**Can agents use tools outside Aras Innovator?**

Yes. Edge AI acts as an MCP client, so an agent can be connected to an external MCP server and can discover and call the tools that server exposes.

Who controls which tools an agent can use?

Tools are granted per agent by an Edge AI Administrator. An agent can only call tools it has been granted, and changes to its tool grants take effect on its next run.

What authentication do external MCP servers support?

API key, or no authentication for a public server. An external server is reached with the credential configured on it rather than with the identity of the person using the agent. End-user identity is propagated to the Aras-managed tool server, which is where Innovator permissions apply.

Can one agent call another?

Yes. Agent-to-agent delegation lets an agent route part of a request to a specialist agent you nominate as a sub-agent. Where a sub-agent needs approval for a data change, that approval is



surfaced to you in your own conversation.

Limitations and Known Behaviors

Are all ItemTypes supported?

The assistant works against your Innovator schema without hardcoded item definitions, so in principle any ItemType exposed through Edge API can be queried. In practice, ItemTypes belonging to some Aras applications, including Variant Management, Technical Documentation, and Simulation Management, may not be fully queryable where Edge API schema exposure for those applications is incomplete. Validate coverage for these applications before relying on them.

Are there limits on how much data one answer can cover?

Yes. Retrieval is bounded per call and large aggregations may be truncated. When a result is capped or truncated, the assistant says so and asks you to narrow the request.

Why does the assistant sometimes report how many records it used?

For calculations over a set, the assistant states how many records were included, because records with incomplete data are excluded from the calculation. Treat that count as part of the answer.

Will the same question always give the same answer?

Within a conversation, identical questions against unchanged data return consistent results. Wording may vary between runs even when the underlying result does not, so compare the data rather than the phrasing.

What happens if no records match?

The assistant tells you that no matching items were found and states the filters it applied, so you can see whether it interpreted the question as you intended.

Does the assistant filter unsafe or inappropriate requests?

Yes. Content safety filtering is applied to conversations. A blocked message is not processed.

Regions and Data Residency



Where does Edge AI run?

Edge AI runs in the United States, Europe, and Japan. The service balances across those regions automatically, choosing where your tenant runs and shifting capacity as the service scales, so performance and availability stay steady without you managing infrastructure.

Can I have processing stay in a specific region?

Yes. Where your organization requires processing to remain within a specific region, that is configured for your tenant on request. Contact your Aras account representative.

Can I keep inference inside my own environment?

Yes. Register your own model provider and point it at an endpoint you choose. Prompts and responses are then processed within that environment rather than by the Aras-provided model.

Languages**Can I ask questions in a language other than English?**

Yes. The assistant understands the languages supported by the underlying model, and by default it answers in the language you asked in. You can also ask it to reply in a different language at any point in the conversation.

Can an administrator set the response language?

Yes. Response language can be steered through the agent's instructions, for example to standardize on one language across a team regardless of how the question was phrased.

Are PLM data values translated?

No. Item names, descriptions, and property values are returned as they are stored in Innovator. The assistant translates its own explanations and summaries, not your data.

Getting Help and Support**Where do I find the product documentation?**

Click **Documentation** in the help menu in the global header. This opens the Edge AI section of docs.aras.com in a new tab.

How do I contact support?

Click **Contact Support** in the help menu. This opens the Aras subscriber portal.

What details should I include in a support case?

The About panel shows the product name and release version, each individually copyable. The Profile menu shows your tenant name, with a copy action that places the tenant ID on your clipboard. Include the release version and tenant ID, along with the agent name and the question you asked.

Should I sanitize what I send to support?

Yes. A support case leaves your organization, so review transcripts, error messages, and screenshots before you attach them, and sanitize or anonymize anything covered by your intellectual property or confidentiality obligations. Real part numbers, program names, supplier names, and specifications are rarely needed to diagnose a problem. What support needs is the shape of the issue: the item type, the property, the action attempted, and the response received.



Edge AI Release Notes

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

2026 September 15 - Edge AI - Release Announcement - Edge AI version 2.0

Aras InnovatorEdge AI Foundational Release 2.0 Is Now Available

September 16, 2026 - Foundational Release

Every engineering team carries a standing backlog of questions its PLM system can already answer. Which parts a change affects. Who owns a document. How many Engineering Change Orders (ECOs) a team is still holding open.

The information is in Aras Innovator, but reaching it has meant knowing the schema, knowing where to look, or asking someone who does.

InnovatorEdge AI 2.0 removes that step, and then goes further.

The assistant no longer only reads Innovator data. It can act on it, under mandatory human approval, inside the workflow where the work is already happening.

- **Conversational PLM intelligence.** Answers from Innovator data in plain language, grounded in live records and scoped to the user's own permissions, with a link to the source record every time.
- **Governed action, not just answers.** The assistant can create records, update properties, change lifecycle state, and delete items in Innovator. Every change requires explicit human approval before it is committed. There is no automatic-commit setting.
- **AI where you work.** Embedded directly in the Innovator interface, aware of the record the user has open. A standalone application is available for work outside Innovator.
- **Enterprise administration.** Two consolidated roles under a single Centralized Identity Access Management (CIAM) application, a governed default assistant in every tenant, Skills and MCP Servers in the Agent Management Platform, and two agent reasoning types.



Conversational PLM Intelligence

Ask the question in plain language. Get the answer from your own data, with a reference to the record it came from.

How It Works

Answers are drawn directly from Innovator through Edge API, executed under the signed-in user's identity, so results are scoped to what that user is permitted to see. When a question is ambiguous, the assistant asks a short clarifying question before running anything. Follow-up questions refine the previous answer without restating context, so a multi-step investigation flows as a conversation rather than a series of new searches.

What It Covers

- Listing and filtering by state, date, owner, or attribute
- Bill of Materials (BOM) structure and where-used queries
- Change analysis across ECOs and change processes
- Version and generation history
- Counts and aggregations by type or classification
- Relationship and data model diagrams, and pie charts, drawn from the result set already returned

Transparency

Reasoning steps are visible inline with each response, so a user can see how an answer was reached and which records it came from, not only what the answer is. Administrators can configure conversation starters to guide first-time users toward the queries that matter most in their organization.

Governed Action: Approval Before Any Change

Version 2.0 is the release where the assistant stops being read-only. It can create records, update properties, change lifecycle state, and delete items in Aras Innovator. It cannot do any of that on its own.



Human approval is a platform invariant, not a configurable option. The agent reasons and proposes freely, then stops. It presents an approval card inline in the conversation showing the intended action, the target item, and the data payload, with approve and reject controls. No separate screen, no approval queue, no second application. Rejecting makes no change and ends the action cleanly. Approval authorizes the attempt, not the outcome. The write executes through Edge API under the user's own identity, so Innovator permissions, lifecycle rules, and workflow constraints all still apply. If Innovator refuses the operation, nothing is written and the assistant reports why. Changes committed through the assistant appear in Innovator as changes made by that user, not by a service account. This is what makes agentic PLM deployable in a governed engineering environment rather than only in a demonstration.

AI Where You Work

Instead of switching applications to find an answer, ask the question in a panel that sits inside Innovator, next to the work.

Embedded in Innovator

The assistant runs in a resizable side panel within Innovator, detects which record the user has open, and uses that record as context automatically.

An engineer reviewing a change order who needs to know who submitted it is one question away from the answer.

Context follows the user as they navigate between records, and can be cleared when the next question is unrelated.

Navigation

Every answer links back to the Innovator record it drew from. Selecting that link opens the record without ending the conversation. The loop stays continuous: question, answer, source record, next question.

More Room When You Need It

The assistant expands to a full-screen view with the session intact, and a standalone application is available at its own URL for work outside Innovator, with the same capabilities as the embedded panel.



Working in Your Own Language

Questions can be asked in any language the underlying model supports, and the assistant answers in the language the question was asked in by default. Administrators can steer response language through agent instructions where a team needs to standardize on one.

Open Tool Interoperability

Version 2.0 opens the assistant beyond Innovator data. Edge AI acts as an MCP client, so an agent can be connected to an external MCP server and can discover and call the tools that server exposes, bringing adjacent systems into the same conversation as PLM data.

Governance travels with it. Tools are granted per agent by an Edge AI Administrator, an agent can only call the tools it has been granted, and grant changes take effect on the agent's next run. Agent-to-agent delegation lets an agent route part of a request to a specialist agent, and where that specialist needs approval for a data change, the approval surfaces in the user's own conversation.

What Changed for Administrators

Role-Based Access

Version 2.0 consolidates the previous per-application access model into two defined roles managed through a single CIAM application: Edge AI User and Edge AI Administrator. A user with no assigned Edge AI role cannot reach the assistant or the Agent Management Platform, so role assignment is a required onboarding and upgrade step.

The Aras AI Assistant

A governed, pre-configured agent now ships with every tenant, delivered as a Deep Agent with high reasoning effort and all Aras-managed skills enabled. An administrator connects it to an Access Point, selects the tools it should use, and publishes it. Its name, agent type, and instructions are locked by Aras and it cannot be deleted. Administrators extend it by adding their own skills, MCP tools, and sub-agents, and can assign a different model where required.

The Agent Management Platform

The Agent Management Platform is where administrators manage agents, models, Access Points, and access.

Version 2.0 adds two areas:



- **Skills:** Reusable instruction bundles that teach agents domain-specific query patterns
- **MCP Servers:** The tool registry, where the Aras Innovator tools and any external tool servers an organization registers are managed together

Aras-managed skills, tools, and model configurations are shared into each tenant as read-only and are marked as Aras-managed, so administrators can see what they inherit and what they own.

Agent Types

Administrators can create agents using two reasoning approaches: the Deep Agent, which plans and executes and suits multi-step investigation, and the ReAct Agent, which reasons and acts iteratively.

Bring Your Own Model

Administrators can register external model providers from the Models screen: Azure OpenAI and Microsoft AI Foundry, OpenAI, Anthropic, and Google, along with OpenAI-compatible endpoints such as Ollama or vLLM through a custom endpoint. Inference can therefore run on infrastructure the customer controls.

Licensing

Edge AI User and Edge AI Administrator counts are defined by your organization's Edge AI subscription. See the [Administrator's Guide](#) for role definitions and the InnovatorEdge service terms for licensing details.

Availability

InnovatorEdge AI 2.0 is generally available from 15 September 2026 as a cloud service, hosted in the United States, Europe, and Japan.

Your Aras Innovator instance can remain on-premises or hybrid; the assistant reaches it through the Edge API layer. Edge API is a prerequisite, and the Aras Innovator AI Assistant Package must be installed on the Innovator instance to enable the embedded panel.

2026 August 11 - Edge AI - Incremental Release - Edge AI version 1.1.3

Aras InnovatorEdge AI Version 1.1.3 Is Now Available

August 11, 2026 - Incremental Release



Introducing product structure navigation in chat, configurable BOM depth, item version comparison, semantic search via AI Search, and European region model hosting.

What's new

Explore product structures and where-used results in the chat

Users can now ask the agent about the composition of an assembly or where a specific part is used, directly in chat, without opening the Structure Browser in Aras Innovator.

- Ask the agent to show the structure of an item and receive a formatted list of components with part number, description, quantity, and unit of measure at each level.
- Ask where a part is used and receive a list of parent assemblies, sorted so production-relevant assemblies (Released lifecycle state) appear first.

Results default to the latest released revision. Specifying a revision or lifecycle state in the question scopes the response accordingly.

Why this matters:

Product structure interrogation is one of the most frequent activities for engineers and change managers. Both BOM and where-used lookups are now available through natural language without a context switch to the Innovator Structure Browser.

Example scenarios:

- A user asks "Show me the BOM of Top Assembly X" and receives a formatted component list with quantities. They follow up with "Show where-used for Part 001" and receive a list of the assemblies it belongs to, sorted by lifecycle state.
- A user asks "Which assemblies use component MP0370-001 in Released state?" and receives a filtered where-used list showing only Released parent assemblies.



BOM and where-used depth now configurable up to 10 levels

Agents can now traverse product structures up to 10 levels deep. When asking about an assembly's structure or where a part is used, the agent navigates the hierarchy to the requested depth. If no depth is specified, the agent applies a default. If the user specifies a depth lower than 10, that value is honored.

Why this matters:

Multi-level assembly interrogation is a common engineering task. The depth cap ensures responses remain usable without overloading the conversation context.

Example scenario:

- A user asks "Show me the BOM of assembly X up to five levels deep" and receives a structure list scoped to five levels. A follow-up asking for level six returns results to that depth, up to the maximum of 10.

Retrieve item versions and compare changes

Users can now retrieve the full version history of an item and compare any two versions side by side through the agent, including non-consecutive versions.

Why this matters:

Tracking what changed between versions is a routine PLM task. It is now available through natural language without navigating to the item's version history in Innovator, and without being limited to comparing only adjacent versions.

Example scenario:

- A user asks "Compare Version 1 and Version 5 of part MP0101" and receives a side-by-side summary of the differences between those two versions.



Semantic search via AI Search

Agents can now answer questions through semantic meaning as well as structured property queries. When a question is better served by meaning than by exact property values, the agent queries the AI Search index and returns a ranked set of relevant items.

Why this matters:

Not every question maps neatly to a known item number or property value. Semantic search surfaces relevant items based on meaning, reducing the need to know the exact structure of the data before asking a question.

Example scenario:

- A user asks "Are there any parts associated with the hydraulics rework?" The agent queries the AI Search index semantically and returns a ranked set of relevant items, then retrieves authoritative detail for the top results.

European region hosting for Aras-provided models

Tenants with regional hosting requirements can now be assigned an Aras-provided model hosted in the European region. Regional model assignment is managed by Aras Customer Success and does not require action from tenant administrators.

Why this matters:

Provides a path to regional data residency compliance for European customers who require LLM inference to remain within the European region.

Example scenario:

- A European tenant's Aras-provided model is assigned to the European region by Aras Customer Success. Administrators see the regionally assigned model in their Models list and



can assign it to agents as they would any other Aras-provided model.

Bug fixes

Query count accuracy improved (APR-531, IH-20742)

Count queries now return accurate totals. Previously, certain query patterns returned counts that did not respect the applied filters. Additionally, the Edge API previously returned a count value even when the query filter referenced a non-existent property; invalid filter queries now correctly return an error.

User identity resolution improved (IH-21062)

Queries that filter by owner, creator, or assignment now correctly resolve the user's full display name and identity record. Previously, queries such as "List all parts created by Admin" or "What is assigned to me?" could fail to match because the agent used the login name rather than the full display name or identity record.

Note: This improvement requires the User and Identity item types to be configured in the Edge API with the required properties and relationships.

Non-consecutive version comparison now supported (IH-21307)

Users can now compare any two versions of an item, not only adjacent versions. Previously, comparing non-consecutive versions, such as Version 1 against Version 5, was not supported.



Write Enabled Badge No Longer Shows Stale Status (IH-20760)

The **Write Enabled** badge on a connection in Access Points previously persisted after write actions were removed from the connection, until a manual schema refresh was triggered. The badge now reflects the current configuration immediately.

Context mode get operations restored (IH-20752)

Agents using context mode can now correctly perform get operations. Previously these operations failed in context mode.

Chat session stability under high load (IH-20754)

Chat sessions no longer become unusable after the service encounters a high-load condition. Sessions now recover correctly.

Average-value queries use the correct field (IH-20755)

Average-value queries such as "What is the average duration of the Design Review activity?" now apply filters against the correct field. Previously these queries sometimes resolved against the Classification field instead of the Activity Name field.

2026 July 14 - Edge AI - Incremental Release - Edge AI version 1.1.2

July 14, 2026 - Incremental Release

Aras InnovatorEdge AI Version 1.1.2 is now available.

Introducing expanded query coverage for relational and historical data, and broader item type support.



What's new

Queries on complex relational properties now resolve consistently

The AI Assistant now correctly handles queries that filter items through properties that reference a related item type in Innovator. Previously these queries returned no results or failed silently. Identity-type properties (such as the person who created, modified, owns, or manages a record) are the most common example of this pattern.

Why this matters: A wide range of PLM queries depend on filtering through a relational property. These queries now work reliably across the item types and property types where they are configured.

Example scenario:

- A user asks "List all Parts created by Mike Miller" and receives a filtered table of results. The same works with variations such as "Parts owned by", "Parts modified by", or "Parts managed by" the same person, each resolved correctly through the Identity-type relationship.

The Agent Guides You When a Query Term Does Not Match

When the agent cannot find an exact match for a term used in a query, it now acknowledges the gap, surfaces the closest available result, and suggests how to refine the question, rather than returning a silent empty response.

Why this matters: Users no longer need to know the agent's exact vocabulary to get useful answers. Near-miss queries now produce a guided response instead of a dead end.

Example scenario:

- A user asks "Show me assembly parts marked as flagged." The agent cannot validate "flagged" as a data field, so it proceeds with what it can resolve: "Interpreted assembly parts as Parts whose Type contains assembly. I could not validate a metadata-backed field for flagged, so I



retrieved using only the assembly filter. There are 244 matching Parts; the first 25 are available in the current result set." The user receives results and knows exactly what was searched.

Query items related through complex type structures

The agent can now correctly retrieve related items in cases where the related item type is a Polyitem. Previously, relationship queries involving these item types returned incomplete or no results.

Why this matters: Product structures in Aras Innovator often include item types that can resolve to multiple concrete types. Agents can now navigate these structures accurately, broadening the range of relationship queries that return correct results.

Example scenario:

- A user asks "Show me all Parts managed by Jane Smith." The agent correctly resolves the "managed by" field as an Identity-type relationship property and returns the expected results rather than an empty or partial set.

Access item history through the agent

Users can now ask the agent to retrieve the change history of a specific item directly in the chat interface.

Why this matters: Item history is a routine PLM lookup: tracking when a part changed, who changed it, and what changed. It is now available without leaving the AI Assistant.

Example scenario:

- A user asks "Show me the history of Part MP0101" and receives a list of historical changes for that item, including dates and the nature of each change.



Date range and duration calculations now return accurate results

The agent now correctly calculates time spans and duration metrics involving Innovator date properties. Previously, queries requiring the agent to compute elapsed time or compare date values across properties returned inconsistent or no results. Workflow activity duration is a key example of this pattern.

Why this matters: Many operational PLM questions involve time: how long a step took, how overdue an item is, or the average cycle time for a process. These queries now produce reliable numeric results.

Example scenario:

- A user asks "What is the average number of days for the Design Review activity?" and receives a computed average duration based on the start and completion date properties available in the activity records.

Bug fixes

Connection Status badge truncated at 1024px viewport (APR-1029)

Fixed a layout issue where the Connection Status badge on the Agent Overview page was hidden or truncated when the browser window was resized to 1024px width.

2026 June 23 - Edge AI - Incremental Release - Edge AI version 1.1.1

Aras InnovatorEdge AI Version 1.1.1 Is Now Available

June 23, 2026 - Incremental Release

Introducing a redesigned agent configuration experience, centralized Access Point management, agent testing with step-by-step transparency, and a renamed Models section.



What's new

Redesigned agent configuration

Administrators who create and manage agents now work in a structured configuration editor. The editor keeps the configuration state visible at every step and saves the agent as a Draft. After saving, a read-only Agent Overview becomes available, providing a direct path back to editing.

Why it matters: Reduces errors during agent setup by making required fields explicit and the agent's current state clear throughout the process.

Example scenario:

- An admin creates a new agent, completes the configuration form, and saves. The agent is now in Draft, not published to end users, while the admin continues to refine and test it before publishing.

Agent testing with execution trace

Before publishing an agent, administrators can test it against live data in a full-page Simulation Chat, visible only to admins. Each response includes an expandable Steps trace showing exactly what the agent did, which data it queried and what decisions it made.

Why it matters: Gives administrators confidence in agent behavior before exposing it to end users, without requiring a separate test environment.

Example scenario:

- An admin opens the Simulation Chat for a newly configured agent, sends a test prompt, and expands the Steps trace to verify the agent is querying the correct dataset and returning results in the expected format, all before publishing.

Access Points - centralized connection management

Edge API connections are now managed from a dedicated Access Points screen. Admins can view



all registered connections in one place, add new connections, see the health status of each connection, and edit connection details after creation.

Why it matters: Gives administrators a single place to manage and monitor all data connections across their tenant, with visibility into connection health before assigning a connection to an agent.

Example scenario:

- An admin navigates to Access Points, sees that one connection is showing an error status, updates the credentials in the connection detail view, and verifies the connection is healthy before the agent that depends on it is republished.

Connection health monitoring on Agent Overview

If a previously working Edge API connection becomes unreachable, the Agent Overview flags the issue and blocks the agent from being republished until the connection is restored or removed.

Why it matters: Prevents an agent from being republished in a broken state, protecting end users from connection-related failures.

Example scenario:

- An API key expires overnight. The following morning, the Agent Overview shows the connection as unhealthy. The admin updates the key in Access Points and the agent can be republished.

Models - updated registry label

The model registry, previously labelled LLM Connections, is now labelled Models. Selectors within agent configuration now distinguish between Reasoning Models and Embedding Models, making model selection clearer for administrators.

Why it matters: The updated labels make it easier to identify and assign the right model type when



configuring an agent.

Permission level visible on Agent Library

Each agent on the Agent Library page now displays its permission level. Administrators can assess agent access at a glance and filter the list by permission level without opening individual agent records.

Why it matters: Reduces the steps required to audit agent access across a tenant.

Example scenario:

- A tenant admin performs a quarterly access review by filtering the Agent Library by permission level, identifying agents with broader access than expected, and adjusting their configuration accordingly.

Important notes

Data write actions not available in this release

Agents cannot create or edit data in Aras Innovator in this release. This capability is planned for a future release.

Bug fixes

Blank page after canceling agent creation (APR-872)

Fixed an issue where canceling the creation of a new agent displayed a blank page instead of returning the administrator to the Agent Library.

2026 May 12 - Edge AI - Incremental Release - Edge AI version 1.1.0

Edge AI Commercial Release 1.1.0 Is Now Available



May 12, 2026 - Incremental Release

Introducing conversational BOM and Where-Used queries, bring-your-own LLM support, agent-level model assignment, and multiregional deployment.

What's new

Query-on-the-fly: BOM and Where-Used

The Natural Language Analytics (NLA) Agent now executes live queries across item datasets for Bill of Materials (BOM) and Where-Used analysis. BOM results are rendered as indented tables directly in the chat interface.

Why it matters: Provides immediate visibility into assembly hierarchies and parent-child relationships without requiring precomputed reports or manual navigation to the Structure Browser.

Example scenario:

- A user asks "Show me the BOM of Assembly X" and receives an indented table showing the first two levels of the hierarchy, including part numbers and quantities.

Bring your own LLM (BYOLLM)

Admins can register and manage external model providers through a centralized LLM Connections interface.

Why it matters: Enables the use of existing model subscriptions.

Example scenario:



- A Tenant Admin configures a specific Azure OpenAI endpoint to ensure agent traffic remains within a designated regional boundary.

Agent-level model assignment

Admins can assign a specific registered LLM to an individual agent via a governed catalog dropdown in the agent configuration menu. A global default model can also be set at the tenant level, ensuring every new agent is preconfigured with an approved model.

Why it matters: Every agent might have different needs in terms of the model to be used.

Example scenario:

- An admin designates either a high-capacity or a speed-optimized model as the tenant default so that any new agent is automatically pre-selected with that model.

Multiregional deployment

The Edge AI Platform is now available for distribution across US and EU regions. All Edge services prominently display the active organization or tenant name in the user menu.

Why it matters: Improves service performance across geographical regions.

Bug fixes

Technical Documentation application support [APR-473]

Resolved an issue where the AI Agent was unable to process requests or retrieve structured content associated with Technical Documentation application ItemTypes.

2026 April 9 - Edge AI - Incremental Release - Edge AI version 1.0.1



Aras InnovatorEdge AI Version 1.0.1 Is Now Available

April 9, 2026 - Incremental Release

What's new

Agent lifecycle management

Agents now move through formal **Draft**, > **Published**, and **Unpublished** states. Administrators see real-time status indicators in both the agent list and detail view. Published agents are locked for editing until explicitly unpublished. Only published agents are visible to end users in the agent selection list.

Why it matters: Gives administrators full control over what end users can access:

- Agents under construction stay hidden until they are ready
- Live agents cannot be accidentally half-edited

Example scenarios:

- Create and test a new agent privately in Draft before ever exposing it to users via the preview chat.
- Unpublish a live agent to safely revise its system instructions, then republish. Users see no in-between state.

API connection timestamps

The administration interface now shows **Last Modified** and **Updated On** timestamps for every configured API connection, so you can tell when the cached schema was last synchronized with Aras Innovator.



Why it matters: Eliminates guesswork around stale metadata.

You can now check whether a schema refresh is needed, rather than discovering a discrepancy after an agent misbehaves.

Example scenario:

- After adding a new property to an ItemType in Innovator, check the timestamp to confirm the Edge AI platform has picked up the change. If not, trigger a manual refresh via a connection test or description update.

Bug Fixes

Inconsistent chart output for date-based queries (APR-472)

Date-driven queries were producing incorrect or inconsistent chart visualizations. Resolved with stricter chart generation prompts, improved data integrity guardrails, and better OData query validation.

Help button missing in standalone mode (APR-603)

The Help button was not appearing in the Edge AI chat header when running in standalone mode. Restored to the expected location in the application header.

2026 April 2 - Edge AI - Release Announcement - Edge AI version 1.0.0

Aras InnovatorEdge AI foundational release 1.0.0 Is Now Available

April 2, 2026 - Foundational Release

Overview

This release announces general availability of **Aras InnovatorEdge AI**, the AI intelligence layer for the Aras Innovator digital thread.

InnovatorEdge AI activates the intelligence already embedded in your Aras Innovator system of record, making product lifecycle data accessible to every user through natural language - governed



by the same permissions, policies, and data model your organization already owns and controls.

What Is Included in This Release

Natural Language Analytics (NLA) Agent

The first production agent in the InnovatorEdge AI ecosystem enables any Innovator user, regardless of PLM expertise, to query structured product data through plain language conversation. Engineers, managers, and executives can ask questions about parts, BOMs, engineering changes, lifecycle states, and project status, and receive results as tables, charts, and Key Performance Indicators (KPIs) with direct links back to source records in Innovator. No AML knowledge. No filter configuration. No administrator dependency.

Aras Innovator AI Assistant

The NLA Agent is delivered through the **AI Assistant**. It is available as an embedded panel within Aras Innovator and as a standalone Edge AI application. The embedded panel is context-aware: it automatically detects the item currently being viewed and scopes queries accordingly.

InnovatorEdge AI Platform

This release establishes the foundational platform on which all current and future Edge AI agents operate:

- **Tenant isolation:** secure separation of agent configurations, API connections, and session data across all tenants



- **Identity propagation:** full CIAM integration ensures every agent interaction respects the individual user's Innovator permissions; no data governance bypass is possible
- **Edge API binding:** all data access flows exclusively through the governed Edge API layer; agents cannot access data outside their authorized scope
- **Agent configuration framework:** administrators configure agents through a no-code UI; instructions, data connections, and starting questions are all configurable without engineering involvement

Why This Matters

PLM data is often the most interconnected product knowledge in the enterprise. The barrier has not been the data it has been an abundance of data often siloed across the same Innovator account.

InnovatorEdge AI removes that barrier by providing a secure gateway to that information, broadening access while keeping every interaction within existing enterprise governance and user authorization. Every query executes under the user's own Innovator session. Every result is drawn from authorized data. Every interaction is ephemeral; no customer data is used to train or improve AI models.

Deployment

InnovatorEdge AI is available for Full Software as a Service (SaaS) and Hybrid deployments. The Aras Innovator instance can remain on-premises; Edge AI connects through the Edge API layer. This release requires an active Aras Innovator subscription and an active InnovatorEdge API subscription.

What Is Coming Next

The NLA Agent is the first in a broader ecosystem. Subsequent releases will introduce:



- **InnovatorEdge AI Task Agents:** Purpose-built agents for Requirements Ingestion, BOM Import, and Smart Variants workflows
 - **Knowledge Base Discovery Agent:** RAG-powered semantic retrieval over structured and unstructured PLM content
 - **Semantic AI Search:** context-aware, relevance-ranked search across all Innovator items
-



Task Agents



Aras InnovatorEdge AI Agents Overview

Aras InnovatorEdge AI introduces a suite of **specialized agents** designed to operate directly on the digital thread.

These agents are not disconnected AI tools; they are governed, identity-aware services that activate the intelligence already embedded within your Aras Innovator system of record.

What are Aras InnovatorEdge AI Agents?

Aras InnovatorEdge AI Agents are specialized AI services built to perform specific analytical or task-oriented functions within the product lifecycle.

They are categorized into two primary groups based on their functional purpose:

1. Conversational Agents (Discovery)

Embedded within the Aras AI Assistant, these agents help users navigate, analyze, and reason across complex product data using natural language.

- **Analytics & Insights Agent:** Provides natural language analytics over structured product data to speed time to insight.
- **Knowledge Analysis Agent:** Utilizes semantic reasoning across documents and digital thread relationships to uncover historical design intent.

2. Task Assistant Agents (Enrichment)

These agents focus on automating high-friction engineering workflows by creating structured relationships and normalizing data.

- **Requirements Ingestion Agent:** Transforms unstructured PDF documents into governed, structured Requirement Items.
- **BOM Assistant Agent:** Normalizes incoming supplier BOM data into standard Aras structures.
- **Smart Variant Agent:** Analyzes product documentation to assist in the creation of variant models and configurations.

Why This Matters?



The introduction of AI agents built directly on the digital thread represents a strategic shift from traditional "bolt-on" AI approaches.

Activating the Existing Foundation

For organizations invested in Aras Innovator, the foundation for scalable AI is already laid. These agents do not require you to start from scratch; they activate the intelligence in the digital thread you already own.

By grounding AI in your system of record, Aras ensures that insights are derived from authoritative, up-to-date data.

Governed and Secure Intelligence

Enterprise AI requires more than just performance - it requires governance.

- **Traceability:** Every action and outcome produced by an agent is recorded as part of the product record, maintaining a traceable chain of evidence for compliance and auditing.
- **IP Protection:** Every API call is contextualized and identity-aware, ensuring that agents only access data the user is authorized to see.
- **Model Sovereignty:** Aras does not use customer digital thread data to train foundation LLMs. All data interactions occur within the customer's governed tenant boundaries.

Solving High-Friction Business Pains

By bridging the gap between unstructured data (like PDFs and spreadsheets) and structured PLM objects, AI agents reduce the dependency on specialized AML/SQL experts. This shifts human effort away from manual data entry and transcription toward high-value validation and decision-making, significantly accelerating the path from "engineered-to-order" to "configure-to-order."



Requirements Ingestion Agent - User Guide

Overview

The **Requirements Ingestion Agent** is a **Task Assistant Agent** within the **InnovatorEdge AI** framework.

It enriches the digital thread by transforming requirements contained in unstructured PDF documents into structured, governed Items within **Aras Innovator Requirements Engineering (RE)**.

What the Requirements Ingestion Agent Does

The agent operates directly inside the system of record to normalize incoming document data. Its purpose is to:

- **Extract** requirement statements from PDF documents.
- **Separate** engineering content from document presentation and formatting.
- **Convert** extracted content into governed **Requirement Items** under a **Requirements Document Item**.
- **Reduce** manual transcription effort for large technical specifications.

Important

The agent is user-invoked and human-validated. No data is committed to Aras Innovator until the user confirms the import summary.

Preparing the Document for Import

The quality of extraction depends on the structural conventions of the source PDF file.

The agent analyzes structural patterns, including:

- **Requirement Boundaries:** Whether requirements are clearly separated by numbering and/or headings.
- **Directive Phrases:** Presence of action-oriented statements (e.g., "Add," "Replace") indicating modification actions.



- **Integrated Content:** Identifying whether lists, tables, or images are part of requirement statements.
- **Document Commentary:** Distinguishing between core requirements and notes, justifications, or general commentary.

Constraints:

- **Format:** Supports a single **.pdf** file.
- **Size:** Maximum file size is **10 MB**.

Requirements Import

1 Upload 2 Processing 3 Review 4 Import [Start Import](#)

Upload Requirements

Requirements Document
Choose a PDF file containing requirement specifications.

Upload file

Drag and drop a file here or click to upload

Max size: 10MB.

Requirements.pdf 0.65 MB X

Extraction Instructions (Optional)
Provide additional instructions for the Requirements Extraction Agent to get the best results from your specific requirements document format.

Custom parsing instructions

0 / 10000

By using this agent, you agree to [Aras Legal Terms](#) and [Privacy Policies](#).

Aras Innovator Instance: edge-ai-agents-demo

Agent Instructions - Defining Ingestion Rules

Agent Instructions configure how the AI interprets the document during the ingestion phase. These apply globally across the document and shape how boundaries and titles are determined.

Examples:

- **Section-Based Rules:** "Start a new requirement at each hierarchical section number such as 6.1.1. Use the section number as the Requirement Title."



- **Action-Based Rules:** "Create a separate requirement for each action sentence that begins with Add, Replace, or Delete."
- **Exclusion Logic:** Instructions can specify what content to ignore, such as "Do not treat Notes as requirements".

What the Agent Extracts

During parsing and segmentation, the agent identifies and segments the following content types:

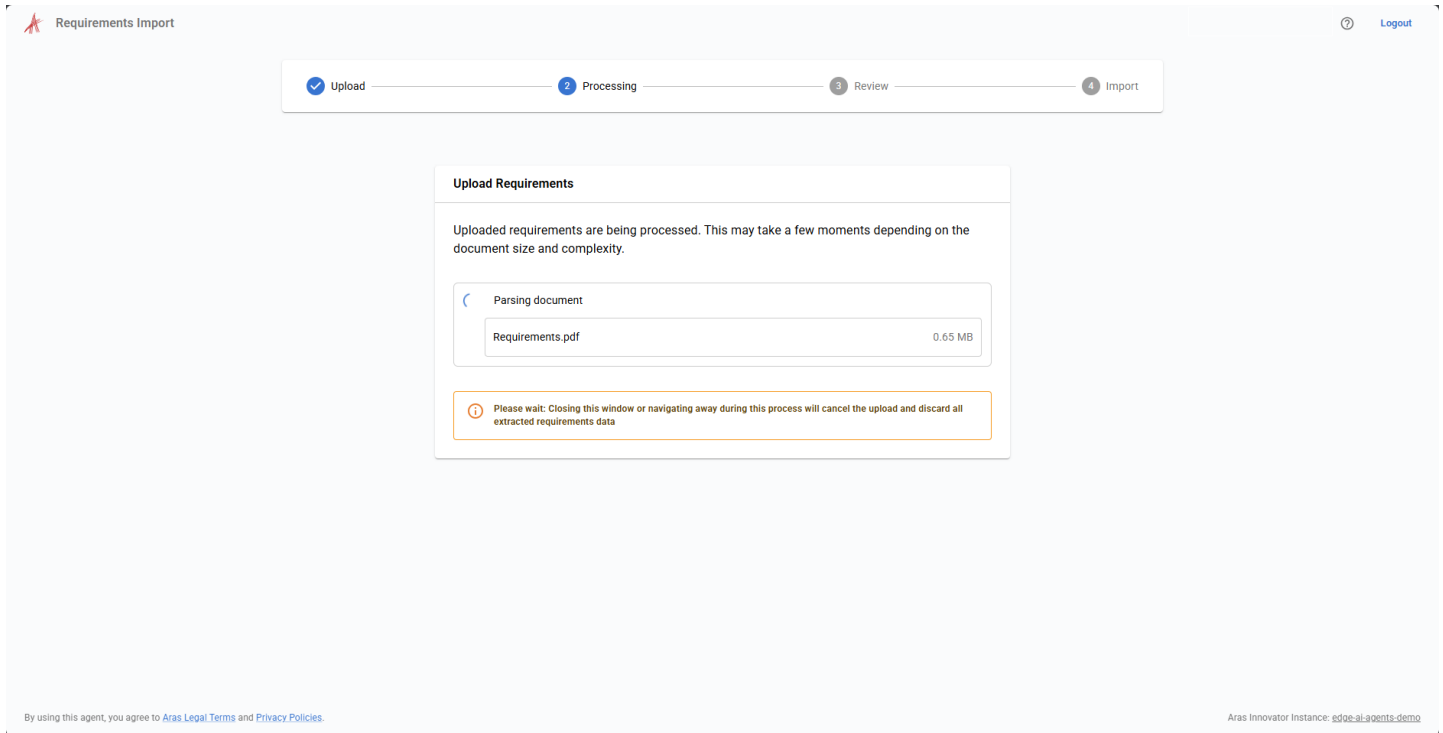
- **Text:** Paragraph-based statements extracted as structured text, which is fully editable during review.
- **Tables:** Tabular content associated with a requirement is preserved. Table content is not editable during review but can be adjusted later in Aras Innovator.
- **Graphics:** Images and diagrams are identified and prepared for import as **Requirement Item** content.

The Processing Phase

Once the import is initiated, the agent enters a multi-stage technical extraction process:

1. **Parsing:** Analyzing structural elements like headings, numbering schemes, and paragraph blocks.
2. **Content Detection:** Identifying specific segments that contain requirement-relevant data.
3. **Formalization:** Constructing candidate Requirement entities containing derived titles, text, and related element references.





Important

Closing the browser window or navigating away during the processing phase will cancel the upload and discard all extracted data.

Error Handling during Processing

If the agent encounters complex formatting or connection issues it cannot resolve, it will report an **Import failed** status.

In the event of a failure, the specific cause is displayed (e.g., "Failed to classify page n").

Selecting **Try again** returns you to the Upload screen to attempt ingestion again with adjusted instructions.

Reviewing Requirements

After processing is complete, the agent presents the **Review Requirements** interface. This phase allows engineers to apply domain expertise to the AI's interpretation, ensuring the structured model is accurate before it is committed to Aras Innovator.



The Review Interface Layout

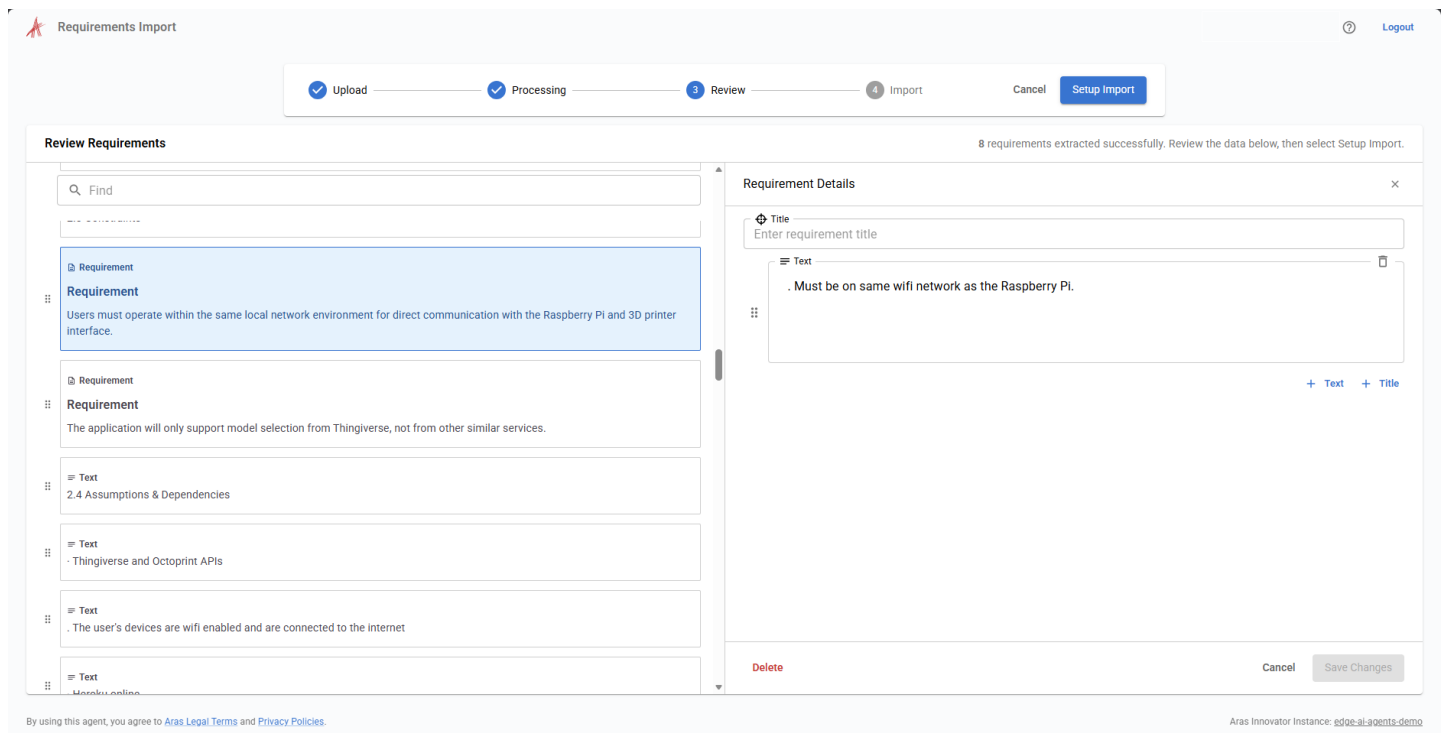
The interface uses a split-view layout designed for high-volume auditing:

- **Navigation Pane (Left):** A vertical, scrollable list of all extracted items.
- **Persistent Search:** A **Find** bar remains docked at the top of the navigation pane, allowing you to filter items by keyword even while scrolling through long documents.
- **Element Details (Right):** A contextual pane that displays the full content of the item currently selected in the navigation list.

Item Classification and Organization

Each extracted item in the list is explicitly labeled with its identified type and a corresponding icon to help users quickly distinguish between different content blocks:

- **Requirement (Bullseye icon):** Formal engineering requirements.
- **Text (Three-line icon):** Standard paragraph text or section headers.
- **Table (Grid icon):** Data structured in rows and columns.
- **Graphic (Picture icon):** Visual elements, which the agent identifies as either photos/pictures or UML Diagrams.

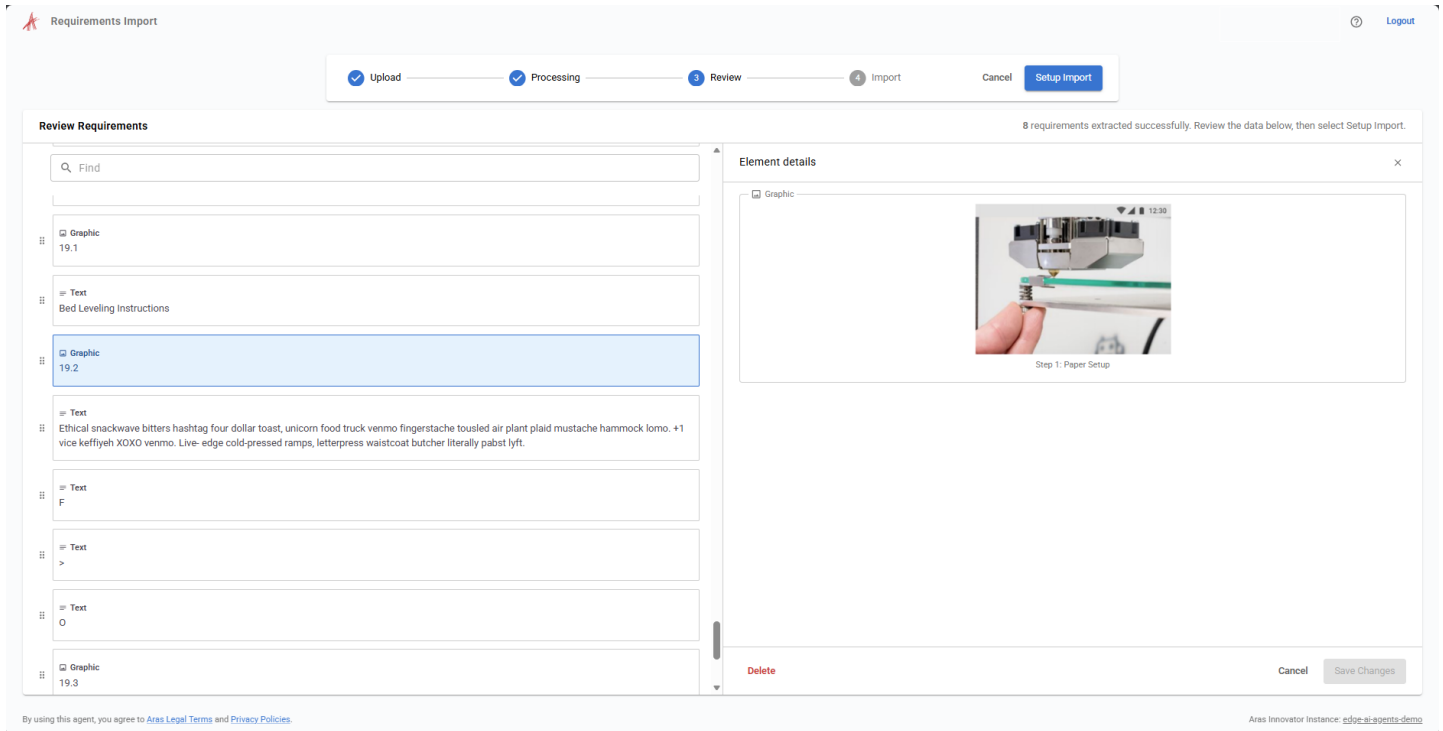


Human-in-the-Loop Actions

The interface provides several tools for refining the extracted data model:

- **Drag-and-Drop Reorganization:** Each item in the navigation pane features a handle icon on the far left, allowing you to drag and drop items to reorder them or change the document hierarchy.
- **Inline Editing:** Selecting a **Requirement** or **Text** item opens an editable view in the Element Details pane. You can manually update **Requirement Titles** and body text to correct any AI parsing errors.
- **Adding Content:** Within the details pane for Text elements, you can select **+ Title** or **+ Text** to add new headers or paragraphs that may have been missing from the initial extraction.
- **Deleting Elements:** Any irrelevant content—such as headers, footer text, or discarded notes—can be permanently removed by selecting the **Delete** option at the bottom of the details pane.
- **Inspecting Graphics and Tables:**
 - **Graphics:** You can view high-resolution previews of extracted images and diagrams to ensure they are associated with the correct requirement.
 - **Tables:** The details pane renders the extracted grid for visual verification. While table content is not editable in this view, the structure is preserved for the final import.





Important

Changes made in the detail view must be saved within the preview to be included in the final import.

Importing into Aras Innovator

Once validation is complete, you provide a **Requirement Document Title** and initiate the formal import sequence.

Item Creation and Governance

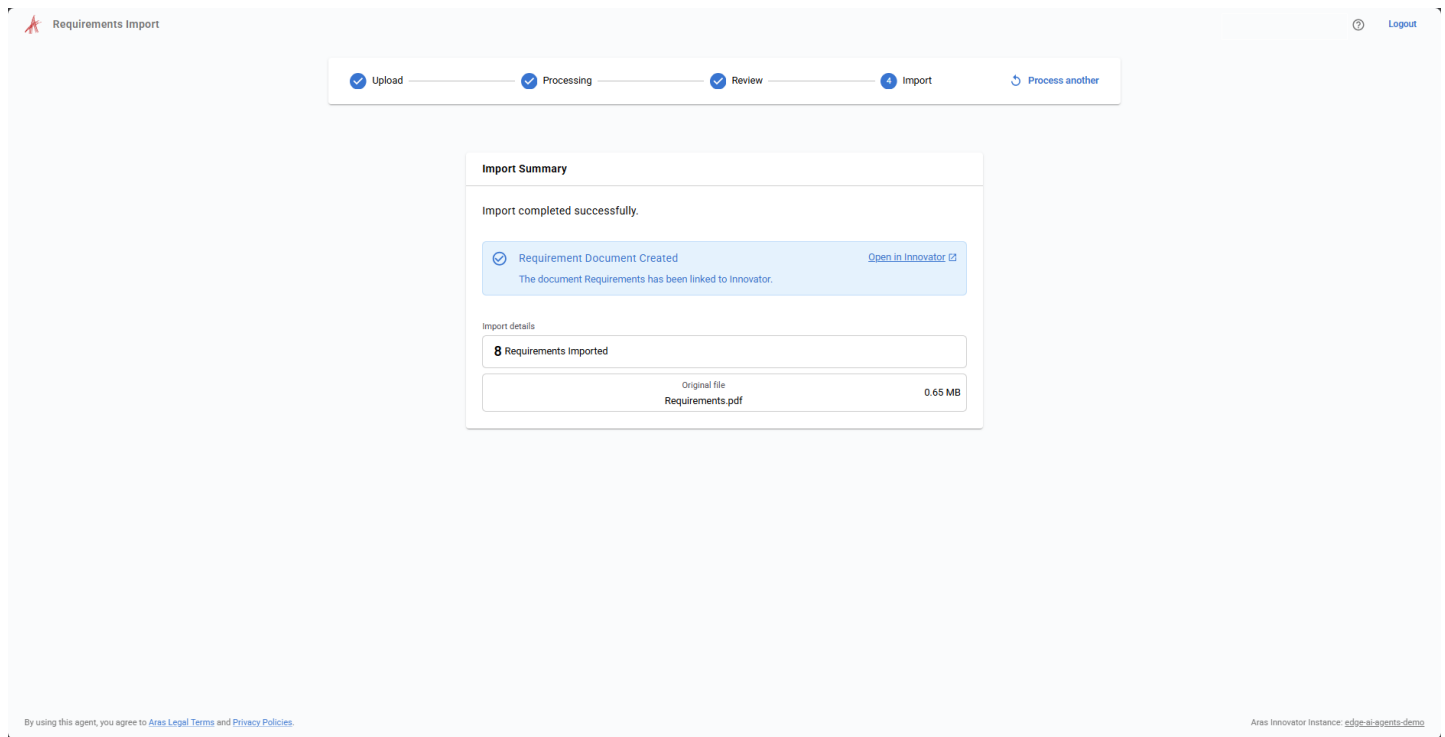
The agent creates the following native objects:

1. **Requirements Document Item:** Exactly one parent container is created per session.
2. **Requirement Items:** Individual items are generated for every validated extraction.
3. **Relationship Binding:** Structured links connect the document to its requirements in the correct sequence.
4. **Graphic Commitment:** Images and diagrams are uploaded as **File Items** and associated with the corresponding records.



Verification in Aras Innovator

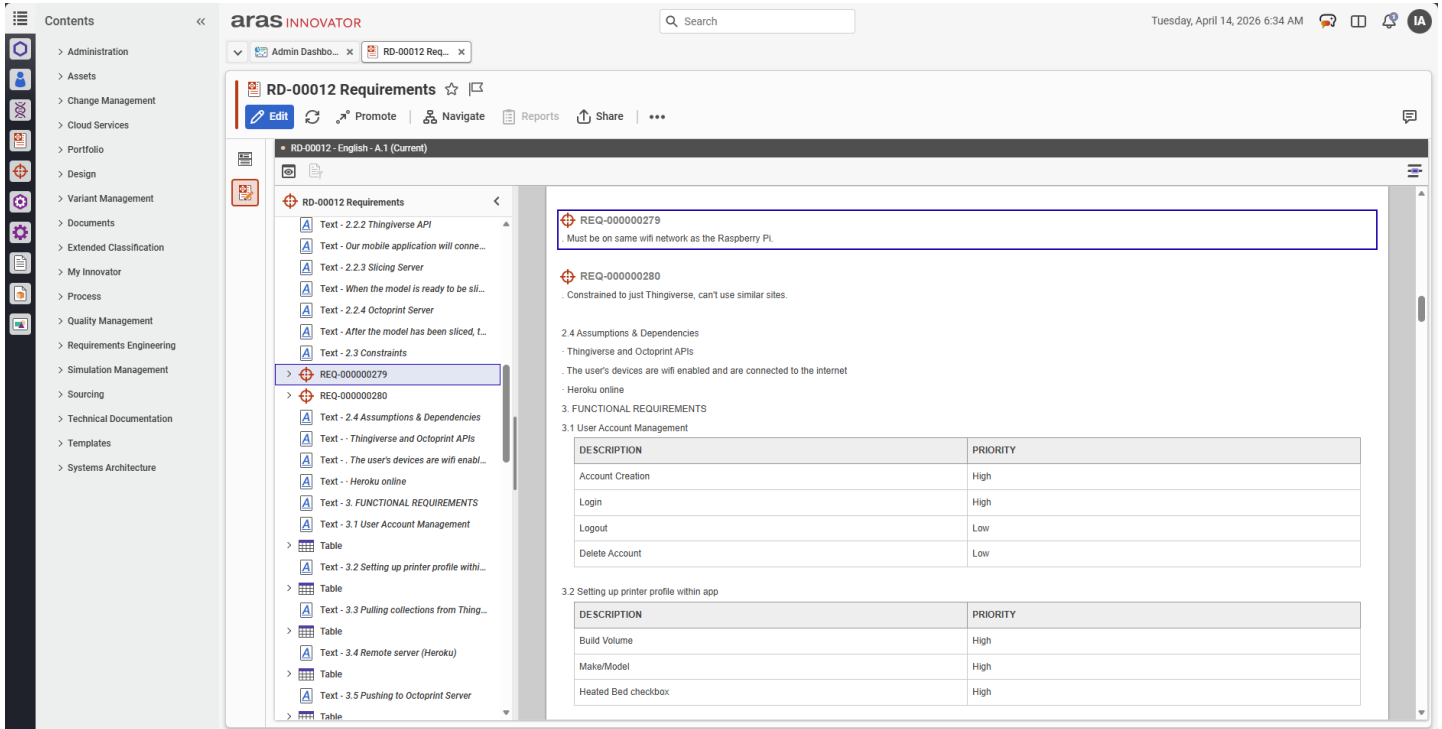
Upon successful completion, an **Import Summary** is displayed.



The screenshot displays the 'Requirements Import' interface in Aras Innovator. At the top, a progress bar shows four steps: 'Upload' (completed), 'Processing' (completed), 'Review' (completed), and 'Import' (current step, indicated by a blue circle with a white 'A'). A 'Process another' button is located to the right of the progress bar. Below the progress bar, the 'Import Summary' section is visible. It contains the text 'Import completed successfully.' followed by a blue box with a checkmark icon and the text 'Requirement Document Created'. Below this, it states 'The document Requirements has been linked to Innovator.' and provides a link 'Open in Innovator'. Underneath, the 'Import details' section shows '8 Requirements Imported'. A table below this lists the imported files: 'Original file' and 'Requirements.pdf' with a size of '0.65 MB'. At the bottom left, there is a small text: 'By using this agent, you agree to [Aras Legal Terms](#) and [Privacy Policies](#).' At the bottom right, it says 'Aras Innovator Instance: edg-ai-agents-demo'.

Selecting **Open in Innovator** opens the standard interface, where the document and requirements participate in standard governance, including lifecycle management and versioning.





The screenshot shows the Aras InnovatorEdge AI interface for a Requirements Engineering project. The left sidebar contains a 'Contents' menu with categories like Administration, Assets, Change Management, Cloud Services, Portfolio, Design, Variant Management, Documents, Extended Classification, My Innovator, Process, Quality Management, Requirements Engineering, Simulation Management, Sourcing, Technical Documentation, Templates, and Systems Architecture. The main workspace displays 'RD-00012 Requirements' with a tree view on the left and a detailed view on the right. The tree view shows a hierarchy of requirements, including 'REQ-000000279' and 'REQ-000000280'. The detailed view shows the content of 'REQ-000000279', which includes a description, assumptions, and dependencies. The detailed view also includes two tables: '3.1 User Account Management' and '3.2 Setting up printer profile within app'.

DESCRIPTION	PRIORITY
Account Creation	High
Login	High
Logout	Low
Delete Account	Low

DESCRIPTION	PRIORITY
Build Volume	High
MakeModel	High
Heated Bed checkbox	High

Guidelines & Best Practices

- **Human-in-the-Loop:** All AI-generated extractions must be reviewed and confirmed by an engineer before system commitment.
- **Structural Consistency:** Extraction quality is highest when documents follow consistent section and numbering conventions.
- **Permission Awareness:** The import process respects your native Aras Innovator permissions and assigned roles.
- **Error Handling:** If an ingestion job fails (e.g., a classification failure), the system provides a specific error message and a **Try again** option to restart the process.



BOM Import Agent - User Guide

Overview

The **BOM Import Agent** is a **Task Assistant Agent** within the **InnovatorEdge AI** framework. It enriches the digital thread by transforming Bill of Materials (BOM) data contained in engineering spreadsheets into structured, governed Items within **Aras Product Engineering (PE)**.

What the BOM Import Agent Does

The agent operates directly inside the system of record to normalize incoming supplier data. Its purpose is to:

- Extract part metadata and hierarchical relationships from Excel files.
- Convert unstructured or semi-structured spreadsheet rows into governed **Part** and **Part BOM** Items.
- Reduce manual transcription effort for large, multi-level structures.
- Establish traceability by linking imported items to the authoritative digital thread.

The agent is user-invoked and human-validated. No data is committed to Aras Innovator until the user confirms the import summary.

Preparing the Document for Import

The quality of extraction depends on the structural conventions of the source Excel file. The agent analyzes:

- **Hierarchy Indicators:** Single-column leveling (1, 1.1) or stair-step indentation across columns.
- **Property Labels:** Header rows defining Part Numbers, Names, and Quantities.
- **Metadata Consistency:** Use of consistent units of measure and procurement types.

Constraints:

- **Format:** Supports a single **.xlsx** file.
- **Size:** Maximum file size is **10 MB**.
- **Structure:** The file must define exactly one **Root (Level-0) Part**.



 BOM Import
 ? [Logout](#)

1 Upload Document

2 Processing

3 Review BOM

4 Import

Start Import

Upload BOM

BOM Document

Choose a file containing BOM data.

Max size: 10MB. Supported file types: .xlsx, .xls.



Drag and drop a file here or click to upload

Advanced Camera Drone Bill of Materials no dashes

0.02 MB

×

Extraction Instructions (Optional)

Provide additional instructions for the Requirements Extraction Agent to get the best results from your specific requirements document format.

Custom parsing instructions

0 / 10000

By using this agent, you agree to [Aras Legal Terms](#) and [Privacy Policies](#).

BOM Agent Instructions - Defining Mapping Rules

Agent Instructions configure how the AI interprets specific spreadsheet columns. These apply globally across the document during the ingestion phase.

Users can provide natural language rules to align source data with the Aras Data Model:

- **Mapping Example:** *"Map 'Assembly Identification' column values to 'Name' Part property;"*
- **Mapping Example:** *"Map 'Actual' column values both to Part and BOM 'Status' properties."*

What the Agent Extracts

During extraction, the agent identifies and segments the following content:

- **Part Metadata:** Unique identifiers (item_number), names, and descriptions.
- **BOM Relationships:** Parent-child links (source_id to related_id), quantities, and find numbers.
- **Graphics:** Embedded images are identified and prepared for import as **Part Thumbnails**.

 BOM Import
? [Logout](#)

1 Upload Document
2 Processing
3 Review BOM
4 Import

Start Import

Upload BOM

Uploaded BOM is being processed. This may take a few moments depending on the document size and complexity.

Uploading...

Advanced Camera Drone Bill of Materials no dashes

0.02 MB

! Please wait: Closing this window or navigating away during this process will cancel the upload and discard all extracted BOM data

By using this agent, you agree to [Aras Legal Terms](#) and [Privacy Policies](#).

Validating Extracted BOM Structures

After extraction, the agent presents a structured **Tree Preview** for review. This phase allows engineers to apply domain expertise before system commitment:

- **Hierarchical Search:** Filter and highlight text across thousands of rows to verify critical items.
- **Inline Editing:** Modify editable properties (Description, Quantity) directly in the grid.
- **Visual Review:** Verify extracted thumbnails against part descriptions.

 BOM Import
 Logout

Upload Document

Processing

Review BOM

Import

Import to innovator

New Import

Search

TYPE	DESCRIPTION	PART NUMBER	NAME
▼ -	Master Top-Level Assembly	100-0000	Pro Camera Drone
▼ -	Main Structural Frame	200-0100	Airframe Assembly
-	3mm Carbon Fiber Base	300-0110	Carbon Chassis
▼ -	Pre-built Motor & Arm	300-0120	Motor Arm Assy
-	15mm Carbon Fiber Tube	400-0121	Arm Tube
▼ -	2207 Stator Motor	400-0122	Brushless Motor Assy
-	Laminated Steel Core	500-0123	Motor Stator
-	24AWG Enamel Wire	500-0124	Copper Winding Coil
-	NS2 Curved Magnets	500-0125	Neodymium Magnets
-	ABEC-7 Steel Bearing	500-0126	Motor Bearings
▼ -	Electronic Speed Control	400-0127	ESC PCBA
-	4-Layer 2oz Copper	500-0200	ESC Bare Board
-	N-Channel 40V 100A	500-0201	Power MOSFETs
-	32-bit Motor MCU	500-0202	ESC Controller IC
-	470uF 35V Electrolytic	500-0203	Power Capacitors
-	1k Ohm 1% 0402	500-0204	SMD Resistors
▼ -	Central Processing Unit	200-0200	Main Avionics Module
-	Aluminum Heatsink Case	300-0210	Avionics Housing
▼ -	Main Drone Motherboard	300-0220	Flight Controller PCBA
-	6-Layer High-Density	400-0221	FC Bare Board

PART PROPERTIES

Id

88CBA61DE1F643D4A123F20725DC9359

Type

Cost

Description

Main Structural Frame

Part Number

200-0100

Keyed Name

Name

Airframe Assembly

Thumbnail

-

Managed By Id

BOM RELATIONSHIP

Id

2500D39E252941809A5D8974EEC82FB9

Quantity

1

Source Id

2A553F8DFBC7427293C686FB376CACB0

Related Id

88CBA61DE1F643D4A123F20725DC9359

By using this agent, you agree to [Aras Legal Terms](#) and [Privacy Policies](#).

Importing into Aras Innovator

Once validation is complete, the agent executes an atomic import sequence:

- 1. Part Creation:** Creates all new Part items identified in the structure.
- 2. Relationship Binding:** Establishes the Part BOM links using the verified hierarchy.
- 3. Thumbnail Upload:** Attaches extracted graphics to Part records.

Important

The agent is optimized to **Create New Parts** only. It does not automatically update existing parts in the system. Upon success, a direct link to the **Root Part** is displayed.


Importing to Innovator
?
[Logout](#)

✓ Upload Document

✓ Processing

✓ Review BOM

4 Import

Uploading to Innovator

Your BOM is being imported to Aras Innovator. This may take a few moments.

Submitting 49 part(s) in a single request...

Advanced Camera Drone Bill of Materials no dashes

49 Parts and 48 Part_BOM relationships to import

ⓘ

Please wait: Closing this window or navigating away during this process will cancel the upload and discard all extracted BOM data.

By using this agent, you agree to [Aras Legal Terms](#) and [Privacy Policies](#).

©2026 Aras Corporation All Copyrights Reserved

95

The screenshot displays the Aras Innovator user interface. On the left is a navigation menu with options like Administration, Change Management, Design, Documents, Extended Classification, My Innovator, Portfolio, Sourcing, and Variant Management. The main area shows the 'Part' form for part number 100-0000. The form includes fields for Name (Pro Camera Drone), Type, Unit (EA), Make / Buy (Make), Cost, Effective Date, and Long Description (Master Top-Level Assembly). Below the form is a tabbed interface with 'BOM Structure' selected. This tab shows a hierarchical Bill of Materials (BOM) structure with columns for Part Number, Revision, State, Sequence, Quantity, Claimed By, Name, Effectivity, and Reference De... The BOM structure lists various components like Camera Gimbal Assy, Gimbal Frame, Pitch Motor, Roll Motor, Yaw Motor, Camera Module, Lens Assembly, Image Sensor PCBA, Sensor Bare PCB, CMOS Sensor, Image Processor IC, Passives (R, C, L), Battery Pack Assy, Li-Po Cell, and BMS PCBA, each with associated revision, state, sequence, and quantity information.

Clicking the Root Part link opens the standard Aras Innovator interface. Here, you can see the **Part Form** populated with the name, description, and thumbnail extracted by the agent.

Navigate to the **BOM Structure** tab on the Part record to view the multi-level hierarchy as it exists in the system of record. Every link, quantity, and sequence created by the agent is now part of the authoritative product record.

Guidelines & Best Practices

- **Review AI Confidence:** Pay close attention to items where data was semi-structured via custom instructions.
- **Human-in-the-Loop:** All AI-generated outputs must be reviewed by a qualified engineer before system commitment.



- **Resolution Mode:** Ensure you are logged in correctly; your import will respect your native Aras Innovator permissions.



Smart Variants Agent - User Guide

This User Guide details the end-to-end process of using the **Smart Variants Agent** to transform existing Bill of Materials (BOM) configurations into structured **Variant Management** models within Aras Innovator.

Overview

The **Smart Variants Agent** is a purpose-built Task Assistant Agent within the **InnovatorEdge AI** framework.

It enriches the digital thread by analyzing the configuration relationships within a product's 150% BOM data to infer rules and automate the creation of governed Variant Management items.

What the Smart Variants Agent Does

The agent facilitates the configuration process by performing the following:

- **Direct System Access:** Reads BOM structures directly from Aras Innovator via the Edge API, requiring no manual document uploads.
- **Rule Extraction:** Sends a BOM representation to the AI Agents Service to derive variant rules through deterministic co-occurrence analysis.
- **Intelligent Visualization:** Displays a merged **150% BOM**, coupled (**AND**) rules, either/or (**XOR**) rules, and a **dependency matrix (Implies/Exclude)**.
- **Governed Commitment:** Creates Variant Management entities in Innovator only after the user executes the final import.

Purpose

- **Manual Reduction:** Reduces the engineering effort required to manually model variability from BOM data.
- **Verification Separation:** Separates the analysis phase from the import phase to ensure mandatory human validation before any data is committed to Innovator.
- **Digital Thread Integration:** Supports downstream configuration management with structured Variant Management objects.

Preparation



The quality of the AI-driven analysis is directly proportional to the integrity of the source data in Aras Innovator.

For optimal results, ensure:

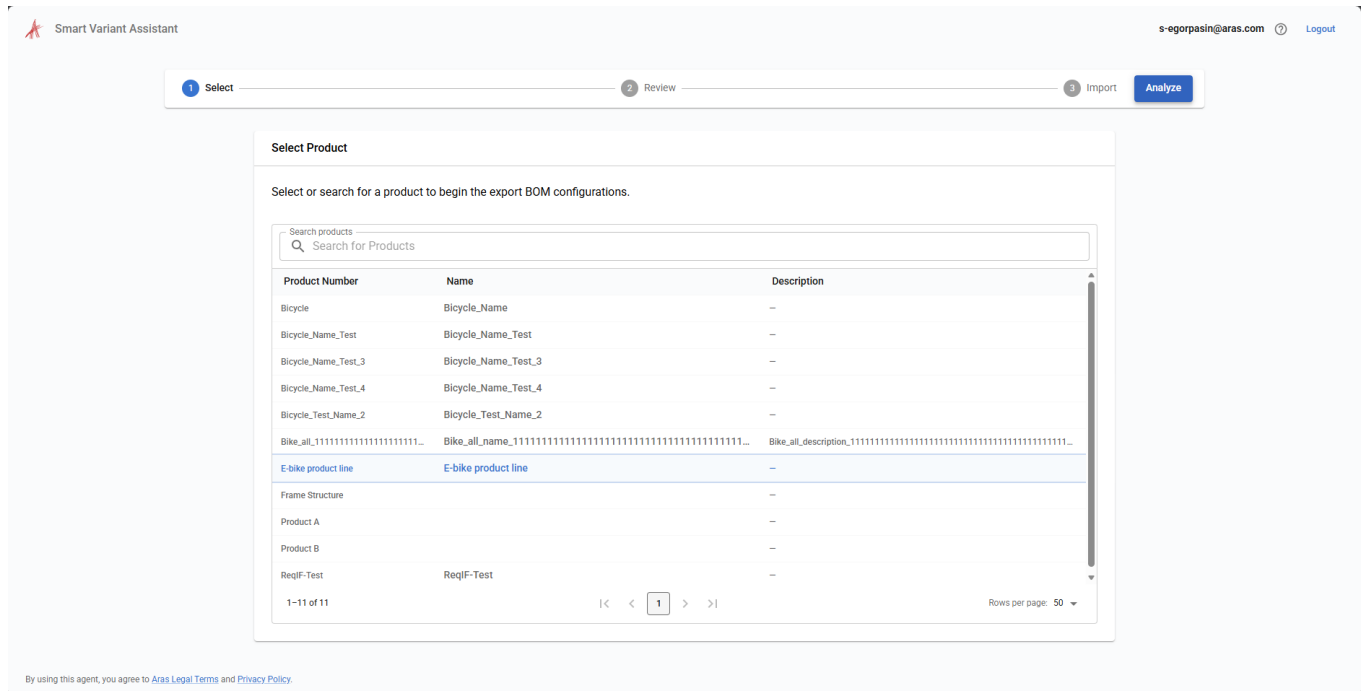
- **Multiple Configurations:** The product must have multiple 100% BOM configurations. The agent compares these to determine which components are mandatory, optional, coupled, or mutually exclusive. Single-configuration products will not produce meaningful rules.
- **Standardized Naming:** Parts with identical keyed_name values across configurations are identified as the same component. Inconsistent naming can result in duplicate or missing rules.
- **Structural Depth:** The agent retrieves and analyzes structures up to **nine nesting levels** of Part_BOM navigation properties. Structures exceeding this limit may not be fully represented.
- **Clean Data:** Ensure relationships in the system of record are clearly defined, as missing or ambiguous links reduce the accuracy of inferred rules.

Product Selection and Analysis

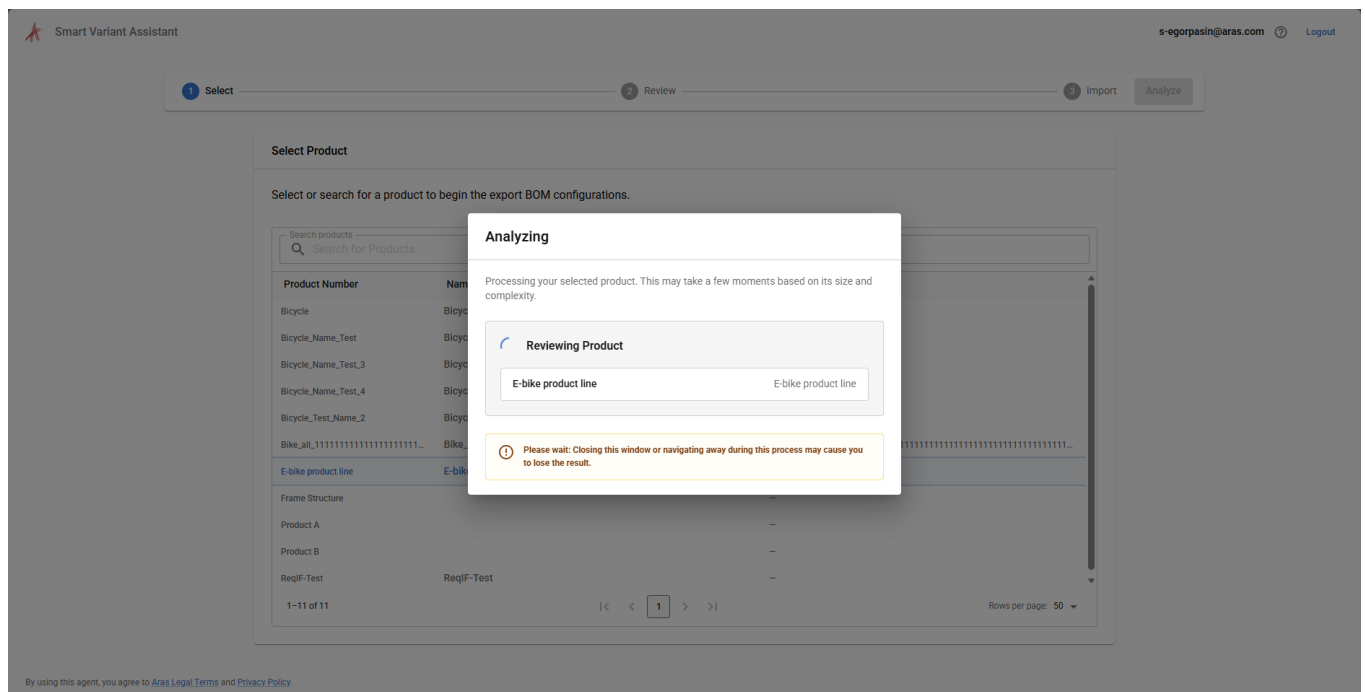
The workflow begins in the **Select Product** interface, where users identify the target for configuration modeling.

1. **Identify Target Product:** Use the **Search products** field to filter the list by name, item number, or description.
2. **Select Target:** Click a row in the table to highlight a single product.





3. Click Analyze: Selecting Analyze in the top-right header initiates the process.
4. Monitor Progress: A non-dismissible Analyzing modal tracks progress as the agent fetches the BOM expansion and generates configuration rules.



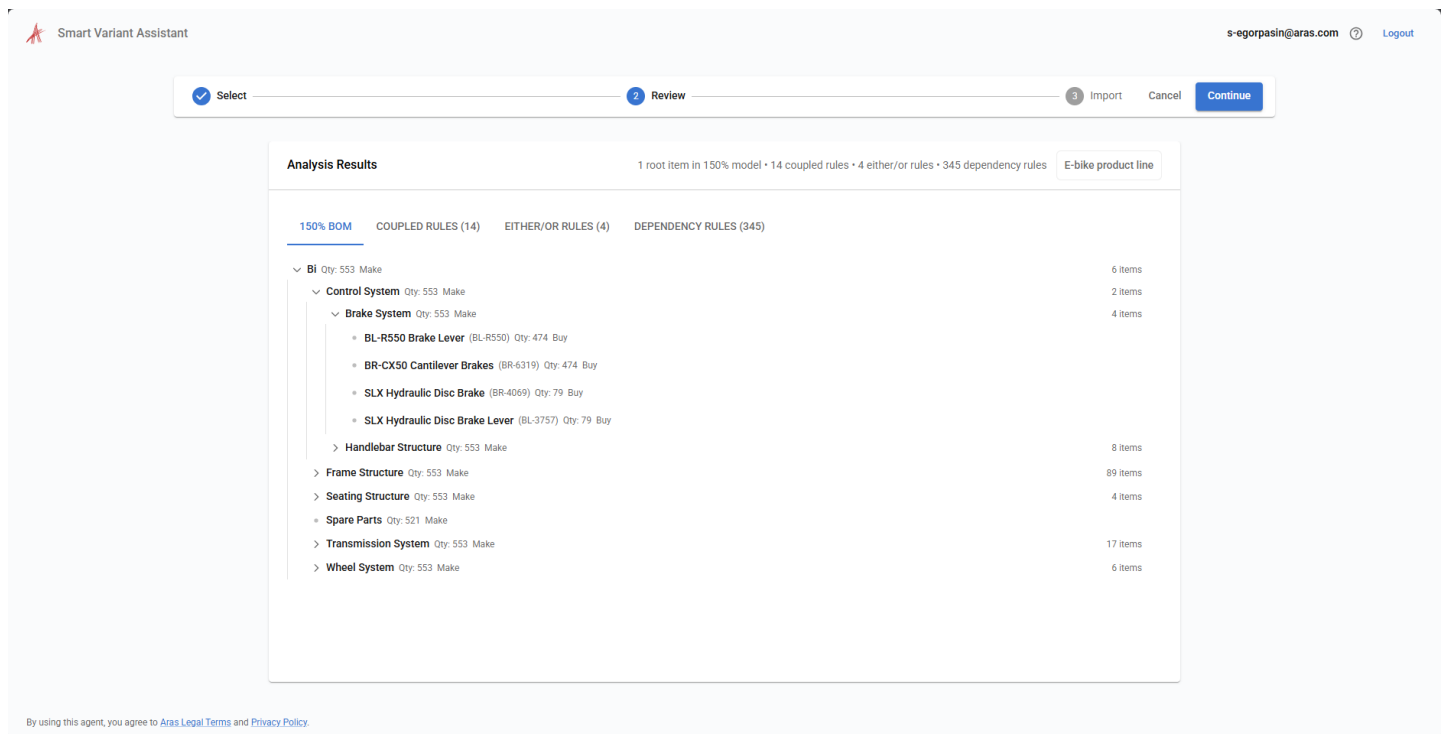
Validation and Review

Once analysis is complete, the agent moves to the **Review** step to present the **Analysis Results**. This environment allows domain experts to verify the AI's interpretation before system commitment.

The Review Interface Layout

The interface summarizes findings in a single line (e.g., "1 root item in 150% model & 14 coupled rules & 4 either/or rules & 345 dependency rules") and presents detailed data in four functional tabs:

- **150% BOM:** An expanded tree view showing the merged hierarchy across all configurations. Mandatory items are distinguished from optional items.



- **Coupled Rules:** A searchable table of parts that always appear together across analyzed configurations.

Smart Variant Assistant
s-egorpasin@aras.com
Logout

Select
Review
Import
Cancel
Continue

Analysis Results

1 root item in 150% model • 14 coupled rules • 4 either/or rules • 345 dependency rules E-bike product line

150% BOM **COUPLED RULES (14)** EITHER/OR RULES (4) DEPENDENCY RULES (345)

#	Rule Elements (must all be selected together)
1	BR-CX50 Cantilever Brakes AND BL-R550 Brake Lever AND A Black Carbon blades, Aluminum Alloy Steerer Fork AND SMR3 Pro Black Carbon Saddle AND Carbon Seatpost AND Road Bike 700c Wheelset AND DHL-92 rear disc hub AND DHG-150 front disc hub AND 26" 100mm 32h AND Stainless Black Spoke AND Apache 700 X 25c AND Threaded 63mm Bottom Bracket AND SPD-SL Clipless Road Pedals AND 11-32 Nickel-Plated Cassette AND 11-42 Nickel-Plated Cassette
2	24.4" Aluminum Flat Bar AND A Black Steel Fork AND Fat Bike 26" Wheelset AND 26" 100mm 32h AND Stainless Black Spoke AND SL-1240 front disc hub AND DHL-197 rear disc hub AND Bulldozer 26" X 3.9" 72tpi AND Flat Steel Pedal AND 10-42 Chromoly Steel Cassette
3	30.7" Aluminum Riser Bar AND 150-170mm Travel Fast-Black Coated Aluminum Fork AND Mountain Bike 27.5" Wheelset AND Stainless Black Spoke AND M1900 Spline 27mm rim - 27.5" AND DHG-150 front disc hub AND DHL-92 rear disc hub AND Rail 27.5" X 1.95" 72tpi AND MTB SPD Pedals AND 10-50 Stamped Steel Cassette
4	X1 11-Speed Trigger Shifter AND SLX Hydraulic Disc Brake AND SLX Hydraulic Disc Brake Lever AND Mountain Bike Air Rear shock AND SMP Well Black Nylon Saddle AND SL Aluminum Switch, dropper, 30.9mm AND Aluminum 11-Speed Long Rear Derailleur AND Non-Threaded 73mm Bottom Bracket
5	EA90 Aluminum 3.5" Stem AND 3.5" EA90 Aluminum Tubing AND 8" Steerer Clamp AND 31.8mm Handlebar Clamp AND 4mm Steel Allen Bolt
6	170mm 11-Speed Carbon Fiber Standard Chaining C... AND 170mm Carbon Fiber Hollow Crank Arm AND Steel Spider Bolts AND X-GlideR 55T Chaining AND Aluminum Alloy Crank Spider
7	EA90 Aluminum 3.9" Stem AND 4mm Steel Allen Bolt AND 31.8mm Handlebar Clamp AND 3.9" EA90 Aluminum Tubing AND 8" Steerer Clamp

1-14 of 14 |< < 1 > >| Rows per page: 50

By using this agent, you agree to [Aras Legal Terms](#) and [Privacy Policy](#).

- Either/Or Rules:** A view identifying mutually exclusive groups where exactly one option is valid per configuration.



Smart Variant Assistant
s-egorpasin@aras.com
Logout

Select
Review
Import
Cancel
Continue

Analysis Results

1 root item in 150% model • 14 coupled rules • 4 either/or rules • 345 dependency rules E-bike product line

150% BOM COUPLED RULES (14) **EITHER/OR RULES (4)** DEPENDENCY RULES (345)

Search either/or rules
Filter by part name...

#	A Elements		B Elements
1	BL-R550 Brake Lever BR-CX50 Cantilever Brakes A Black Carbon blades, Aluminum Alloy Steerer Fork Carbon Seatpost SMR3 Pro Black Carbon Saddle 11-32 Nickel-Plated Cassette 11-42 Nickel-Plated Cassette SPD-SL Clipless Road Pedals Threaded 63mm Bottom Bracket Apache 700 X 25c Road Bike 700c Wheelset 26" 100mm 32h DHG-150 front disc hub DHL-92 rear disc hub Stainless Black Spoke	XOR	SLX Hydraulic Disc Brake SLX Hydraulic Disc Brake Lever X1 11-Speed Trigger Shifter Mountain Bike Air Rear shock SL Aluminum Switch, dropper, 30.9mm SMP Well Black Nylon Saddle Aluminum 11-Speed Long Rear Derailleur Non-Threaded 73mm Bottom Bracket
2	Spare Parts	XOR	30.7" Aluminum Riser Bar 150-170mm Travel Fast-Black Coated Aluminum Fork 10-50 Stamped Steel Cassette MTB SPD Pedals Mountain Bike 27.5" Wheelset DHG-150 front disc hub DHL-92 rear disc hub M1900 Spline 27mm rim - 27.5" Stainless Black Spoke Rail 27.5" X 1.95" 72psi
3	SLX Hydraulic Disc Brake SLX Hydraulic Disc Brake Lever X1 11-Speed Trigger Shifter Mountain Bike Air Rear shock SL Aluminum Switch, dropper, 30.9mm SMP Well Black Nylon Saddle Aluminum 11-Speed Long Rear Derailleur	XOR	BL-R550 Brake Lever BR-CX50 Cantilever Brakes A Black Carbon blades, Aluminum Alloy Steerer Fork Carbon Seatpost SMR3 Pro Black Carbon Saddle 11-32 Nickel-Plated Cassette 11-42 Nickel-Plated Cassette SPD-SL Clipless Road Pedals

1~4 of 4 |< < 1 > >| Rows per page: 50

By using this agent, you agree to [Aras Legal Terms](#) and [Privacy Policy](#).

- **Dependency Rules:** An interactive matrix mapping Implies (if A, then B) and Exclude (A and B cannot coexist) relationships.

Use the interactive toolbar to Expand All or Collapse All groups during auditing.

Smart Variant Assistant
 s-egorpasin@aras.com [Logout](#)

Select
Review
Import
Cancel
Continue

Analysis Results

1 root item in 150% model • 14 coupled rules • 4 either/or rules • 345 dependency rules E-bike product line

150% BOM COUPLED RULES (14) EITHER/OR RULES (4) DEPENDENCY RULES (345)

Expand All Collapse All 105 cols • 110 rows Mutual implies Implies Mutual excludes Excludes

	17.3" Alumi...	Bell	11 Speed S...	16.5" Alumi...	15.7" Alumi...	24.4" Alumi...	X1 11-Spee...	30.7" Alumi...	BR-CX50 Ca...	BL-R5
Bi										
Control System										
Handlebar Structure										
17.3" Aluminum Drop Bar										
Bell										
11 Speed STI Shifters										
16.5" Aluminum Drop Bar										
15.7" Aluminum Drop Bar										
24.4" Aluminum Flat Bar										
X1 11-Speed Trigger Shifter										
30.7" Aluminum Riser Bar										
Brake System										
BR-CX50 Cantilever Brakes										
BL-R550 Brake Lever										

By using this agent, you agree to [Aras Legal Terms](#) and [Privacy Policy](#).

System Commitment

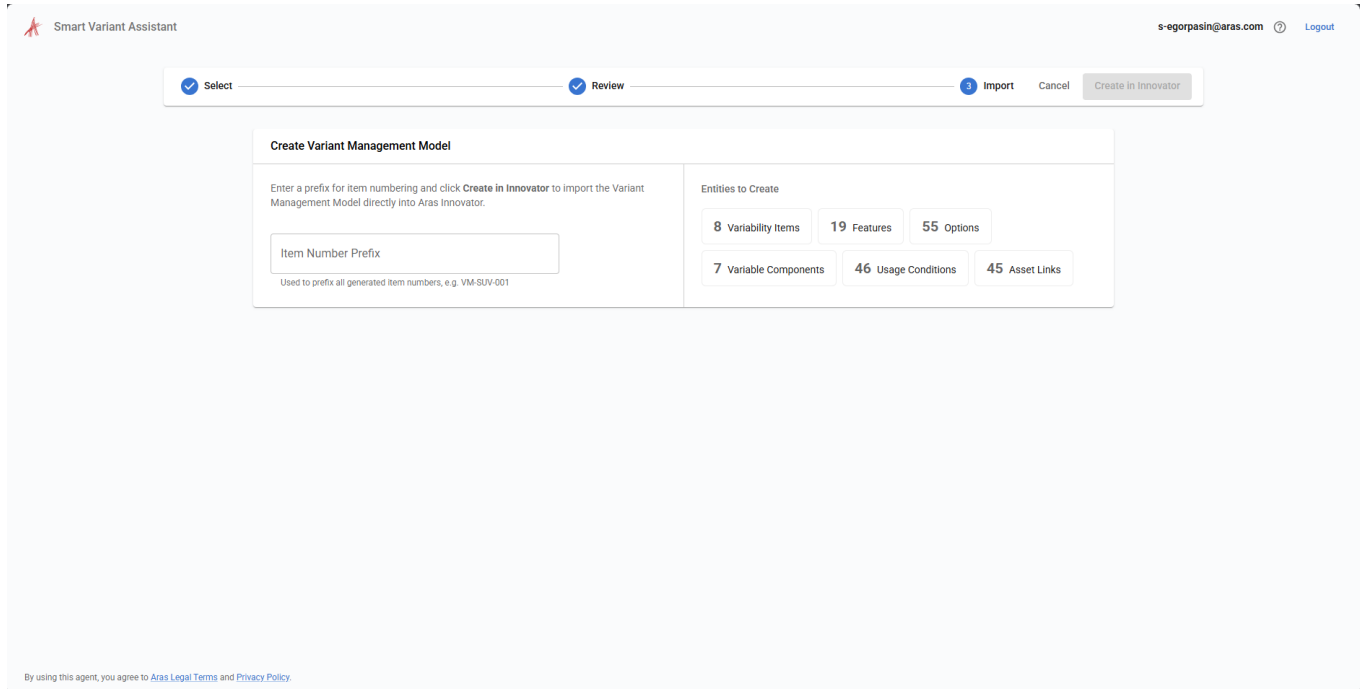
After clicking **Continue** in the review phase, the process moves to the final commitment step.

Create Variant Management Model

This screen allows you to configure the final import parameters:

- Review Entities to Create:** A summary panel lists the counts of entities to be generated, including **Variability Items**, **Features**, **Options**, **Variable Components**, **Usage Conditions**, and **Asset Links**.
- Define Identifiers:** Enter an **Item Number Prefix** (e.g., VM-eBike). This is prepended to all generated item numbers.

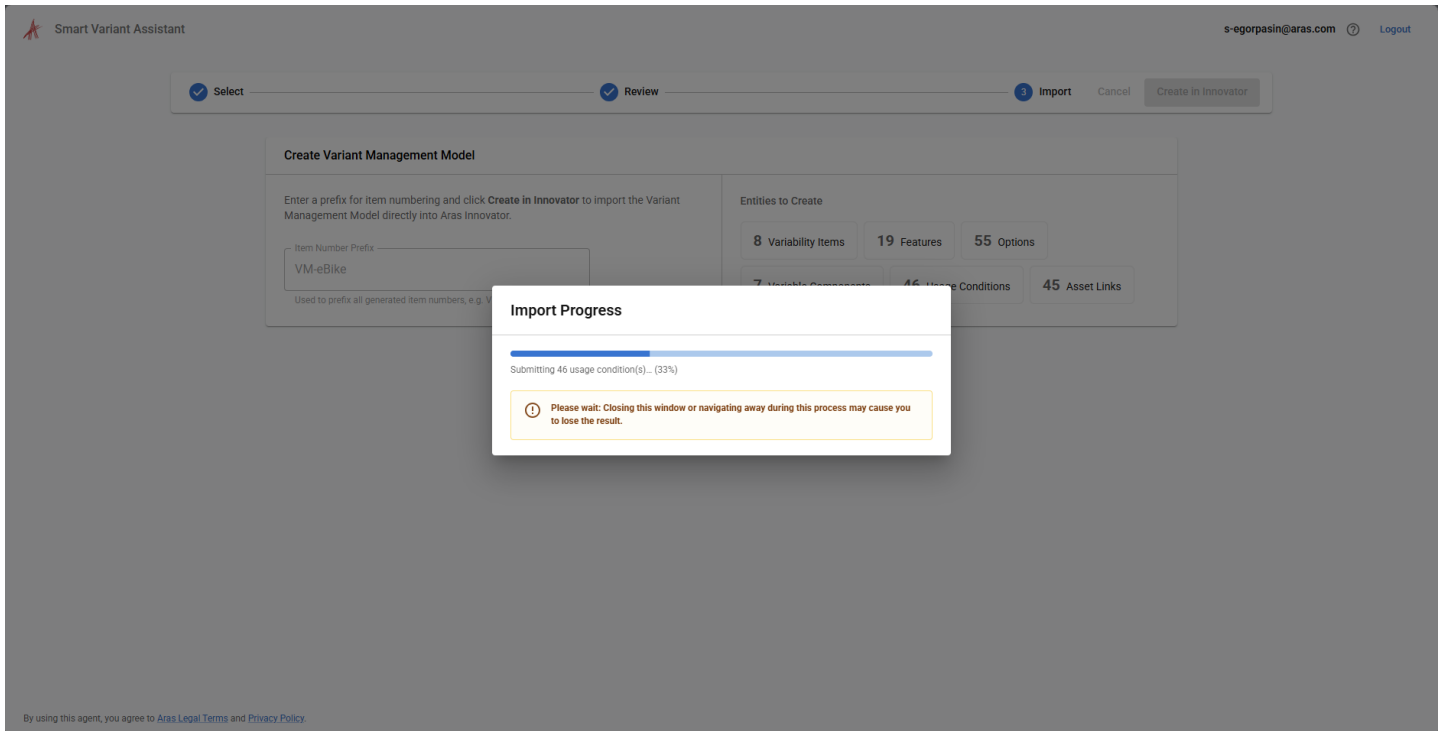




The screenshot shows the 'Smart Variant Assistant' interface. At the top, there's a progress bar with three steps: 'Select' (checked), 'Review' (checked), and 'Import' (active). Below the progress bar, the main section is titled 'Create Variant Management Model'. It contains a text input field for 'Item Number Prefix' with a hint: 'Enter a prefix for item numbering and click **Create in Innovator** to import the Variant Management Model directly into Aras Innovator.' Below the input field, a small note says 'Used to prefix all generated item numbers, e.g. VM-SUV-001'. To the right, under 'Entities to Create', there are six buttons showing counts: '8 Variability Items', '19 Features', '55 Options', '7 Variable Components', '46 Usage Conditions', and '45 Asset Links'. At the bottom right, there are buttons for 'Import', 'Cancel', and 'Create in Innovator'. The top right corner shows the user 's-egorpasin@aras.com' and a 'Logout' link. At the bottom left, a small text says 'By using this agent, you agree to [Aras Legal Terms](#) and [Privacy Policy](#)'.

3. Click **Create in Innovator**: This initiates the **Import Progress** modal, which executes in three phases:

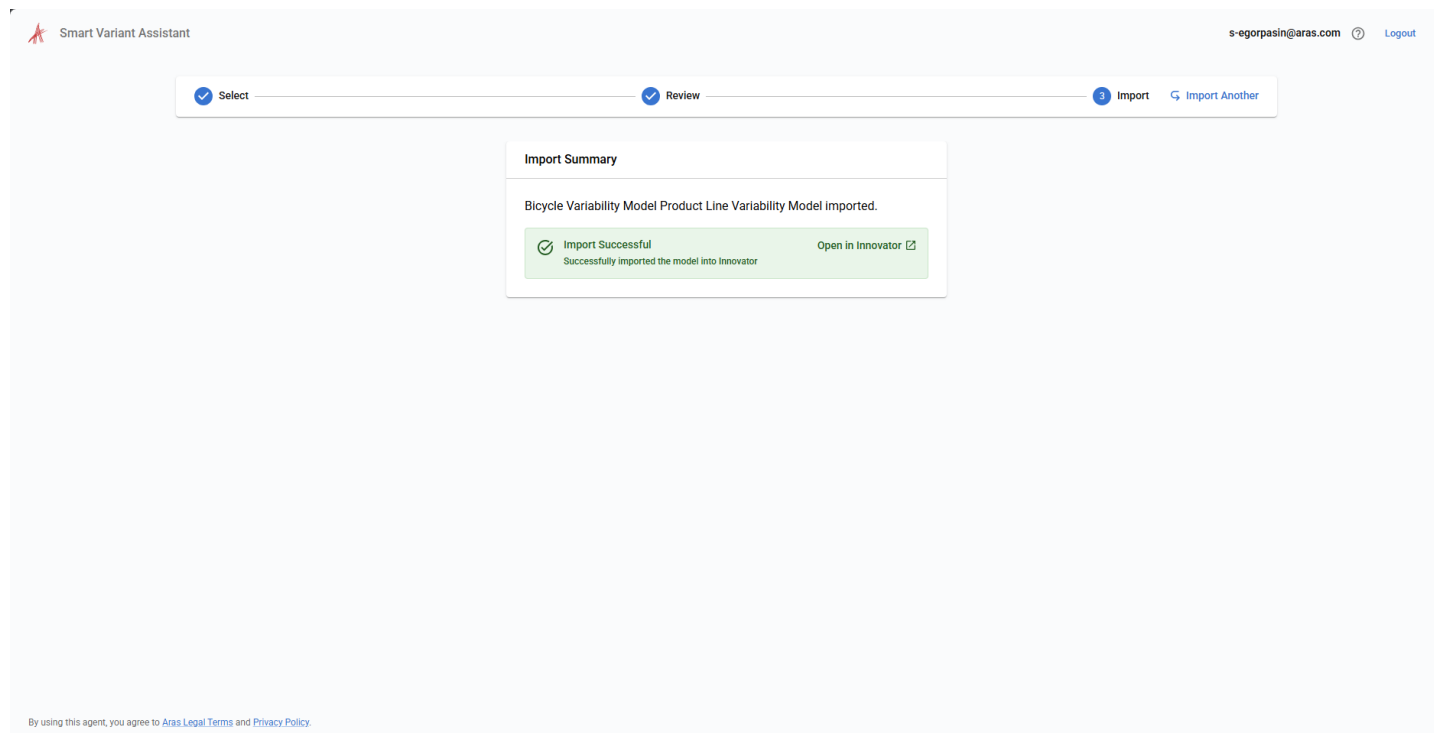
- Phase 1: Core `vm_VariabilityItem` (including features and options).
- Phase 2: `vm_Usage` conditions.
- Phase 3: `vm_VariableComponent` and asset relationships.



Import Summary

Upon successful commitment, an **Import Summary** is displayed.

- **Open in Innovator:** Click the provided link to open the **Root Variability Item** in the native Aras Innovator interface.
- **System Records:** In Innovator, you can navigate the resulting **Variability Structure** and **Usage Conditions** tabs to verify the generated model.
- **Continuous Workflow:** Select **Import Another** to return to the selection step for a different product.



Native Innovator Interaction

Once the model is committed to Aras Innovator, it becomes part of the authoritative digital thread. You can perform the following directly within the native interface:

- **Metadata Review:** View the generated **Number**, **Revision**, and **State** (defaulting to Preliminary).

The screenshot shows the Aras Innovator web application interface. At the top, there's a search bar and a user profile icon. Below the header, the main content area is titled 'VM-eBike-001 Bicycle Variability Model'. It includes a sidebar with navigation options: 'Variability Item', 'Relevant Features', 'Variability Items', 'Variability Structure', and 'Usage Conditions'. The 'Variability Item' section is active, showing a form with fields for 'Number' (VM-eBike-001), 'Revision' (A), 'State' (Preliminary), 'Assigned Creator', 'Name' (Bicycle Variability Model), 'Designated User', and 'Description' (Variability model capturing optional components and configuration rules for a customizable bicycle). Below this, the 'Variability Items' section is active, displaying a table with columns: Number, Name, Description [...], and Sequence. The table is empty, showing a 'No results found' message. The interface also includes a search bar at the top and various action buttons like 'Edit', 'Refresh', 'Promote', 'Navigate', 'Reports', 'Share', and 'More'.

Variability Items: shows a flat list for auditing all child entities, displaying their system-generated numbers, descriptions, and current lifecycle states.

You can select individual entries to perform native Innovator actions such as edits or lifecycle promotions.

The screenshot displays the Aras InnovatorEdge AI interface for the 'VM-eBike-001 Bicycle Variability Model'. The interface includes a top navigation bar with the Aras logo, a search bar, and user information. Below the navigation bar, there's a sidebar with icons for various functions. The main content area is divided into several sections:

- Model Header:** Shows the model name 'VM-eBike-001 Bicycle Variability Model' with options to Edit, Refresh, Promote, Navigate, Reports, Share, and a menu icon.
- Variability Item Form:** Contains fields for Number (VM-eBike-001), Revision (A), State (Preliminary), Assigned Creator, Name (Bicycle Variability Model), Designated User, and a Description (Variability model capturing optional components and configuration rules for a customizable bicycle).
- Relevant Features:** A tabbed interface with 'Variability Items' selected, showing a list of items with columns for Number, Name, Rev., State, Description, and Sequence.

Number	Name	Rev.	State	Description [...]	Sequence
VM-eBike-002	Handlebar Structure	A	Preliminary	Options for handlebar types, shifters, and ...	10
VM-eBike-003	Brake System	A	Preliminary	Options for brake types and levers.	20
VM-eBike-004	Frame Structure	A	Preliminary	Options for frame material, size, and fork t...	30
VM-eBike-005	Seating Structure	A	Preliminary	Options for saddle and seatpost types.	40
VM-eBike-006	Wheel System	A	Preliminary	Options for wheelset and tire types.	50
VM-eBike-007	Transmission System	A	Preliminary	Options for drivetrain, crankset, cassette, ...	60
VM-eBike-008	Spare Parts	A	Preliminary	Optional spare parts for maintenance.	70

At the bottom of the table, it shows 'Page: 1 of 1' and '7 Results'.

- **Variability Structure:** Navigate the Variability Structure tab to see the multi-level hierarchy of items and features established by the agent.



The screenshot displays the Aras Innovator interface for the 'VM-eBike-001 Bicycle Variability Model'. The top navigation bar includes the Aras Innovator logo, a search bar, and a user profile icon. Below the navigation bar, there's a breadcrumb trail and a set of action buttons (Edit, Refresh, Promote, Navigate, Reports, Share). The main content area is divided into two sections: 'Variability Item' and 'Variability Structure'. The 'Variability Item' section shows fields for Number (VM-eBike-001), Revision (A), State (Preliminary), Assigned Creator, Name (Bicycle Variability Model), Designated User, and Description. The 'Variability Structure' section shows a tree view of the model's components, including Handlebar Structure, Brake System, and Frame Structure, each with its own revision and state.

Governance and Permissions

All items created by the agent are subject to standard **Aras Innovator Governance**.

This means:

- **User Permissions:** Your ability to see, edit, or delete the generated Variant Management items is strictly tied to your native Aras Innovator identity and assigned roles (e.g., Configuration Management Group).
- **Lifecycle Management:** Committed items follow native lifecycle transitions (e.g., Preliminary to In Review) and versioning rules established in your environment.

Guidelines & Best Practices

- **Ensure Multiple Configurations:** The agent identifies mandatory, optional, or mutually exclusive components by comparing different 100% BOMs. Products with only a single configuration will not produce meaningful rules.
- **Standardize Part Naming:** Components must use consistent keyed_name values across all variants to be identified as the same item. Inconsistent naming results in duplicate or missing configuration rules.



- **Adhere to Input Constraints:** To ensure reliable processing and retrieval, keep search field inputs under **100 characters** and item number prefixes to a maximum of **12 characters**.
- **Human-in-the-Loop Validation:** All AI-derived rules and the 150% BOM must be reviewed and confirmed by an engineer before committing the model to the system of record.
- **Maintain Active Session State:** Do not close the browser window or navigate away while the **Analyzing** or **Import Progress** modals are active to prevent the loss of session data.

Troubleshooting

Issue	Solution
No products found	Check search text; clear search. Confirm products exist and you have read permissions.
Analysis takes too long	Large BOMs require more time; keep the modal open to prevent session loss.
Analysis fails	Check your network; the AI Agents Service must be reachable.
Import errors	Read error messages in the dialog. Try a different prefix or check permissions.
Partial import failure	If a later phase fails, earlier phases may have committed data. Remediation in Innovator may be required.
Open in Innovator missing	Ensure VITE_INNOVATOR_URL is set in the configuration.
Rules seem incorrect	Verify the product has multiple BOM configurations and consistent part naming.
403 Error after Analyze	Rotated refresh tokens may be stale; ensure the authentication layer is synchronized.



Impact Analysis AI Agent - User Guide

Overview

The **Impact Analysis AI Agent** is a **Task Assistant Agent** within the **InnovatorEdge AI** framework. It operates directly inside Aras Innovator's Unified Change Management (UCM) system of record to assist Change Authors and Change Specialists during the drafting phase of an engineering Change Request (CR).

The agent analyzes the Change Request against your product structure and historical Change Order patterns, proposes a set of Affected Items with explicit AI-generated rationale, and lets the engineer review and selectively commit those changes, all without leaving the agent workspace.

What the Impact Analysis AI Agent Does

The agent supports engineers through four consecutive steps:

- **Select** the target Change Request from a searchable list of Draft-state CRs.
- **Analyze** the CR context by running five AI specialists in parallel against your product structure and historical Change Order records.
- **Review** a recommended set of Affected Items, each with a suggested action (Revise, Release, or Obsolete) and an AI-generated justification.
- **Commit** the accepted recommendations to the Change Request as Affected Item relationships in Aras Innovator, with rationale recorded in the SSVC comment thread for traceability.

Important

The agent is user-invoked and human-validated. No data is written to Aras Innovator until users choose to **Apply Changes**.

Prerequisites

Before starting an analysis session, confirm the following:

- **CR content:** The target Change Request must have substantive text in the **Title** and **Reason of Change** fields. Sparse or empty fields reduce recommendation quality.



- **Product structure:** Parts in the CR's existing Affected Items should have their structural links and child relationships correctly declared in Aras Innovator. Gaps in the product graph reduce recommendation coverage.
- **Item number requirement:** The agent identifies recommended items by their `item_type` and `item_number` values. Items without an `item_number` value may be skipped or misidentified in the recommendation set. For standard UCM ItemTypes this is not an issue. For custom ItemTypes where `item_number` is not required or not unique, recommendations for those items may not work as expected - verify data quality before running analysis.
- **Lifecycle state:** The Change Request must be in **Draft** status. Only Draft CRs appear in the selection list.
- **Permissions:** Your Aras Innovator account must have read and update access to Change Requests and read access to Affected Items (Parts, CADs, Documents). Contact your system administrator if you are unsure whether your account has the required permissions.

Selecting a Change Request

Open the Impact Analysis application from the URL provided by your system administrator and log in. The application opens at the Initializing step, showing a searchable list of all Draft-state Change Requests in your connected Aras Innovator instance.

Search by CR number or title keyword, select the row for the CR you want to analyze, and click **Load Data**. The application advances to the **Preparing** step.

Important

Only Change Requests in **Draft** status appear in this list. If the CR you are looking for does not appear, verify its lifecycle state in Aras Innovator before proceeding.

Preparing the Analysis

The **Preparing** step displays the loaded Change Request context: its title, Reason of Change, and Description, drawn directly from Aras Innovator. These fields are read-only here; if the content needs correction, update it in Innovator first and reload.



On the right, an optional **Analysis Instructions** field lets you focus or constrain the analysis before it runs. Instructions are free text and apply for this session only. Examples:

- "Prioritize historically validated Change Orders and call out high-risk affected items."
- "Focus only on components within the chassis sub-assembly. Ignore optical elements."
- "This change is safety-critical. Apply conservative recommendations."

When ready, click **Start Analysis**. Use **Back** to return and select a different CR.

Analysis in Progress

After clicking **Start Analysis**, the agent runs five AI specialists in parallel.

Their progress is visible on screen; each shows a running indicator while active and a checkmark when complete:

- **Change Management Expert:** Orchestrates the analysis and structures the final recommendations.
- **Part Model Checker:** Audits part properties and structural relationships.
- **Rationale Analyzer:** Builds a justification for each recommended item.
- **Prior Change Order Checker:** Searches historical Change Orders for relevant patterns.
- **ARAS Expert:** Reads live data directly from your Aras Innovator instance to ground recommendations in your actual product record.

Once the five specialists complete, a **Recommendation Validator** runs as a final step to consolidate and validate the recommendation set before results are presented.

Important

Do not close the browser window, refresh, or navigate away while the analysis is running. This will terminate the session and discard all progress. Use the Cancel button to safely abort.

Agent-Initiated Clarification

In some sessions, after the specialists complete, the agent pauses and presents a plain-language question before showing final recommendations. This occurs when the specialists produce conflicting



signals, for example, when multiple historical Change Orders point to different scopes, and the agent needs user guidance to proceed.

The interface shows the question and a set of clickable option chips, each representing a distinct resolution path. Select the chip that best matches your intent and click **Continue**. The agent resumes and advances to the Summary.

Impact Analysis

Progress: 1. Initializing (checked) 2. Preparing (checked) 3. Summary (active) 4. Applying (disabled) Cancel Apply Changes

Waiting for input
5 of 5 steps complete

Context: CR: CR-00012218 Similar COs (2)

Agent response:
I found one scope conflict that materially affects the recommendation set: CAM-DOC-2200 appears in the current CR, but the specialists disagree on whether it is actually relevant. Which scope do you want applied?

Keep CAM-DOC-2200 in scope Remove CAM-DOC-2200 from scope

Continue

By using this agent, you agree to [Aras Legal Terms](#) and [Privacy Policies](#).

Aras Innovator Instance: 145078-optimum-qrtime

At any point during analysis you can click the **CR** badge or the **Similar COs** badge in the header to review the full detail of the loaded Change Request or the historically similar Change Orders the agent identified.

Reviewing Recommendations

When analysis is complete, the **Summary** step presents the results.

Read the Summary narrative first. Before evaluating individual items, read the paragraph at the top of the results. It describes the reasoning the agent applied, including any scope decisions made in response to clarification, and provides the context that makes the individual recommendations interpretable.



Recommendation list: Each recommended Affected Item appears as a card showing the item number, its ItemType, and a suggested action badge (Revise, Release, or Obsolete). Expand a card to see:

- **Rationale:** A synthesized justification from the Change Management Expert.
- **Checkers Insights:** Individual evidence notes from each specialist that flagged the item, showing exactly which historical Change Orders or structural patterns drove the recommendation. Use these to cross-reference against the source records in Aras Innovator before committing.

Important

Two separate "Action" concepts

The suggested action badge on a recommendation card (Revise, Release, Obsolete) is the agent's analytical recommendation based on historical patterns.

The Action column in the **Aras Innovator** Affected Items grid is a separate UCM field set through normal change governance. These two values are independent and are not expected to match automatically.

Accepting and rejecting items: Each card has a checkbox, selected by default.

Uncheck any item to exclude it from the commit.

The **Apply Changes (N)** counter in the header updates as you toggle selection.

Use the **All / Selected / Not Selected** filter to focus the view. Apply Changes is disabled when no items are selected.

Use **Back** to return to the **Preparing** step to revise your instructions and re-run. This discards the current recommendation set.

Applying Changes to Innovator

Click **Apply Changes (N)** to commit the selected items. The agent writes the Affected Item relationships to the Change Request in Aras Innovator in a single operation. A progress indicator is shown while the commit is in progress.

Important



Do not close the browser window or navigate away during the commit. Use **Process Another** after a successful apply to start a new session for a different CR without re-authenticating.

Successful Apply

On success, an **Apply Changes** “**Summary**” is displayed with a count of relationships created and an **Open in Innovator** link to the Change Request record.

Apply Errors

If one or more items fail, the page shows the specific error returned by Aras Innovator. The most common causes are:

- **Relationship already exists:** The agent recommended an item already present in the CR's Affected Items. This is expected when the agent confirms an existing item's scope, i.e., the relationship already exists in Innovator. Verify the item in Innovator and update the action on the existing relationship manually if needed.
- **Insufficient permissions:** Your Aras Innovator account does not have write access to the Affected Items relationship on this Change Request.

Verifying Results in Aras Innovator

After a successful apply, open the Change Request in Aras Innovator to confirm:

1. Navigate to the **Affected Items** tab and verify the expected items are present with the correct Type and Item identifier.
2. Navigate to the **Secure Messages** section. The agent records a structured rationale entry for each committed item, creating a durable audit trail linked to the Change Request.

The Change Request remains in **Draft** status. The agent does not advance it through its lifecycle. Continue your normal UCM workflow from this point.

Guidelines and Best Practices

- **Richer CR context yields better recommendations.** Ensure the Change Request has substantive text in the Title, Description, and Reason of Change fields before starting. A title like



"Update part" provides far less signal than "Replace aluminum clamp bracket with steel for load tolerance."

- **Use instructions to focus, not to override.** Custom instructions are most effective when they narrow the scope to a specific sub-assembly, a risk category, a constraint. The agent's analysis is grounded in your actual product graph; instructions complement but do not replace that evidence.
- **Use Checkers Insights for verification.** The per-specialist evidence trail is designed to support your validation step in Aras Innovator, not just to justify acceptance. Cross-reference the Prior Change Order Checker's cited CO numbers against the source records before committing.
- **The agent does not promote your CR.** Committing Affected Items does not change the lifecycle state of the Change Request. Continue your normal UCM workflow in Aras Innovator after applying.

Troubleshooting

Symptom	Resolution
CR does not appear in the selection list.	Verify the CR is in Draft status in Aras Innovator. Only Draft CRs are shown.
Analysis does not advance past the connecting state.	The agent service is not responding. Refresh and retry. If the issue persists, contact your system administrator.
" Analysis failed " on the Summary step.	Retry. If the CR has very limited text in its Title, Description, or Reason of Change, add meaningful content in Innovator before retrying.
Apply returns "already exists" for all items.	The agent confirmed the existing Affected Items scope is correct. No new relationships need to be created. Manually update action fields in Innovator if needed.
Open in Innovator link does not resolve.	Verify your network access to the Aras Innovator instance shown in the application footer.



Aras InnovatorEdge AI Agents - Release Notes

This page includes Aras InnovatorEdge AI Agents Release Notes starting with the most recent.

2026 August 18 - Edge AI Agents - Impact Analysis AI Agent v.1.0.1 - Release Notes

Impact Analysis Edge AI Agent 1.0.1 is now live!

The business features and benefits are as follows:

- **Maintenance:** Updated underlying infrastructure to maintain availability of the Impact Analysis Edge AI Agent.
-

2026 August 18 - Edge AI Agents - Smart Variants Agent v.1.0.3 - Release Notes

Smart Variants Edge AI Agent release 1.0.3 is now live!

The business features and benefits are as follows:

- **Maintenance:** Updated underlying infrastructure to maintain availability of the Smart Variants Edge AI Agent.
-

2026 August 18 - Edge AI Agents - BOM Import Agent v.1.0.3 - Release Notes

BOM Import Edge AI Agent 1.0.3 is now live!

The business features and benefits are as follows:

- **Maintenance:** Updated underlying infrastructure to maintain availability of the BOM Import Edge AI Agent.
-

2026 August 18 - Edge AI Agents - Requirements Ingestion Agent v.1.0.4 - Release Notes

Requirements Ingestion Edge AI Agent release 1.0.4 is now live!

The business features and benefits are as follows:

- **Maintenance:** Updated underlying infrastructure to maintain availability of the Requirements Ingestion Edge AI Agent.

2026 March 23 - Edge AI Agents - Impact Analysis AI Agent v1.0.0 - Release Announcement

Impact Analysis AI Agent release 1.0.0 is now live!

Overview

Aras is pleased to announce the general availability of the Impact Analysis AI Agent v1.0.0. Operating as a specialized Task Assistant Agent, this module integrates directly into your change management workspace to analyze unstructured context fields, evaluate configuration dependencies across product branches, and automate the identification of mandatory Affected Items.

Why This Matters

Tracing downstream change propagation across deeply nested assembly hierarchies has historically forced Change Authors and Specialists into resource-intensive database lookups. The Impact Analysis AI Agent addresses this friction by reallocating human engineering hours away from manual traversal and transcription toward authoritative verification and high-value design validation. By grounding all predictive insights inside your authoritative system of record, Aras ensures that change evaluations are context-aware, reproducible, and fully auditable.

What Is Included in This Release



Query-on-the-Fly Analysis Engine

When a Change Request is submitted for analysis, the agent automatically runs five AI specialists in parallel to evaluate the change scope, examine product structure, and identify which items need to be included:

- **Change Management Expert:** Orchestrates the analysis and assembles the final recommendations.
- **Part Model Checker:** Audits part properties and structural relationships.
- **Rationale Analyzer:** Builds a justification for each recommended item.
- **Prior Change Order Checker:** Searches historical Change Orders for relevant patterns.
- **ARAS Expert:** Reads live data directly from your Aras Innovator instance to ground recommendations in your actual product record.

Human-in-the-Loop Review Dashboard

The agent guides engineers through a four-stage wizard flow designed to ensure no change is committed without deliberate human review:

1. **Prepare:** Load the target Change Request, review its context fields (title, description, reason for change, existing Affected Items), and optionally provide custom analysis instructions to focus the agent's scope.
2. **Analyzing:** Monitor live progress as the five specialist micro-experts execute in parallel, with individual expandable progress cards and a streaming reasoning summary.
3. **Summary:** Review the full set of recommended Affected Items, each with a suggested action (Release, Revise, or Obsolete) and an explicit AI-generated justification. Filter and toggle individual recommendations before committing.
4. **Applying:** Apply selected changes back to the Change Request in a single transaction. A success confirmation and direct link back to the record in Aras Innovator are provided. Use



Process Another to start a new analysis cycle without re-authenticating.

Automated Compliance Archiving

When recommendations are accepted, they are written to the native Unified Change Management (UCM) Affected Items grid. The rationale for each committed item is automatically recorded in the Aras Secure Social Visual Collaboration (SSVC) comment thread on the Change Request, creating a durable audit trail.

Agent-Initiated Clarification

When the agent detects ambiguity, such as multiple historical Change Orders with divergent resolutions, it presents clickable suggestion chips directly in the Summary interface. Engineers can respond to these clarifications in a single click, guiding the agent toward higher-confidence recommendations without re-running the full analysis.

Deployment & Delivery Model

The agent is available as a software entitlement under the Aras InnovatorEdge AI Task Agents subscription model.

Software Prerequisites & Version Compatibility

- **Platform Dependencies:** Requires that Aras Unified Change Management (UCM) be installed and active as a baseline prerequisite before configuration.
- **Subscription Alignment:** Requires active Aras Innovator, Aras InnovatorEdge API, and Aras InnovatorEdge AI licenses.



- **Retro-Compatibility:** Fully compatible with any supported version of Aras Innovator running standard UCM configurations.

Professional Services and Configuration

To ensure that agent evaluations align with your unique data properties, custom workflows, and corporate configuration rules, the Impact Analysis AI Agent is delivered as a modular, configurable package. Deployment and customer-specific environment adaptations are fulfilled through a separately contracted Aras professional services engagement.

2026 June 23 - Edge AI Agents - Smart Variants Agent v.1.0.2 - Release Notes

Smart Variants Agent commercial release 1.0.2 is now live!

The business features and benefits are as follows:

- **Edge AI Legal Terms:** The Product Specific Licensing Terms for Aras InnovatorEdge AI are now available on the Privacy Notice and Legal Terms page.

2026 June 23 - Edge AI Agents - BOM Import Agent v.1.0.2 - Release Notes

BOM Import Agent commercial release 1.0.2 is now live!

The business features and benefits are as follows:

- **Parallel import support:** The agent now runs reliably across multiple browser tabs simultaneously.
- **Edge AI Legal Terms:** The Product Specific Licensing Terms for Aras InnovatorEdge AI are now available on the Privacy Notice and Legal Terms page.

2026 June 23 - Edge AI Agents - Requirements Ingestion Agent v.1.0.3 - Release Notes



Requirements Ingestion Agent commercial release 1.0.3 is now live!

The business features and benefits are as follows:

- **Parallel Session Reliability:** You can now run Requirements Ingestion imports in parallel across multiple browser tabs.
- **Edge AI Legal Terms:** The Product Specific Licensing Terms for Aras InnovatorEdge AI are now available on the Privacy Notice and Legal Terms page.

2026 April 28 - Edge AI Agents - Smart Variants Agent v.1.0.1 - Release Notes

Smart Variants Agent commercial release 1.0.1 is now live!

The business features and benefits are as follows:

- **Performance Optimization:** Reduction in the time required to analyze product configurations.
- **Enhanced Logic Extraction:** Improved co-occurrence analysis accuracy within the AI Agents Service.

Enhancements

Improved Extraction Engine

This update introduces an optimized engine for the AI Agents Service, specifically tuned to handle the complex co-occurrence analysis required for variant rule extraction.

Benefits: Decreases the engineer's wait time when invoking the "Analyze" action on multi-configuration product structures.

Example use case: An engineer managing a modular industrial machinery line can analyze dozens



of valid equipment setups to automatically identify mutually exclusive sensor packages and mandatory power supply components across different product models.

2026 March 28 - Edge AI Agents - BOM Import Agent v.1.0.1 - Release Notes

BOM Import Agent commercial release 1.0.1 is now live!

The business features and benefits are as follows:

- **Flexible Data Ingestion:** Added support for common engineering spreadsheet layouts beyond single-column formats.
 - **Process Transparency:** Direct capture of raw system errors during the Aras Innovator commitment phase.
 - **Graceful Resource Management:** Improved control over background parsing jobs to ensure system resources are freed upon user cancellation.
-

Enhancements

Support for Stair-Step BOM Hierarchies

The agent now supports "stair-step" (staircase) structural leveling in Excel files, where hierarchy is indicated by column indentation rather than a single level-index column.

Benefits: Expands the variety of supplier BOM formats that can be imported without manual file reformatting.

Example use case: A Procurement Manager can now import a legacy supplier spreadsheet where



assembly levels are represented by shifted columns (staircase format) directly into Aras Innovator.

Enhanced Part Identification Engine

Implementation of an upgraded analysis engine to improve the throughput and accuracy of part metadata extraction from engineering spreadsheets.

Benefits: Increases the speed of the initial parsing phase for large-scale Bill of Materials.

Example use case: Ingesting a master BOM with thousands of rows will result in faster classification of part numbers, names, and quantities.

Bug Fixes

Graceful Job Cancellation - IH-12953

Fixed an issue where the "Cancel" button would return the user to the home page but allow the background parsing job to continue running; background tasks are now stopped immediately upon cancellation.

Readable Validation Errors - IH-12446

Resolved a display bug where raw Innovator length violations (e.g., Part names exceeding 64 characters) were hidden; clear descriptive messages now appear in the UI for remediation.

Dialog Overflow Handling - IH-16174

Fixed a display issue where long reference designators or part identifiers caused the Import Progress dialog to overflow its container width.



2026 April 28 - Edge AI Agents - Requirements Ingestion Agent v.1.0.2 - Release Notes

Requirements Ingestion Agent commercial release 1.0.2 is now live!

The business features and benefits are as follows:

- **Reliability & Performance:** Improvements to extraction speed and accuracy through an upgraded analysis engine.
 - **Data Integrity Safeguards:** New navigation guards prevent accidental progress loss during active ingestion cycles.
 - **Actionable Feedback:** Enhanced error processing transforms raw system technicalities into human-readable instructions.
-

Enhancements

Upgraded Extraction Engine

This release introduces an enhanced backend extraction engine designed to increase the precision of requirement statement identification within complex PDF documentation.

Benefits: Improves the reliability of automated parsing, reducing the manual correction effort during the review phase.

Example use case: An engineer processing a high-density technical specification will see fewer misclassified text blocks and more accurate requirement boundaries.

Import Navigation Guards



To prevent data loss, the system now implements a navigation guard that monitors the browser state during active imports.

Benefits: Protects the engineer's progress by preventing accidental page reloads or tab closures from discarding uncommitted data.

Example use case: If a user accidentally hits the browser's "Back" button or F5 during a large document import, the application will trigger a warning to confirm the action.

User-Friendly Aras Innovator Error Reporting

This update replaces raw OData system errors with descriptive, actionable messages when a connection or system extension issue occurs.

Benefits: Allows users to troubleshoot integration issues immediately without requiring administrator intervention to read logs.

Example use case: If an Aras Innovator instance is missing the required Requirements Engineering extension, the UI will clearly state the missing dependency instead of returning an empty response.

Bug Fixes

Post-Import Workflow Continuity - IH-16266

Resolved a 404 error that occurred when users attempted to start a new ingestion cycle by clicking the "Process another" button immediately following a successful import.

Job Stability with Custom Instructions - IH-16400



Fixed a "Task Canceled" exception that caused ingestion jobs to fail when complex custom parsing instructions were applied to large-scale PDF files.

Import Summary UI Alignment - IH-16174

Corrected a display bug where long property names or descriptions in the import result dialog would exceed the width of the UI container, causing layout distortion.

2026 April 14 - Edge AI Agents - Smart Variants Agent 1.0.0 - Release Announcement

Smart Variants Agent release 1.0.0 is now live!

Overview

Aras is pleased to announce the general availability of the Smart Variants Agent v1.0.0, delivered via the InnovatorEdge AI framework.

The agent ingests product Bill of Materials (BOM) data from Aras Innovator, analyzes configuration relationships across variants, and assists authors in creating a Variant Management (VM) model.

Why This Matters

Defining and maintaining variant structures across complex product families historically requires massive manual effort.

The Smart Variants Agent reduces this friction by shifting human effort from manual transcription to high-value validation. By automating the derivation of rules from existing engineering BOMs, organizations can faster model variability structures, usage conditions, and variable components while maintaining a strict digital thread.



What is included in this release

End-to-End BOM Analysis and Review

The agent provides a comprehensive workflow to move from raw engineering data to a verified variant model:

- **Product Selection:** A paginated list allows users to search for products by **name**, **item number**, or **description**.
- **Deep BOM Expansion:** The "Analyze" action loads a deep BOM expansion for the selected product and sends a normalized graph to the AI Agents Service.
- **Deterministic Analysis:** The agent derives variant rules through deterministic co-occurrence analysis.
- **LLM Enrichment:** Large Language Models are used to enrich the variant model, ensuring a robust import payload.

Human-in-the-Loop Review

Before any data is committed to the system of record, engineers must perform a review of the analysis results:

- **150% BOM Tree:** A merged structural view with the first two levels expanded by default.
- **Rule Tables:** Dedicated, searchable tables for **Coupled (and)** rules and **Either/Or (XOR)** rules.
- **Dependency Matrix:** An interactive matrix for **Implies** and **Exclude** rules with an expand/collapse toolbar for easier navigation.

Deployment and Delivery

Delivery Model



Task Agents are available as a software entitlement under the **Aras InnovatorEdge AI Task Agents** subscription. Each agent is delivered as a configurable package designed to be adapted for customer-specific requirements through a separately contracted professional services engagement.

Software Prerequisites

- **Subscriptions:** Requires active Aras Innovator, Aras InnovatorEdge API, and Aras InnovatorEdge AI subscriptions.
- **Version Compatibility:** any supported version of Aras Innovator (retro-compatibility)
- **Applications:** Aras Variant Management

Known Behaviors and Limitations

- **BOM Depth:** The Edge App uses a fixed OData expansion depth reaching up to **nine levels** inclusive. BOMs exceeding this depth may result in incomplete analysis.
- **UI Editing Constraints:** The review UI is currently read-only for rules and the 150% BOM; adjustments must be made to source data in Innovator and re-analyzed to change outcomes.
- **Partial Import Failure:** If an error occurs in a later import phase (e.g., Phase 3), data from earlier phases may already be committed to Innovator. The UI provides error messages for remediation, such as prefix collisions.
- **Single-Configuration Products:** The agent requires multiple BOM configurations to produce meaningful variant rules; products with only one configuration will not infer co-occurrence relationships.
- **Search Limitations:** Search field inputs are capped at **100 characters** to prevent fetch failures.
- **LLM Transparency:** The LLM-enriched model used for payload construction is internal and not reviewable in the UI; users review the deterministic rules only.

What Is Coming Next

The initial Task Agent package is the first release in an expanding catalog. Additional agents and features planned for 2026 include:



- **Change Impact Analysis Agent:** AI-assisted analysis of change propagation across the BOM, identifying downstream impacts on parts, documents, and processes.
- **Priority Expansion:** Further agents will be prioritized based on customer advisory input and pilot outcomes.

2026 April 9 - Edge AI Agents - Requirements Ingestion Agent v.1.0.1 - Release Notes

Requirements Ingestion Agent commercial release 1.0.1 is now live!

The business features and benefits are as follows:

- **Requirements Import Acceleration:** Reduction in time required to move requirements into Aras Innovator through parallelization.
- **Enhanced Review Experience:** Full-screen validation interface with persistent search and cross-tab session support.
- **System Connectivity Visibility:** Clear indicators of the target Aras Innovator instance currently connected.
- **Integrated Compliance & Support:** Direct access to Aras documentation, legal terms, and support resources.

Enhancements

Accelerated Requirement Import Performance

This release introduces batching and parallelization to the Requirements Document import process. By processing 200 items per request, the agent now handles the creation, updating, and mapping of requirements to Document Items significantly faster.



Benefits: Reduces the engineering wait time for large documents.

Example use case: An engineer uploading a large technical rulebook can now complete the entire import cycle in a single session without risk of gateway timeouts.

Improved Validation and Review Interface

The Review phase UI has been extended to a full-screen layout. This update includes a search field that remains visible even while scrolling through long requirement lists.

Benefits: This helps maintain search context and provides a spacious environment for detail-oriented data review.

Example use case: An engineer reviewing hundreds of extracted requirements can scroll through the entire document while keeping the search bar accessible to filter and verify specific items.

Automatic Instruction Cleanup and Multi-Tab Support

The agent now automatically cleans up the Custom parsing instructions field when a user logs out or re-logins.

Simultaneously, instructions are synchronized across multiple browser tabs during an active session, allowing users to modify logic in one tab and see it reflected in others.

Benefits: Enhances security by ensuring old parsing logic does not persist across different user sessions while supporting flexible browser-based workflows within a single session.

Example use cases: A user modifies a custom instruction in one browser tab to better segment requirement text and sees the updated instruction automatically applied when viewing the import page in a second tab.

Connected Instance Visibility



The user interface now explicitly displays the URL of the Aras Innovator instance to which the agent is currently connected.

Information regarding the target Innovator environment is retrieved directly from the Edge Application configuration.

Benefits: Provides immediate clarity for users working in multi-environment landscapes, ensuring requirements are imported to the intended system of record.

Native Help, Support, and Legal Integration

A new Help menu (accessible via the ? icon) provides a dropdown with direct links to official Aras InnovatorEdge documentation and the subscriber support portal. Additionally, links to Aras Legal Terms and Privacy Policies are now persistently visible across all agent pages.

Benefits: Facilitates faster self-service and issue resolution while ensuring users have constant access to governed usage policies.

Bug Fixes

Changes made in preview not imported - IH-12917

Fixed a critical issue where modifications made during the preview phase (such as requirement deletions or text updates) were discarded if a user switched browser tabs during the setup phase. The agent now correctly imports the updated version of requirements and the Requirement Document.

Timeout while importing to Innovator - IH-12667



Resolved a 504 Gateway Timeout error that occurred when attempting to import large requirement datasets.

Improper handling of silent job failures - IH-11854

Fixed an issue where the agent incorrectly reported a "completed successfully" status with null results when an internal error occurred during job execution. The agent now correctly identifies a Failed status and displays the relevant error to the user. A **Try Again** button is shown redirecting back to the Upload page.

Incorrect total requirement count display - IH-11801

Resolved a display bug where the total requirement count incorrectly showed the initial parsed number after a successful import, failing to reflect requirements deleted by the user during the Preview phase.

2026 April 9 - Edge AI Agents - BOM Import Agent v.1.0.0 - Release Announcement

Aras InnovatorEdge AI Agents – BOM Import Agent release 1.0.0 is now live!

Overview

Aras is pleased to introduce the **BOM Import Edge AI Agent** as part of the first generation of AI agents delivered through InnovatorEdge AI.

This Task Assistant Agent enriches the digital thread by transforming Bill of Materials (BOM) data contained in engineering spreadsheets into structured data managed inside Aras Innovator. The agent operates fully within the governed Aras Innovator system of record and requires human



validation and review before any items are created in the database.

What Is Included in This Release

End-to-End BOM Normalization

The agent extracts part metadata and hierarchical relationships from Excel source documents (.xlsx file type). It segments content into candidate items including Parts, Part BOMs, and associated properties, creating one Root Part (0-level) per ingestion session and recursive Part BOM relationships linking parents to children.

Agent Instructions (Custom Mapping Rules)

The BOM Import Agent supports global and custom instructions to define how spreadsheet content is interpreted:

- **Natural Language Mapping:** Users can describe column mappings to ensure data aligns with specific Aras properties, such as mapping "Assembly Identification" to the "Name" Part property.
- **Hierarchy Detection:** The agent can interpret structure via single-column level numbering (1, 1.1) or stair-step formatting.
- **Uniqueness & Merging:** The algorithm automatically identifies the primary (unique) key column to merge and consolidate duplicate parts within the uploaded structure.

Validation Preview and Search

After extraction, the agent presents a candidate structure for review:

- **Hierarchical Tree Preview:** View the assembled BOM in a structured grid.
- **Full-Text Search:** Filter and highlight text across all extracted columns to verify critical data.



- **Property Editing:** Modify editable Part and BOM properties (Name, Description, Qty) directly in the preview before import.

Image Extraction

The agent identifies part images embedded in the source file and prepares them for import as Thumbnails linked to the corresponding Part items in Aras Innovator.

Why This Matters

The key outcome of the BOM Import Agent is structured BOMs managed as governed objects. By separating content from source file formatting, teams can operate on engineering data rather than manually transcribing spreadsheet rows. This shifts human effort away from tedious transcription and toward validation and engineering intent, where domain expertise is most valuable.

Deployment

The BOM Import Agent is delivered through the InnovatorEdge AI platform and operates using the Edge API, which provides the secure execution layer for AI services interacting with Aras Innovator.

- **Connectivity:** The Edge API manages communication between the agent runtime and the system of record to ensure all operations occur within governed platform boundaries.
- **Prerequisites:** The agent depends on the Aras Product Engineering (PE) application for the data model and governance framework used to manage Part and Part BOM objects.
- **Supported Versions:** Release 33 (or latest compatible).

Known Behaviors and Limitations

- **Supported Formats:** Usage is currently restricted to Excel (.xlsx) files; support for PDF, CSV, or Word is currently out of scope.
- **File Size:** Supports a single file upload per session with a maximum size of 10 MB.



- **Creation Policy:** The agent creates new Parts only and does not currently support matching against or updating existing parts in the system.
- **Relationship Scope:** Exclusively supports Part BOM relationships; properties relying on different relationships (e.g., Cost or Weight) are not supported.
- **Hierarchy Editing:** Users cannot currently re-attach parts or change parent-child structures via drag-and-drop within the preview grid.
- **Metadata Flags:** Required properties may not be highlighted in the UI due to current Edge API metadata reporting limitations.

2026 March 10 - Edge AI Agents - Requirements Ingestion Agent v.1.0.0 - Release Announcement

Requirements Ingestion Agent release 1.0.0 is now live!

Overview

Aras is pleased to announce the general availability of the **Requirements Ingestion Agent**, the first purpose-built Task agent within the **InnovatorEdge AI** framework.

While conversational agents like the NLA Agent focus on answering questions, Task Agents take active engineering action to ingest, normalize, and structure product data.

The Requirements Ingestion Agent automates high-friction PLM workflows by transforming unstructured requirements trapped in PDF documents into governed items directly within Aras Innovator Requirements Engineering.

What Is Included in This Release

Requirements Ingestion Engine

The agent enables you to ingest requirements from external PDF source documents and convert



them into structured requirement objects within the system of record.

This eliminates manual transcription errors and ensures that every requirement remains traceable from the source document to the PLM record from the moment it enters the digital thread.

Why This Matters

Manual transcription of engineering specifications has historically been a significant bottleneck in product development.

The Requirements Ingestion Agent removes this barrier by shifting human effort from tedious data entry to high-value validation and design intent.

By automating the normalization of incoming document data, organizations can accelerate their digital thread while maintaining strict requirements quality and governance.

Deployment

The Requirements Ingestion Agent is available for both **Full SaaS** and **Hybrid** deployments. For Hybrid models, the Aras Innovator instance remains on-premises and connects to the Edge AI service through the Edge API layer via outbound HTTPS (port 443).

Professional Services and Configuration

To ensure each deployment is grounded in a customer's unique data model, workflow conventions, and business rules, the Requirements Ingestion Agent is delivered as a configurable package. Deployment and adaptation for customer-specific environments require a separate Aras professional services engagement.



Software Prerequisites

Task Agents operate within the InnovatorEdge AI ecosystem and require the following:

- **Applications:** Aras Requirements Engineering.
- **Innovator Version:** any supported Aras Innovator version
- **Requirements Engineering Version:** Release 33 (14.0.2).
- **Subscriptions:** Active Aras Innovator, InnovatorEdge API, and Aras InnovatorEdge AI subscriptions.
- **Vault Access:** Proper configuration of Edge API file vault settings is required to handle graphics and file uploads during the import phase.

Known Behaviors and Limitations

- **Supported Formats:** Validation is currently restricted to PDF documents; other formats are currently out of scope.
- **File Size:** The agent supports a single file upload per session with a maximum size of 10 MB.
- **Table Editing:** Extracted tabular content is preserved and associated with requirements, but these tables are not editable during the review phase prior to import.
- **Structural Dependency:** The quality of extraction is highly dependent on the structural conventions (such as numbering and headings) used in the source document.

What Is Coming Next

This release is the first in an expanding catalog of agents designed to optimize end-to-end PLM business processes. Future updates planned for 2026 include:

- **BOM Import Agent:** Normalizes incoming bill of materials data from supplier and partner formats and maps it to standard Aras Innovator structures.
 - **Smart Variants Agent:** Analyzes existing BOMs and supporting product documentation to produce variant models and configurations within Aras Innovator.
-



AI Search



AI Search - Business Features and Benefits

What is AI Search?

AI Search is a semantic search capability built into Aras Innovator. It provides natural language search across Innovator items and the files attached to them.

Administrators define which ItemTypes and file types are searchable. AI Search then makes that scope available to every user through a single search experience, with results filtered by each user's existing Innovator permissions.

Why This Matters

Traditional Innovator search matches on exact keywords and requires the user to know the right terms, the right ItemType, or the right filter combination. Engineers, quality reviewers, and program managers often know what they are looking for in plain language, not in the vocabulary the data model uses.

AI Search closes that gap. A user can search using everyday language, and AI Search returns the items and documents that are conceptually relevant, not only the ones containing an exact keyword match.

Key Business Benefits

- **Faster discovery.** Users find relevant parts, documents, and records without needing to know exact item numbers, property names, or filter syntax.
- **Administrator-controlled scope.** Administrators decide which ItemTypes and file types are included. Nothing is indexed unless an administrator has explicitly configured it.
- **No bypass of existing permissions.** AI Search results respect the Innovator permission model. A user does not see a result they would not already be authorized to see in standard Innovator search.
- **Works alongside file content.** When file handling is enabled for an ItemType, AI Search can also surface relevant files attached to items, not only item properties.
- **Available on SaaS and on-premises/private-cloud.** AI Search supports full on-premises deployment, in addition to SaaS.



Target Audiences

- **Engineers and designers** searching for parts, specifications, or prior work without knowing exact item numbers or classification paths.
- **Quality and compliance reviewers** locating documents and records by topic rather than by filename.
- **Innovator administrators** who configure and maintain what is searchable for their organization.
- **External developers and integrators** who consume the AI Search API through Edge API to build their own search experiences or agent tools.

Use Case Example

An engineer starting a new design wants to check whether a similar part already exists before creating a new one. Rather than searching by exact item number or browsing through classification trees, the engineer types a plain language description into AI Search. AI Search returns a ranked list of items that are conceptually related, each linking directly back to the source record in Innovator, filtered to only the items the engineer is already authorized to see.



Get Started with AI Search

Aras InnovatorEdge Account

AI Search is part of the Edge platform. To get access, your organization requires an active Aras InnovatorEdge AI Subscription (AS-1242).

1. The designated tenant owner in your organization follows **Request an Edge account** to get access.
2. Other people in your organization will contact their tenant owner to request user access.
3. After onboarding, you will receive your login credentials in your Edge account (tenant).

Prerequisites

Before enabling AI Search, confirm the following:

- You have administrator access to the target Innovator instance.
- You have decided your deployment model: SaaS or OnPrem. The steps below cover enabling each one.
- Your Innovator instance is version 14.0.15 or later.
- Aras InnovatorEdge AI Subscription: AS-1242

Enable AI Search

On SaaS

For Aras-hosted Innovator SaaS instances, the AI Search service components are deployed and operated by Aras. Enabling AI Search requires adding two files to your Work repository and running your deployment pipeline. Follow **Installation of AI Search for SaaS** for the full procedure.

OnPrem/Private Cloud

AI Search can be deployed entirely on your own infrastructure using Docker containers, for organizations with strict data residency or network isolation requirements. Follow **Installation of AI Search for OnPrem/Private Cloud** for the full procedure, including the Docker Compose stack, required environment variables, certificate setup, and the Innovator-side package.



Through Edge API

External developers and integrators consuming AI Search results programmatically should see **Consuming the Semantic Search API Endpoint** in the AI Search User Guide for authentication and endpoint details.

Configure Your First Indexing Scope

Once AI Search is enabled and the Innovator-side package is installed, an administrator configures what is searchable.

1. Open the AI Search page in Innovator.
2. Select the gear icon in the top right corner of the page to open the AI Search Configurator. This icon is visible only to Innovator administrators.
3. Type an ItemType name in the **Select an ItemType...** field to find the type you want to index.
4. Select **Add** to expand the ItemType row, then configure it by adding at least one property, enabling file handling with file types, or enabling file properties in the File Properties section.
5. Select **Apply**.

This is enough to confirm AI Search is working end to end. For the full configuration reference, including title and subtitle templates, schema import and export, and monitoring indexing status, see the **AI Search Configurator - Admin Guide**.



Installation of AI Search

Disclaimer

This page covers deploying AI Search on a Docker host using the production Docker Compose stack. It is intended for infrastructure engineers and system administrators responsible for deploying containerized services. For Aras-hosted Innovator SaaS instances, see Installation of AI Search for SaaS instead.

What you are deploying

AI Search runs as three long-running services (ais-api, ais-indexer-worker, ais-parser-worker) plus two one-shot init containers (ais-db-migrations, parser-staging-init) that run once at startup and then exit. These connect to external services you must provision before installation:

```
AIS containers
  - Aras Innovator      (HTTP + OAuth + Vault)
  - Temporal            (workflow orchestration, port 7233)
  - SQL Server          (inside the existing Innovator database, port 1433)
  - RabbitMQ / Azure Service Bus (CDC events via Edge API, AMQP 1.0)
  - Embeddings provider (OpenAI-compatible API, port 443)
```

None of these services are bundled with this stack; all must be provisioned before installation. The prerequisites section below documents what must be in place.

Prepare the deployment files

This guide is the source of the deployment files. There is no separate archive or repository. You create a working directory on the Docker host and populate it by copying the file contents shown in this document.

Target layout:



```

ai-search/
â"œâ"€â"€ docker-compose.yaml # Copy from the "docker-compose.yaml" section below
â"œâ"€â"€ .env # Copy from the ".env" section below, then fill in your values
(Â§3.2)
â"€â"€â"€ certs/ # Directory for the OAuth certificate (see Â§3.3)

```

Steps:

1. Create the working directory:

```

mkdir -p ai-search/certs
cd ai-search

```

2. Create a file named docker-compose.yaml in this directory and paste the contents from the docker-compose.yaml section below.
3. Create a file named .env in this directory and paste the contents from the .env section below, then fill in your values as described in Â§3.2.
4. Leave certs/ empty. You will place the OAuth certificate there in Â§3.3.

Run all subsequent commands from the ai-search/ directory.

docker-compose.yaml



```

# Production AI Search stack â€” standalone, image-based.
# All services use published Docker Hub images; no source builds.
#
# Prerequisites:
# - Copy .env.example to .env next to this file and fill in all required values.
# - Place the OAuth certificate file in the certificate directory (default: ./certs/).
#   The file must be named <INNOVATOR_OAUTH_CLIENT_ID>.pfx (e.g. IndexerWorker.pfx).
#   Override the directory with CERTS_HOST_DIR in .env if certs live elsewhere.
# - Innovator Server and its SQL Server are reachable from the host via the URLs in .env.
# - External RabbitMQ (from the Edge API/CDC stack) is reachable and AMQP_CONNECTION_STRING is
  set in .env.
#   The broker must have the rabbitmq_amqp1_0 plugin enabled (AI Search uses AMQP 1.0).
# - A Temporal frontend endpoint must be reachable at TEMPORAL_HOST (see .env.example).
#   This file contains only AIS services â€” Temporal itself is an external dependency.
# - Vault Server must be reachable from the indexer worker containers (see .env.example).
#   Ensure the vault hostname is resolvable from Docker containers via extra_hosts, DNS, or
#   host.docker.internal as appropriate for your deployment.
#
# IMAGE_TAG in .env controls which version of all AI Search images is deployed.
#
# Minimum recommended hardware:
#   CPU: 4 vCPUs (ais-parser-worker uses all available cores for ML inference)
#   RAM: 8 GB (parser-worker loads ~4 GB of ML models; remainder for API, indexer, and
  Temporal)
#   Disk: 20 GB for the parser-staging volume (transient workspace during file indexing;
  actual size depends on the number and size of files indexed concurrently)

x-logging: &default-logging
  driver: json-file
  options:
    max-size: "50m"
    max-file: "5"

name: aras-ai-search

services:

  # -----
  # One-shot init containers
  # -----
  ais-db-migrations:
    image: ${IMAGE_REGISTRY:?IMAGE_REGISTRY is required}/db-migrations:${IMAGE_TAG:?IMAGE_TAG is
  required}
    restart: "no"

```



```

environment:
  ASPNETCORE_ENVIRONMENT: Production
  AisDb__SchemaInstallConnectionString: "${AIS_DB_SCHEMA_CONNECTION_STRING:?
AIS_DB_SCHEMA_CONNECTION_STRING is required}"
  extra_hosts:
    - "${DOCKER_HOST_GATEWAY:-host.docker.internal}:host-gateway"
  networks:
    - ais-network

parser-staging-init:
  image: busybox:1.36.1
  # setgid 2770 + GID 5000 owner: new files inherit group ownership; workers get GID 5000 via
group_add.
  command: ["sh", "-c", "mkdir -p /var/lib/aras/parser-staging && chown root:5000
/var/lib/aras/parser-staging && chmod 2770 /var/lib/aras/parser-staging"]
  volumes:
    - parser_staging:/var/lib/aras/parser-staging
  restart: "no"
  networks:
    - ais-network

# -----
# AI Search API (SPA host + JSON API)
# -----
ais-api:
  image: ${IMAGE_REGISTRY:?IMAGE_REGISTRY is required}/api:${IMAGE_TAG:?IMAGE_TAG is required}
  depends_on:
    ais-db-migrations:
      condition: service_completed_successfully
  environment:
    ASPNETCORE_ENVIRONMENT: Production
    ASPNETCORE_URLS: http://+:8080
    Innovator__InstanceId: "${INNOVATOR_INSTANCE_ID:?INNOVATOR_INSTANCE_ID is required}"
    Innovator__InnovatorServerUrl: "${INNOVATOR_SERVER_URL:?INNOVATOR_SERVER_URL is required}"
    AisDb__DataConnectionString: "${AIS_DB_DATA_CONNECTION_STRING:?
AIS_DB_DATA_CONNECTION_STRING is required}"
    AisDb__SearchConnectionString: "${AIS_DB_SEARCH_CONNECTION_STRING:?
AIS_DB_SEARCH_CONNECTION_STRING is required}"
    Temporal__TargetHost: "${TEMPORAL_HOST:?TEMPORAL_HOST is required}"
    Temporal__Namespace: "${TEMPORAL_NAMESPACE:-default}"
    Temporal__Credentials__ApiKey: "${TEMPORAL_API_KEY:-}"
    Amqp__ConnectionString: "${AMQP_CONNECTION_STRING:?AMQP_CONNECTION_STRING is required}"
    Cdc__ConfiguratorUrl: "${CDC_CONFIGURATOR_URL:?CDC_CONFIGURATOR_URL is required}"
    Cdc__Origin: "${CDC_ORIGIN:-aisearch}"

```



```

    Cdc__InstanceName: "${CDC_INSTANCE_NAME:-default}"
    Embeddings__Endpoint: "${EMBEDDINGS_ENDPOINT:?EMBEDDINGS_ENDPOINT is required}"
    Embeddings__ApiKey: "${EMBEDDINGS_API_KEY:?EMBEDDINGS_API_KEY is required}"
    Embeddings__ModelId: "${EMBEDDINGS_MODEL_ID:-text-embedding-3-small}"
    VectorSearchSettings__VectorDimension: "${VECTOR_DIMENSION:-1536}"
ports:
  # Bound to localhost â€” place a TLS-terminating reverse proxy (nginx, Traefik, etc.) in
front for external access.
  # Health endpoints (no auth required):
  #   GET /health/live â€” liveness (instant, no external checks); suitable for K8s
liveness probe
  #   GET /health/ready â€” readiness (checks SQL, Temporal, CDC, Embeddings); suitable for
load-balancer
  - "127.0.0.1:5080:8080"
restart: unless-stopped
extra_hosts:
  - "${DOCKER_HOST_GATEWAY:-host.docker.internal}:host-gateway"
logging: *default-logging
networks:
  - ais-network

# -----
# AI Search Temporal Worker (.NET â€” indexing workflows + incremental consumer)
# -----
ais-indexer-worker:
  image: ${IMAGE_REGISTRY:?IMAGE_REGISTRY is required}/indexer-worker:${IMAGE_TAG:?IMAGE_TAG
is required}
  depends_on:
    ais-db-migrations:
      condition: service_completed_successfully
    parser-staging-init:
      condition: service_completed_successfully
  environment:
    ASPNETCORE_ENVIRONMENT: Production
    Innovator__InstanceId: "${INNOVATOR_INSTANCE_ID:?INNOVATOR_INSTANCE_ID is required}"
    Innovator__InnovatorServerUrl: "${INNOVATOR_SERVER_URL:?INNOVATOR_SERVER_URL is required}"
    Innovator__OAuthServerUrl: "${INNOVATOR_OAUTH_URL:?INNOVATOR_OAUTH_URL is required}"
    Innovator__ClientId: "${INNOVATOR_OAUTH_CLIENT_ID:-IndexerWorker}"
    Innovator__CertificatePath: "/Container/Certificates/${INNOVATOR_OAUTH_CLIENT_ID:-
IndexerWorker}.pfx"
    Innovator__CertificatePassword: "${ARAS_OAUTH_CERTIFICATE_PASSWORD}"
    Innovator__ServiceUserName: "${INNOVATOR_SERVICE_USER:-vadmin}"
    AisDb__DataConnectionString: "${AIS_DB_DATA_CONNECTION_STRING:?
AIS_DB_DATA_CONNECTION_STRING is required}"

```



```

Temporal__TargetHost: "${TEMPORAL_HOST:?TEMPORAL_HOST is required}"
Temporal__Namespace: "${TEMPORAL_NAMESPACE:-default}"
Temporal__Credentials__ApiKey: "${TEMPORAL_API_KEY:-}"
Amqp__ConnectionString: "${AMQP_CONNECTION_STRING:?AMQP_CONNECTION_STRING is required}"
IncrementalConsumer__StartupDelaySeconds: "5"
IncrementalConsumer__HeartbeatSeconds: "60"
IncrementalConsumer__MaxProcessingAttempts: "5"
IncrementalConsumer__RetryBackoffSeconds: "2"
ParserStaging__Root: "/var/lib/aras/parser-staging"
ParserStaging__MaxFileBytes: "536870912"
ParserStaging__TransferBufferBytes: "16777216"
Embeddings__Endpoint: "${EMBEDDINGS_ENDPOINT:?EMBEDDINGS_ENDPOINT is required}"
Embeddings__ApiKey: "${EMBEDDINGS_API_KEY:?EMBEDDINGS_API_KEY is required}"
Embeddings__ModelId: "${EMBEDDINGS_MODEL_ID:-text-embedding-3-small}"
VectorSearchSettings__VectorDimension: "${VECTOR_DIMENSION:-1536}"
FileParsing__MaxTokens: "${FILE_PARSING_MAX_TOKENS:-1500}"
FileParsing__MergePeers: "${FILE_PARSING_MERGE_PEERS:-true}"
FileParsing__DoOcr: "${FILE_PARSING_DO_OCR:-false}"
FileParsing__TableMode: "${FILE_PARSING_TABLE_MODE:-accurate}"
FileParsing__DoTableStructure: "${FILE_PARSING_DO_TABLE_STRUCTURE:-true}"
Temporal__ActivityPolicies__CheckFileProcessedStartToCloseSeconds: "10"
Temporal__ActivityPolicies__DownloadFileStartToCloseSeconds: "300"
Temporal__ActivityPolicies__DownloadFileHeartbeatSeconds: "120"
Temporal__ActivityPolicies__ParseAndStageChunksStartToCloseSeconds: "900"
Temporal__ActivityPolicies__ParseAndStageChunksHeartbeatSeconds: "300"
Temporal__ActivityPolicies__PersistStagedChunksStartToCloseSeconds: "120"
Temporal__ActivityPolicies__EmbedFileChunksStartToCloseSeconds: "300"
Temporal__ActivityPolicies__EmbedFileChunksHeartbeatSeconds: "120"
Temporal__ActivityPolicies__CleanupStagedFileStartToCloseSeconds: "30"
volumes:
  - "${CERTS_HOST_DIR:-./certs}:/Container/Certificates:ro"
  - parser_staging:/var/lib/aras/parser-staging:rw
# ais-staging (GID 5000): write access to the parser-staging shared volume.
group_add:
  - "5000"
restart: unless-stopped
stop_grace_period: 60s
extra_hosts:
  - "${DOCKER_HOST_GATEWAY:-host.docker.internal}:host-gateway"
logging: *default-logging
networks:
  - ais-network

```

```
# -----
```



```

# File Parser Worker (Python â€” docling-based file parsing)
# -----
ais-parser-worker:
  image: ${IMAGE_REGISTRY:?IMAGE_REGISTRY is required}/parser-worker:${IMAGE_TAG:?IMAGE_TAG is required}
  depends_on:
    ais-db-migrations:
      condition: service_completed_successfully
    parser-staging-init:
      condition: service_completed_successfully
  environment:
    PARSER_WORKER__TEMPORAL__TARGET_HOST: "${TEMPORAL_HOST:?TEMPORAL_HOST is required}"
    PARSER_WORKER__TEMPORAL__NAMESPACE: "${TEMPORAL_NAMESPACE:-default}"
    PARSER_WORKER__TEMPORAL__INNOVATOR_INSTANCE_ID: "${INNOVATOR_INSTANCE_ID:?INNOVATOR_INSTANCE_ID is required}"
    PARSER_WORKER__TEMPORAL__API_KEY: "${TEMPORAL_API_KEY:-}"
    PARSER_WORKER__STAGING_ROOT: "/var/lib/aras/parser-staging"
    PARSER_WORKER__MAX_CONCURRENT_ACTIVITIES: "${PARSER_WORKER__MAX_CONCURRENT_ACTIVITIES:-1}"
  volumes:
    - parser_staging:/var/lib/aras/parser-staging:rw
# ais-staging (GID 5000): write access to the parser-staging shared volume.
group_add:
  - "5000"
restart: unless-stopped
stop_grace_period: 60s
extra_hosts:
  - "${DOCKER_HOST_GATEWAY:-host.docker.internal}:host-gateway"
logging: *default-logging
networks:
  - ais-network

volumes:
  parser_staging:

networks:
  ais-network:
    name: ais-network
    driver: bridge

```

.env



[illegible]

```
# Service account used by the indexer worker to perform Innovator operations.
# Default: vadmin
INNOVATOR_SERVICE_USER=vadmin

# Database
aisearch__database=

# SQL Server connection string for data read/write operations.
# Used by ais-api and ais-indexer-worker for general operations. (required)
#
# Copy one of the examples below and substitute your database name, user, and password:
#   External SQL Server:
#       Server=tcp:sql.company.com,1433;Database=innovator;User
#       Id=aisearch;Password=yourpassword;Encrypt=True;TrustServerCertificate=True;
#   SQL Server on the Docker host (Mac / Windows Docker Desktop / Linux):
#       Server=tcp:host.docker.internal,1433;Database=innovator;User
#       Id=aisearch;Password=yourpassword;Encrypt=True;TrustServerCertificate=True;
#       (Linux only: also set DOCKER_HOST_GATEWAY=host.docker.internal above)
AIS_DB_DATA_CONNECTION_STRING=

# SQL Server connection string for vector/full-text search queries.
# Can point to a read-only replica for scaling. (required)
AIS_DB_SEARCH_CONNECTION_STRING=

# SQL Server connection string for schema migrations.
# Needs CREATE TABLE / ALTER TABLE permissions. (required)
# In simple deployments all three strings can be set to the same value.
AIS_DB_SCHEMA_CONNECTION_STRING=

# Message bus (AMQP)
aisearch__amqp__url=

# AMQP 1.0 connection string to the RabbitMQ broker provided by the Edge API/CDC stack.
(required)
# This must point at the same broker that the Edge API exporter publishes CDC events to.
# Note: AI Search uses the AMQP 1.0 protocol (AmqpNetLite). The RabbitMQ broker must have
#       the rabbitmq_amqp1_0 plugin enabled.
#
# Exchange: the Edge API exporter creates and publishes to exchange "ams.exchange"
automatically.
#
#       AI Search does not need to create or configure this exchange.
```




```
# Currently only 1536 is supported. Default: 1536
VECTOR_DIMENSION=1536

# File parsing
FILE_PARSING=true

# Maximum tokens per parsed chunk. Default: 1500
FILE_PARSING_MAX_TOKENS=1500

# Merge peer chunks during parsing. Default: true
FILE_PARSING_MERGE_PEERS=true

# Enable OCR during parsing (slow; enable for scanned PDFs). Default: false
FILE_PARSING_DO_OCR=false

# Table detection mode: fast or accurate. Default: accurate
FILE_PARSING_TABLE_MODE=accurate

# Enable table structure recognition. Default: true
FILE_PARSING_DO_TABLE_STRUCTURE=true

# Parser worker
PARSER_WORKER=true

# Number of parallel file-parsing activities the parser worker runs simultaneously.
# Increase on hosts with more CPU cores and memory to speed up initial indexing. Default: 1
PARSER_WORKER_MAX_CONCURRENT_ACTIVITIES=1
```

Prerequisites

Before starting, verify that every item below is in place on the target host.

Docker

Component	Minimum version
Docker Engine	24.0
Docker Compose plugin	2.20



Check:

```
docker version          # Engine version
docker compose version  # Compose plugin version
```

Operating system

- **Linux (recommended for production):** any distribution that supports Docker Engine 24 or later (Ubuntu 22.04 LTS, RHEL 9, Debian 12, or later)
- **Windows 10 / 11:** Docker Desktop 4.25 or later with WSL 2 backend (test or pilot only)

Shell commands in this guide assume a bash-like shell. On Windows, run them from WSL 2, Git Bash, or a Linux jump host; native PowerShell equivalents differ (e.g. cp â†’ Copy-Item).

Aras Innovator

AI Search requires a running Innovator instance that is reachable from the Docker host:

Requirement	Details
Innovator version	14.0.15 or later
HTTP access	Innovator Server URL reachable from the Docker host
OAuth server	OAuth Server URL reachable from the Docker host
Vault access	Vault Server HTTP endpoint reachable from the AIS containers (required for file content indexing)
Service account	A dedicated service account (default: vadmin) with access to all ItemTypes that will be indexed

Important

Vault on the same Docker host: on Linux, host.docker.internal is not resolved automatically. Set DOCKER_HOST_GATEWAY=host.docker.internal in .env; the compose file injects it as extra_hosts so containers can reach the host. On Mac and Windows Docker Desktop this is automatic.

Important

Vault URL resolution: the vault hostname is obtained from Innovator's internal vault configuration at indexing time and is not set in AIS .env. Ensure the hostname returned by Innovator is resolvable from Docker containers via DNS, extra_hosts, or host.docker.internal as appropriate for your deployment.

Temporal

AI Search uses **Temporal** for workflow orchestration. A Temporal cluster must be running and reachable before deploying AIS. Temporal is **not** bundled with this stack.

Requirement	Details
Temporal version	1.24 or later
Frontend address	:7233 reachable from the Docker host
Namespace	A dedicated namespace is recommended (default: default)
Authentication	Temporal Cloud requires an API key —set it as TEMPORAL_API_KEY in .env. Self-hosted clusters without authentication can leave it empty.

Options:

- **Temporal Cloud:** use your cloud namespace frontend address (.tmpri.cloud:7233)
- **Self-hosted:** see the [Temporal self-hosting guide](#)

SQL Server

AI Search installs the [ais] schema and its tables inside the **Innovator database**; no separate database is created. The same SQL Server instance that hosts Innovator is used:

Requirement	Details
SQL Server version	SQL Server 2025 or later, required for the native vector type used by AI Search. Azure SQL Database with vector support is also supported. If you are on SQL Server 2019 or 2022, contact Aras for the compatibility upgrade path.
Connectivity	TCP port 1433 reachable from the Docker host



Permissions	See Section 2.3
-------------	-----------------

Edge API / CDC stack

AI Search receives real-time change events from Innovator via the Aras InnovatorEdge CDC pipeline, a separate Aras product that is a prerequisite for AI Search and must be installed and running before deploying AIS.

The following components are part of that stack:

Component	Role
edgeapi-eventing-configurator	Registers ItemTypes for SQL Server CDC tracking and provisions message broker queues. Must expose an HTTP endpoint reachable from the AIS containers (CDC_CONFIGURATOR_URL).
edgeapi-eventing-exporter	Streams CDC change events from SQL Server to the message broker. Must connect to the same SQL Server instance as Innovator.
Message broker	Either RabbitMQ or Azure Service Bus. AI Search uses the AMQP 1.0 protocol. For RabbitMQ, the rabbitmq_amqp1_0 plugin must be enabled.

Verify the exporter is connected and polling before starting AIS:

```
docker logs <edgeapi-exporter-container> --tail 5
# Expected: CDC reader mode: Normal, synchronized tables: N, pending: 0
```

If using RabbitMQ, verify the rabbitmq_amqp1_0 plugin is enabled:

```
docker exec <rabbitmq-container> rabbitmq-plugins list | grep amqp1_0
# Expected: [E*] rabbitmq_amqp1_0
```

Network access summary

The following outbound connections must be open from the Docker host:



Destination	Port	Purpose
Innovator Server	80 / 443	Item data, OAuth token requests
OAuth Server	80 / 443	OAuth token requests
Vault Server	80 / 443	File content download for indexing
Temporal frontend	7233 (TCP)	Workflow orchestration
SQL Server	1433 (TCP)	AI Search database
RabbitMQ (AMQP)	5671 (TLS) or 5672 (plain)	CDC event stream
Azure Service Bus	5671 (AMQP TLS) or 443 (AMQP over WebSocket)	CDC event stream (alternative to RabbitMQ)
Edge API configurator	varies	Queue provisioning
Embeddings API (OpenAI-compatible)	443	Vector generation
Docker Hub	443	Pulling Docker images

Minimum hardware

Resource	Minimum	Notes
CPU	4 vCPUs	ais-parser-worker uses all available cores for ML inference
RAM	8 GB	Parser worker loads approximately 4 GB of ML models; remainder for API, indexer, and Temporal
Disk	20 GB	For the parser-staging volume (transient workspace during file indexing; actual size depends on the number and size of files indexed concurrently)

Pre-installation

Generate the certificate

AI Search authenticates to Innovator using an X.509 certificate (OAuth 2.0 impersonate grant). You need two files:

- .pfx: private key and certificate, mounted into ais-indexer-worker
- .cer: public key only, registered in Innovator OAuth

Run the following (requires OpenSSL 3.0 or later):

```
# Generate a 2048-bit RSA key and self-signed certificate valid for 5 years
openssl req -x509 -newkey rsa:2048 -keyout key.pem -out cert.pem \
  -days 1825 -noenc -subj "/CN=IndexerWorker"

# Extract public key "needed for Innovator registration (Section 2.2)
openssl x509 -in cert.pem -out IndexerWorker.cer

# Bundle into .pfx without a password (leave ARAS_OAUTH_CERTIFICATE_PASSWORD empty in .env):
openssl pkcs12 -export -out IndexerWorker.pfx -inkey key.pem -in cert.pem -passout pass:

# Or with a password (set ARAS_OAUTH_CERTIFICATE_PASSWORD=<password> in .env):
openssl pkcs12 -export -out IndexerWorker.pfx -inkey key.pem -in cert.pem -passout pass:
<password>

# Linux / macOS
rm key.pem cert.pem

# Windows (PowerShell)
Remove-Item key.pem, cert.pem
```

If the IndexerWorker client **already exists** in Innovator, you don't need to generate a new certificate. The existing IndexerWorker.pfx may be located in the Innovator OAuth certificate directory. Copy it from there:

```
<InnovatorRoot>/OAuthServer/App_Data/Certificates/IndexerWorker.pfx
```

If you only have the .pfx, extract the public .cer from it:

```
openssl pkcs12 -in IndexerWorker.pfx -nokeys -clcerts -out IndexerWorker.cer -passin pass:
```

Configure Innovator OAuth

If the IndexerWorker OAuth client does not yet exist in your Innovator installation, register it now.



1. Copy Certificate to Innovator

Copy IndexerWorker.cer to the Innovator OAuth certificate directory:

```
<InnovatorRoot>/OAuthServer/App_Data/Certificates/IndexerWorker.cer
```

2. Update oauth.config

Edit /OAuthServer/oauth.config and add the following block inside :

```
<clientRegistry id="IndexerWorker" enabled="true">
  <secrets>
    <secret type="JwtBearerAssertionServerSecret">
      <certificate filePath="App_Data/Certificates/IndexerWorker.cer" />
    </secret>
  </secrets>
  <allowedScopes>
    <scope name="Innovator" />
  </allowedScopes>
  <allowedGrantTypes>
    <grantType name="impersonate" />
  </allowedGrantTypes>
  <tokenLifetime accessTokenLifetime="3600" />
</clientRegistry>
```

Configuration details:

- id="IndexerWorker" must match INNOVATOR_OAUTH_CLIENT_ID in .env (default: IndexerWorker)
- filePath is the relative path to the .cer file inside the OAuthServer directory

Restart the OAuth Server

Restart the OAuth server process to apply the new client registration.

Grant SQL Server permissions

Important



Managed deployments: if the three AIS_DB_* connection strings have already been provisioned by a DBA or platform administrator, skip this section; pass the provided strings to .env and proceed to next step, i.e., Obtain the AMQP connection string.

AI Search migrations run inside the Innovator database and create the [ais] schema and its tables. No new database is created. All three connection strings should point to the existing Innovator database.

The login used by AIS_DB_SCHEMA_CONNECTION_STRING must have:

Permission	Purpose
db_owner on the Innovator database	Create schema, tables, and manage role membership

The logins for AIS_DB_DATA_CONNECTION_STRING and AIS_DB_SEARCH_CONNECTION_STRING receive their minimum required database roles (ais_data_role, ais_search_role) automatically during migration. In simple deployments all three connection strings may reference the same login.

```
-- Create a dedicated login for AI Search (substitute your own name and password)
CREATE LOGIN aisearch WITH PASSWORD = '<strong-password>';

-- Grant db_owner on the Innovator database
USE [innovator]; -- substitute your actual Innovator database name
CREATE USER aisearch FOR LOGIN aisearch;
ALTER ROLE db_owner ADD MEMBER aisearch;
```

Important

db_owner is required only for the initial migration run. The migration creates the [ais] schema, custom roles (ais_schema_install_role, ais_data_role, ais_search_role), and assigns logins to them automatically. After the first successful run you can demote the schema login. Refer to the code block that follows.



```
USE [innovator];
ALTER ROLE ais_schema_install_role ADD MEMBER aisearch;
ALTER ROLE db_owner DROP MEMBER aisearch;
```

Obtain the AMQP connection string

AI Search uses the AMQP 1.0 protocol to consume CDC events from the message broker. Two broker options are supported.

RabbitMQ (on-premises, self-hosted):

```
# TLS (recommended for production, port 5671):
amqps://<user>:<password>@<rabbitmq-host>:5671/

# Plain (development / internal network only, port 5672):
amqp://<user>:<password>@<rabbitmq-host>:5672/
```

Obtain the credentials from the team that manages the Edge API / CDC stack.

Important

RabbitMQ must have the rabbitmq_amqp1_0 plugin enabled (see the Edge API prerequisites check above). Use amqps:// (TLS) in production.

Azure Service Bus (cloud-hosted):

```
# Standard Azure Service Bus connection string (from Azure Portal):
Endpoint=sb://<namespace>.servicebus.windows.net/;SharedAccessKeyName=<policy>;SharedAccessKey=
<key>

# AMQP URI form (port 5671 â€” TLS):
amqps://<policy>:<key-url-encoded>@<namespace>.servicebus.windows.net
```

Obtain the connection string from **Azure Portal** â†’ **your Service Bus namespace** â†’ **Shared access policies** â†’ **your policy** â†’ **Primary Connection String**.



Important

Azure Service Bus natively supports AMQP 1.0; no plugin configuration is required. Use the Standard or Premium tier; the Basic tier does not support topics.

Set the value as `AMQP_CONNECTION_STRING` in `.env`.

Obtain embeddings API credentials

AI Search generates vector embeddings via any OpenAI-compatible embeddings API; the application calls the standard `/v1/embeddings` endpoint.

You can use Azure OpenAI, OpenAI, or any compatible provider.

Variable	Description
EMBEDDINGS_ENDPOINT	Base URL of the embeddings API (e.g. <code>https://.openai.azure.com/openai/v1/</code> for Azure OpenAI, <code>https://api.openai.com/...</code> for OpenAI)
EMBEDDINGS_API_KEY	API key for the embeddings provider
EMBEDDINGS_MODEL_ID	Deployment or model name. Use a model that outputs 1536-dimensional vectors (e.g. <code>text-embedding-3-small</code>).
VECTOR_DIMENSION	Output dimension of the model. Must be 1536; fixed by the database schema.

Important

Azure OpenAI: find the endpoint and key in Azure Portal `â†’` your Azure OpenAI resource `â†’` Keys and Endpoint. Use the deployment name you gave the model as `EMBEDDINGS_MODEL_ID`.

Installation

All commands in this section are run from the `ai-search/` directory:

```
cd ai-search/
```



Configure the Temporal connection

Set TEMPORAL_HOST in .env to your Temporal frontend address:

```
# Temporal Cloud:
TEMPORAL_HOST=<namespace>.tmprl.cloud:7233

# Self-hosted cluster:
TEMPORAL_HOST=temporal.company.com:7233
```

If your cluster uses a non-default namespace, set TEMPORAL_NAMESPACE in .env.

Temporal Cloud requires an API key for authentication. Set it in .env:

```
TEMPORAL_API_KEY=<your-temporal-cloud-api-key>
```

The key is passed to every AI Search service as Temporal:Credentials:ApiKey (the .NET services) and PARSER_WORKER__TEMPORAL__API_KEY (the parser worker). Leave TEMPORAL_API_KEY empty for a self-hosted Temporal cluster that does not require authentication.

Configure the environment file

Open .env in a text editor and fill in all required values:

Variable	Description
IMAGE_TAG	AI Search image version to deploy. Obtain the tag from the Aras release notification.
IMAGE_REGISTRY	Docker Hub organization prefix for AI Search images (e.g. arascorporation)
TEMPORAL_HOST	Temporal frontend address (e.g. temporal:7233)
INNOVATOR_INSTANCE_ID	Unique identifier for this Innovator instance. Allowed characters: A-Z, a-z, 0-9, -, _; maximum 64 characters. Example: innovator-prod
INNOVATOR_SERVER_URL	Innovator HTTP base URL
INNOVATOR_OAUTH_URL	OAuth Server base URL



AIS_DB_DATA_CONNECTION_STRING	SQL Server connection string for data operations
AIS_DB_SEARCH_CONNECTION_STRING	SQL Server connection string for search queries
AIS_DB_SCHEMA_CONNECTION_STRING	SQL Server connection string for schema migrations
AMQP_CONNECTION_STRING	AMQP 1.0 connection string: RabbitMQ (amqps://user:pass@host:5671/) or Azure Service Bus (Endpoint=sb://...)
CDC_CONFIGURATOR_URL	Edge API configurator service URL
EMBEDDINGS_ENDPOINT	Embeddings API base URL (OpenAI-compatible)
EMBEDDINGS_API_KEY	Embeddings API key

Optional variables with defaults (see .env.example for the full list):

INNOVATOR_OAUTH_CLIENT_ID, INNOVATOR_SERVICE_USER, ARAS_OAUTH_CERTIFICATE_PASSWORD, CERTS_HOST_DIR, TEMPORAL_NAMESPACE, EMBEDDINGS_MODEL_ID, VECTOR_DIMENSION, CDC_INSTANCE_NAME, CDC_ORIGIN, and others.

Important

Linux Docker hosts: set DOCKER_HOST_GATEWAY=host.docker.internal in .env. On Linux, host.docker.internal is not injected automatically; this variable tells the containers how to reach services running on the Docker host (Innovator, SQL Server, RabbitMQ).

Place the certificate

Copy the IndexerWorker.pfx file (see Section 2.1) into the certificate directory. The default location is ai-search/certs/IndexerWorker.pfx:

```
cp /path/to/IndexerWorker.pfx certs/IndexerWorker.pfx
```

If you use a different path, set CERTS_HOST_DIR in .env to that directory's absolute path.

Pull images and start the stack

AI Search images are published to Docker Hub; no Docker Hub login is required.



```
# Pull all AI Search images
docker compose pull

# Start the stack in the background
docker compose up -d
```

The ais-db-migrations init container runs automatically on first start and creates the AI Search database schema. It exits when complete; the main services start afterward.

Important

External access: ais-api binds to 127.0.0.1:5080 (loopback only). To expose it outside the host, place a reverse proxy (such as nginx or a cloud load balancer) in front of it.

Important

After a host reboot: all containers start automatically (restart: unless-stopped). During the first 15 to 30 seconds, transient connection errors may appear in logs while Temporal initializes. This is expected and resolves automatically.

Verification

Allow 30 to 60 seconds for all containers to reach a running state, then run the checks below.

Container status

```
docker compose ps
```

Expected: ais-api, ais-indexer-worker, and ais-parser-worker are Up; ais-db-migrations and parser-staging-init are **Exited (0)**.

Readiness health check

```
curl -s -o /dev/null -w "%{http_code}" http://localhost:5080/health/ready
```

Expected: 200



Individual probe results (one line per subsystem):

```
curl -s http://localhost:5080/health/ready | jq . # or: python3 -m json.tool
```

You can also query each subsystem individually:

```
curl -s http://localhost:5080/health/ready/sql-data
curl -s http://localhost:5080/health/ready/sql-search
curl -s http://localhost:5080/health/ready/innovator
curl -s http://localhost:5080/health/ready/embeddings
curl -s http://localhost:5080/health/ready/temporal
curl -s http://localhost:5080/health/ready/cdc-configurator
```

Important

The subsystem names in these health endpoints correspond directly to the six service health indicator dots visible in the AI Search Configurator. See the **AI Search Configurator - Admin Guide** for reference.

Temporal workers

1. Open the Temporal UI (Temporal Cloud dashboard or your self-hosted Temporal UI).
2. Navigate to Workers for the namespace configured in TEMPORAL_NAMESPACE (default: default).
3. Confirm that workers are shown as polling the ai-search-index- task queue.

Container logs

```
docker compose logs --tail 50 ais-api
docker compose logs --tail 50 ais-indexer-worker
```

A healthy startup log for ais-api includes:

```
info: Microsoft.Hosting.Lifetime[0]
      Application started. Press Ctrl+C to shut down.
```

First-time configuration



After the stack is running and healthy, install the AI Search component and configure which Innovator ItemTypes AI Search should index.

Install the AI Search component

The AI Search component (AISearch AML package) adds the AI Search button to the Innovator command bar and is required for the Search UI to be accessible.

1. Obtain the AISearch package from the Aras AI Search release archive.
2. Install it in Innovator using the Aras Update tool.
3. In Innovator, navigate to Administration → Variables → ais_SearchUIUrl and set its value to the URL through which users' browsers can reach the AI Search UI:
 - With a reverse proxy (recommended for production): use the proxy's public URL, for example `https://ais.company.com`
 - Without a reverse proxy: ais-api binds to 127.0.0.1:5080 (loopback only by default). Expose the port to the network and use the Docker host's accessible address, for example `http://:5080`
4. Reload Innovator. The AI Search button appears in the command bar.

Access the Configurator

The AI Search UI runs inside Innovator as part of the command-bar integration. Opening the AIS API URL directly in a browser shows "Authentication Required" because the SPA requires the Innovator session context.

To reach the AI Search Configurator:

1. Open Innovator as an administrator.
2. Select the AI Search button in the command bar; the AI Search UI opens within Innovator.
3. Administrator users see a gear icon in the top-right corner of the AI Search UI. Select it to open the Configurator.

Define the indexing scope

1. In the AI Search Configurator, add the ItemTypes you want to make searchable (for example, Part, Document, ECO).
2. For each ItemType, select which properties to include in the search index.



3. Select Apply.

AI Search will provision the required message broker queues and bindings via the CDC configurator, enable SQL Server CDC tracking on the selected tables, and start initial indexing workflows in Temporal to index existing items.

Verify initial indexing

Monitor progress in the Temporal UI:

- Navigate to **Workflows** and filter by Workflow Type initial-item (item indexing) or initial-file-indexer (file indexing).
- Running workflows indicate indexing is in progress; they complete automatically.

You can also monitor progress in the Indexing Status section of the AI Search Configurator. See the **AI Search Configurator - Admin Guide** for details.

Troubleshooting

Certificate not found or authentication errors

Symptom: ais-indexer-worker logs show FileNotFoundException for the PFX, OAuth 401 Unauthorized responses, or a CryptographicException wrapping UnauthorizedAccessException: Access to the path '...pfx' is denied / IOException: Permission denied.

Checks:

1. Confirm the file exists at /.pfx.
2. Verify the container can read it:

```
docker compose exec ais-indexer-worker sh -c "echo /Container/Certificates/*"
```

This only proves the file is listed, not that it is readable; the AIS images run as a fixed non-root user. Check which one:

```
docker inspect <indexer-worker-image> --format '{{.Config.User}}'
```



If the .pfx was generated or copied with a restrictive mode (e.g. 0600, owner-only) and is owned by root, the non-root container user cannot open it â€” the path exists but the load still fails with UnauthorizedAccessException / Permission denied. Inspect the file's owner and mode as seen from a Linux container (works the same way from a Windows or Mac Docker host):

```
docker run --rm -v <CERTS_HOST_DIR>:/certs busybox stat /certs/<INNOVATOR_OAUTH_CLIENT_ID>.pfx
```

Fix by making the file world-readable (matching the default mode of the .cer file), then restart the worker so it reloads the certificate:

```
docker run --rm -v <CERTS_HOST_DIR>:/certs busybox chmod 644
/certs/<INNOVATOR_OAUTH_CLIENT_ID>.pfx
docker compose restart ais-indexer-worker
```

3. Confirm ARAS_OAUTH_CERTIFICATE_PASSWORD in .env matches the PFX password (can be empty).
4. Verify the OAuth client is registered in Innovator with the correct Client ID.

AMQP connection refused or TLS errors

Symptom: ais-indexer-worker or ais-api logs show AMQP connection errors on startup.

Checks:

1. Verify the AMQP_CONNECTION_STRING scheme and broker type:
 - **RabbitMQ:** amqps:// (port 5671) or amqp:// (port 5672)
 - **Azure Service Bus:** Endpoint=sb://... or amqps://...servicebus.windows.net
2. **RabbitMQ only:** confirm the broker is reachable on the AMQP port:

```
nc -zv <rabbitmq-host> 5671 # TLS (production)
```

Expected: Connection to ... succeeded

3. **RabbitMQ only:** verify the rabbitmq_amqp1_0 plugin is enabled:

```
docker exec <rabbitmq-container> rabbitmq-plugins list | grep amqp1_0
```



4. Azure Service Bus only: confirm the namespace and policy name are correct and the policy has Send and Listen permissions on the topic or queue.

Embeddings API key invalid

Symptom: ais-indexer-worker or ais-api logs show 401 Unauthorized responses from the embeddings endpoint.

Health check:

```
curl -s http://localhost:5080/health/ready/embeddings
```

If it returns Unhealthy, the API key or endpoint URL is incorrect.

Fix: Correct EMBEDDINGS_API_KEY and EMBEDDINGS_ENDPOINT in .env, then restart:

```
docker compose restart ais-api ais-indexer-worker
```

Database migration failed

Symptom: ais-db-migrations container exits with non-zero code; ais-api, ais-indexer-worker, and ais-parser-worker fail to start.

Check the migration logs:

```
docker compose logs ais-db-migrations
```

Common causes:

Error	Fix
Login failed for user	Verify AIS_DB_SCHEMA_CONNECTION_STRING credentials
Cannot open server...	SQL Server not reachable; check host and port and firewall
User does not have permission to perform this action	Grant db_owner to the login on the Innovator database

Temporal worker not polling



Symptom: Temporal UI shows no workers on the expected task queue; indexing workflows do not start.

Checks:

1. Confirm ais-indexer-worker container is **Up**:

```
docker compose ps ais-indexer-worker
```

2. Check worker logs for connection errors:

```
docker compose logs --tail 100 ais-indexer-worker | grep -i  
"temporal\|error"
```

3. Verify the health probe for Temporal:

```
curl -s http://localhost:5080/health/ready/temporal
```

4. Confirm TEMPORAL_HOST in .env is correct and reachable from Docker:

```
docker run --rm --network ais-network busybox nc -zv <temporal-host>  
7233
```

Health endpoint returns Degraded or Unhealthy

The /health/ready response body contains the status of each subsystem. Query individual probes to isolate the failing component:

```
curl -s http://localhost:5080/health/ready | jq . # or: python3 -m json.tool
```

Cross-reference with the subsystem-specific troubleshooting sections above.

If issues persist after following the steps above, contact Aras Technical Support via the Subscriber Portal with your tenant ID, the symptom observed, and any error details from the relevant container



logs.



Installation of AI Search on SaaS

This guide explains how to enable **AI Search** in your **Aras Innovator** SaaS instance.

The AI Search application (API, indexer worker, embeddings) is deployed and operated on the platform side. To turn it on for your instance you need to add two things to your Work repository and run your deployment pipeline:

1. **AI Search AML package:** adds the "Search" command and its configuration to Innovator.
2. **The OAuth transformation:** registers the indexer as a trusted OAuth client so it can read your data.

Important

Everything that follows is applied through your normal Aras DevOps deployment pipeline. No manual changes on the running instance are required.

Prerequisites

- An active Aras InnovatorEdge AI Subscription (AS-1242).
- AI Search must be entitled for your SDE (Standard Development Environment). Before starting, open a [support incident](#) to request that it be enabled, as it is not enabled by default for all customers.
- Aras DevOps version 1.15 or later.
- Access to your Work repository, the Aras DevOps repository your instance is built from.
- The AI Search release package provided by Aras, a folder named AISearch containing an Import subfolder. Contact your Aras Customer Success Manager or download from the Aras Subscriber Portal to obtain this package.

Step 1: Add the AI Search AML package

1. Copy the released **AISearch** folder into your Work repository at:

```
~WorkRepository/AML-packages/AISearch
```

After copying, the structure looks like this:



```

~WorkRepository/AML-packages/AISearch/Import/CommandBarItem/ais_search.xml
~WorkRepository/AML-packages/AISearch/Import/CommandBarSectionItem/...
~WorkRepository/AML-packages/AISearch/Import/Method/ais_search_click.xml
~WorkRepository/AML-packages/AISearch/Import/Variable/ais_SearchUIUrl.xml

```

2. Register the package in the imports manifest `~WorkRepository/AML-packages/imports.mf`. Add the following entry inside the `imports` element, after the other solution packages. This file already exists; the entry is added to it, not a replacement of the file.

```
<package name="com.aras.innovator.solution.AISearch" path="AISearch\Import" />
```

Step 2: Add the OAuth transformation

Create a new transformation file at the path below, or modify the existing one if it is already present. Add the following `clientRegistry` block inside the existing `clientRegistries` element:

```
~WorkRepositories/TransformationsOfConfigFiles/OAuthServer/OAuth.config
```

Paste this content exactly:



```
<?xml version="1.0" encoding="utf-8"?>
<oauth xmlns:xdt="http://schemas.microsoft.com/XML-Document-Transform">
  <server>
    <clientRegistries>
      <clientRegistry id="IndexerWorker" enabled="true"
        xdt:Transform="InsertIfMissing" xdt:Locator="Match(id)">
        <secrets>
          <secret type="JwtBearerAssertionServerSecret">
            <certificate filePath="App_Data/Certificates/IndexerWorker.cer">
            </certificate>
          </secret>
        </secrets>
        <allowedScopes>
          <scope name="Innovator"></scope>
        </allowedScopes>
        <allowedGrantTypes>
          <grantType name="impersonate"></grantType>
        </allowedGrantTypes>
        <tokenLifetime accessTokenLifetime="3600"></tokenLifetime>
      </clientRegistry>
    </clientRegistries>
  </server>
</oauth>
```

This transformation registers the AI Search indexer as a trusted OAuth client in Innovator. The `IndexerWorker.cer` certificate referenced here is deployed automatically to your Innovator instance by the platform during deployment; no manual certificate handling is required. The transformation is idempotent: it inserts the `IndexerWorker` client only if it is not already present, so re-running the pipeline is safe.

Step 3: Commit and deploy

1. Commit both changes to your Work repository:
 - `~WorkRepository/AML-packages/AISearch/**` and the updated `Work/AML-packages/imports.mf`
 - `~WorkRepository/TransformationsOfConfigFiles/OAuthServer/OAuth.config`
2. Push the changes to your online SDE.
3. Run your deployment pipeline as usual.



The pipeline applies the AML package and the OAuth transformation to your instance. When it completes, the **Search** command is available in Aras Innovator.

Important

AI Search is available in online SDE and production environments only. It is not available in the local development environment (LDE). For guidance on working with your deployment pipeline, see the [Aras DevOps user guide](#).

Verify

- Open Aras Innovator and confirm the **Search** command (AI Search) is present.
- Open the AI Search Configurator using the gear icon and configure at least one ItemType for indexing before running a search. See the **AI Search Configurator - Admin Guide** for the full configuration procedure.

If the Search command does not appear, or the indexer cannot authenticate, re-run the pipeline and confirm both files were committed to the paths shown above.



AI Search Configurator - Admin 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.

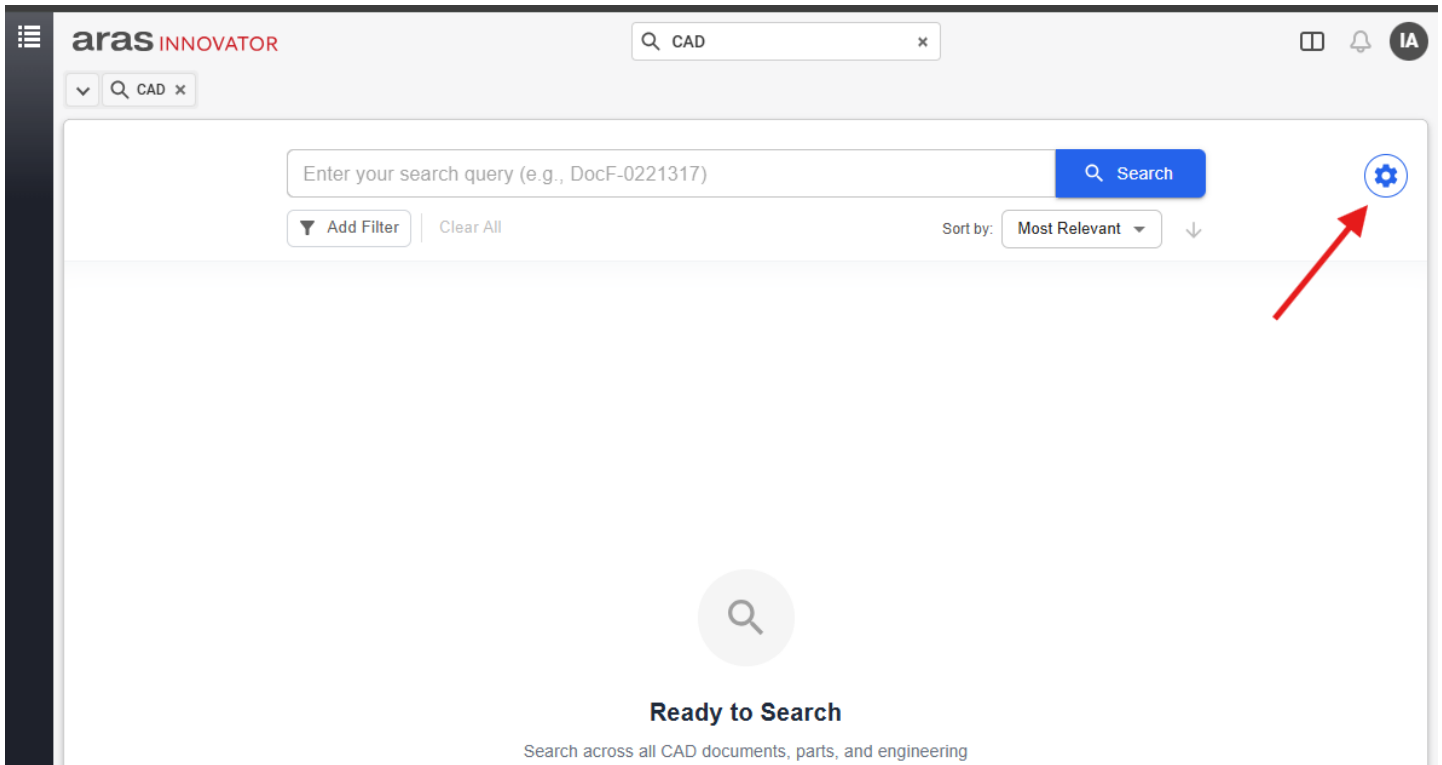
To return to search after configuring, use the **Back to Search** arrow in the Indexing Scope header.

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:

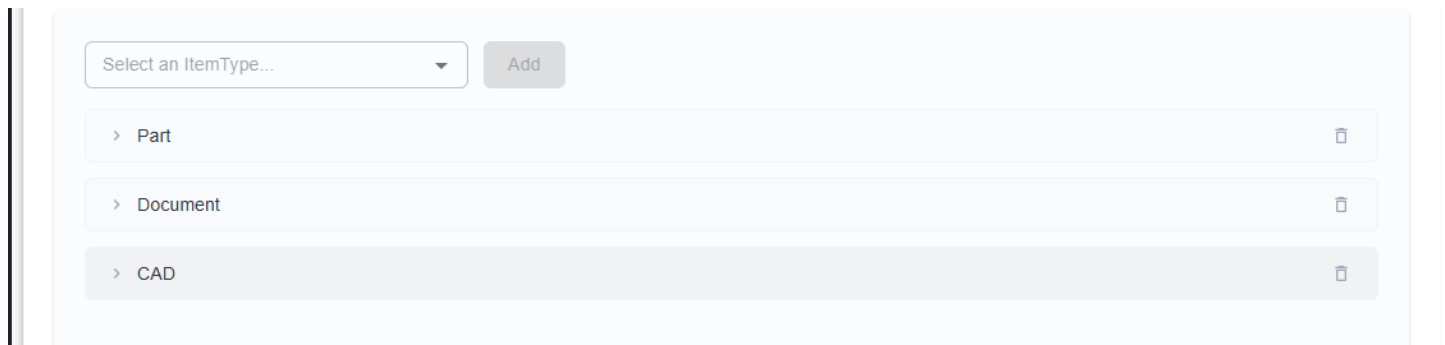
- **Directly to scope:** add the Relationship ItemType itself as an indexable type.
- **As file properties on another ItemType:** configure relationship-sourced file properties under a regular ItemType in the File Properties section (see below).

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.

Properties

Select which properties will be available for vectorization and searching.

FIELD NAME		
item_number		
name		
description		
state		

File Handling Settings

Included File Types

Select file types...

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

md

modified_by_id

modified_on

string

Add

Important

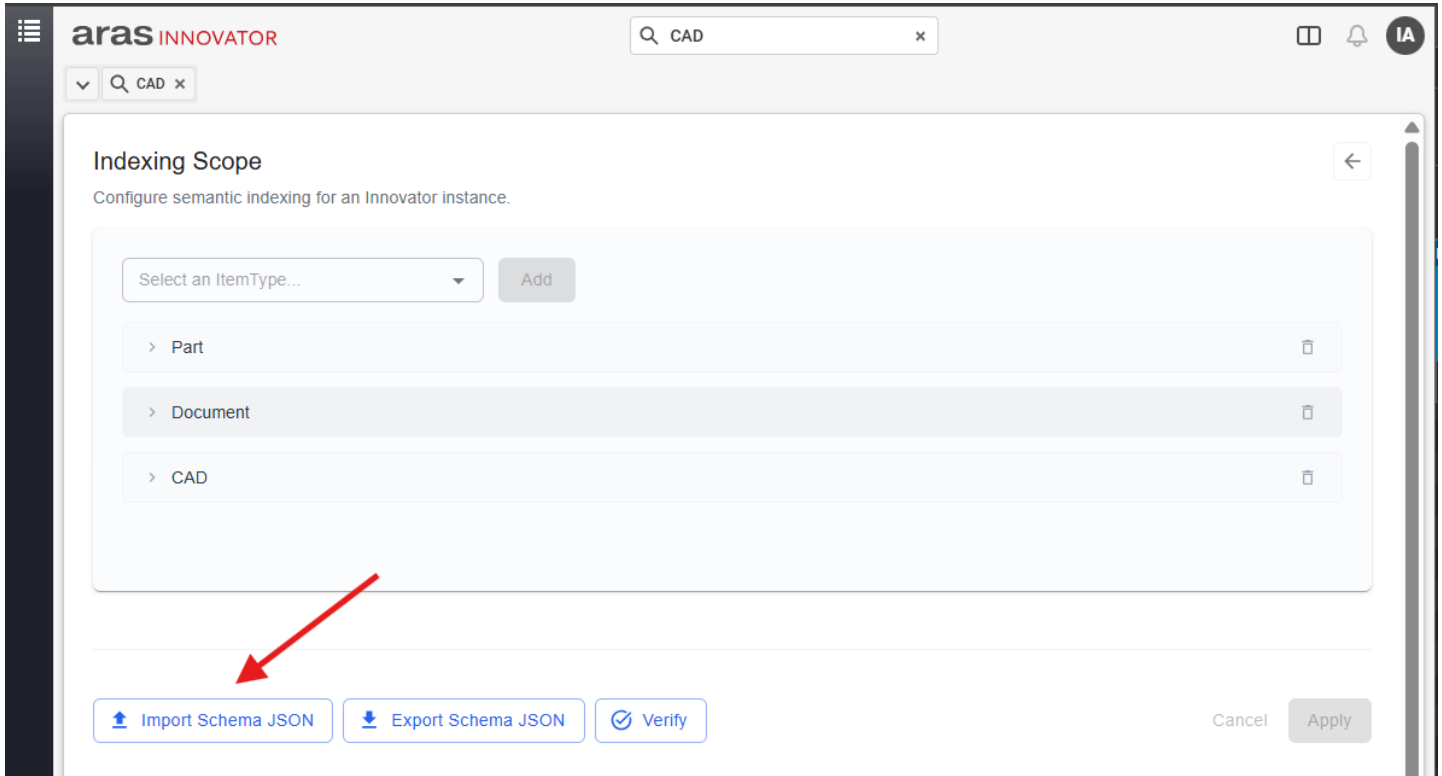
The property selector shows a consistent set of Innovator properties regardless of which ItemType is selected.

ItemType-specific custom properties such as `item_number`, `description`, and `classification` do not appear in the selector.

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 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.





aras INNOVATOR

Q CAD x

Q CAD x

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 <code>keyed_name</code> 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)

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.

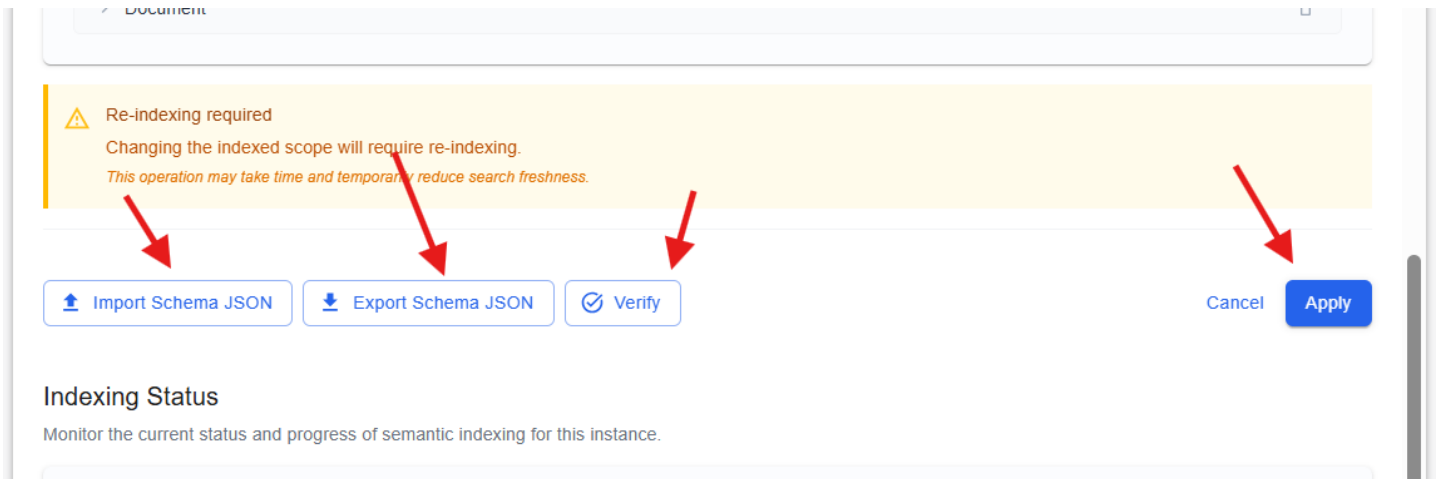
Schema import, export, and Verify

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, Cancel, and Apply.



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 triggers indexing for any new or changed scope.

Outcome	Notification
Success	<i>Configuration applied</i>
Validation failure	<i>Apply failed: field-level error details are shown</i>
Network or server error	<i>Apply failed: error message is shown</i>

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.

This warning appears whenever there are pending changes to apply. Review scope changes carefully before selecting **Apply**, as applying the configuration triggers re-indexing work on the backend.

Important

A UI control to choose between full index rebuild and incremental indexing is not available in this release. Re-indexing behavior is determined by the backend when Apply is submitted.

If the AI Search package is not installed on the 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.

Indexing Status

Indexing Status

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

Last updated: 2026-07-02 15:02 GMT+2

Auto-refresh: 30s

● sql-search

● sql-data

● innovator

● embeddings

● temporal

● cdc-configurator

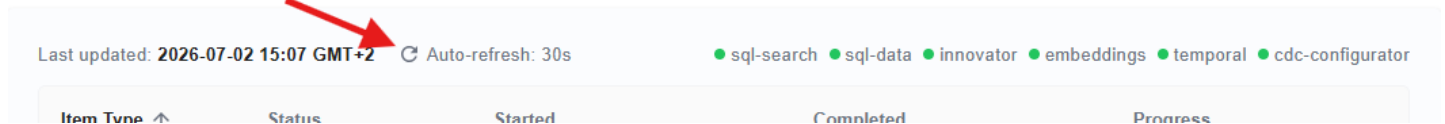
Item Type ↑	Status	Started	Completed	Progress
ITEM INDEXERS				
CAD	Completed	2026-06-23 19:52 GMT+2	2026-06-23 19:52 GMT+2	10 / 10
Document	Completed	2026-06-23 19:52 GMT+2	2026-06-23 19:52 GMT+2	15 / 15
Part	Completed	2026-06-23 19:52 GMT+2	2026-06-23 19:52 GMT+2	25 / 25
Rows per page: 10 1-3 of 3 < > Total count: 50				
FILE INDEXER				
File	Pending			Awaiting discovery



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.

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



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.



Monitoring progress

Indexing Status shows one row per ItemType in scope under **Item Indexers**, plus a single **File Indexer** row when file handling is enabled for at least one ItemType.

Each row 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
---------	---------



Awaiting discovery...	Indexer is pending and the total item count is not yet known
Discovery did not complete	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.

When errors are recorded for a run, the status row shows an N errors link (for example, 3 errors).

Select the status chip or the error link to open the **Indexing Errors** dialog for that indexer.

The dialog shows:

- A summary of the run (status, started/completed times, processed and total counts)

- A list of workflow errors, each with timestamp, activity name, error category, message, and cause
- Expandable technical **details** for each error
- A **Copy error details** action for support diagnostics

Error categories include **Timeout**, **Activity Failure**, and **Application Failure**.

Only the current version of an item, and the current version of any indexed file attached to it, is included in the index. Earlier versions are not 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 Innovator table
An expected file type does not appear in the file type selector	Two file types share the same extension (only the first is shown), or the file type was added within the one-hour cache window	Reopen the configurator to refresh the cache, or check 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 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 status chip or N errors link to open the Indexing Errors dialog and review details
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 Indexing Errors dialog.

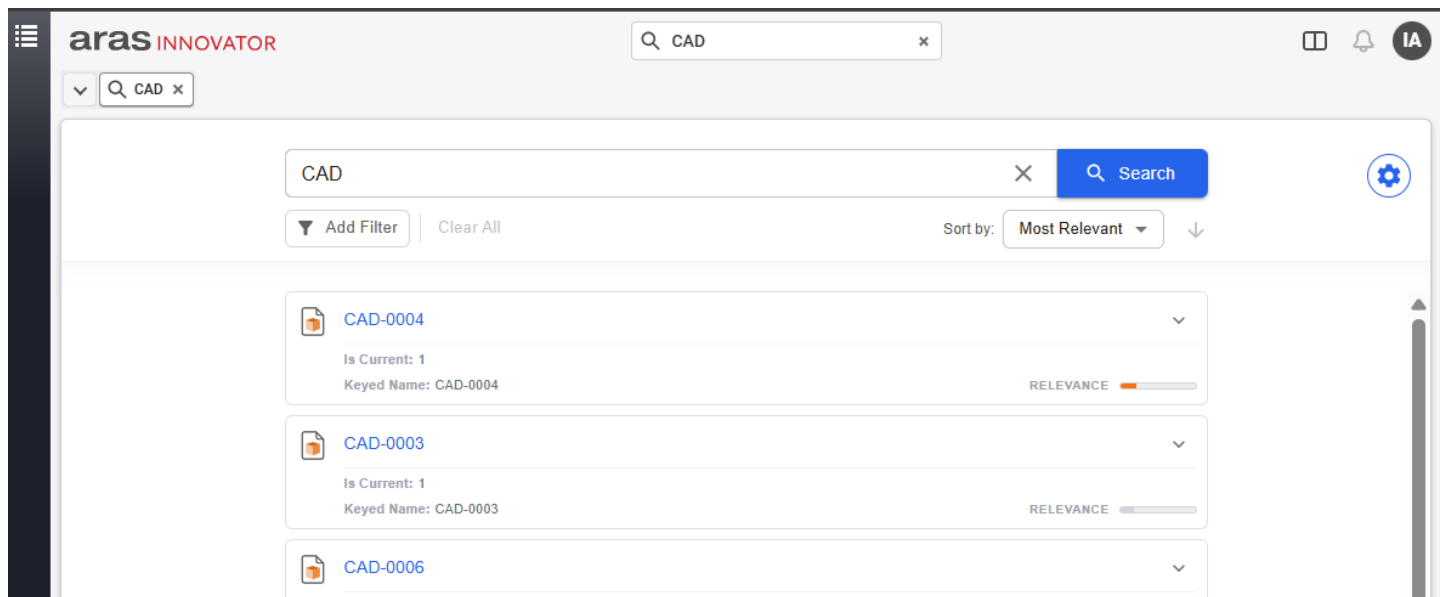


AI Search - User Guide

Visibility and Access Modes

AI Search replaces the standard Enterprise Search entry point in Innovator. It is available as:

- the search field in the Innovator header
- a dedicated AI Search page that opens when you select the search entry point



Accessing AI Search in Innovator

Both the header search field and the dedicated AI Search page accept the same natural language queries and return the same results.

In this release, AI Search is available as an embedded experience inside Innovator. A standalone, out-of-Innovator access mode is not available in this release.

Opening the dedicated AI Search page with a `searchText` query parameter in the URL automatically runs that search when the page loads.

Admin panel access

Administrators see a gear icon in the top-right corner of the dedicated AI Search page. Selecting it opens the AI Search Admin panel, where indexing scope and search presentation settings are configured.

This button is visible only to users with administrator access.

For full Admin panel documentation, see the separate AI Search Configurator - Admin guide.

Interacting with AI Search

Before you search

When you first open the dedicated AI Search page and have not run a search yet, the results area shows Ready to Search with a short description and example query chips, such as:

- "123 module description"
- "Body parts with Preliminary state"
- "Cylinder head manual"
- "Latest modified CAD documents"

Select any example chip to run that search immediately.

Running a search

Type a plain language query into either search entry point and select **Search**.

Describe what you are looking for in everyday language: a concept, a material, a function, or a design intent.

AI Search matches by meaning rather than by exact keyword.

You can also press **Enter** in the search box to run the search.

While a search is in progress, the **Search** button is disabled until the current request completes or is stopped.

The **X** icon inside the search box clears the text you typed.

It does not clear existing results or reset the page. To reset after a failed or empty search, use **Clear search and try again** in the results area.

Queries must be at least 3 characters long. Submitting a shorter query shows the following message:

Search query too short
Please enter at least 3 characters to search. Try a longer keyword or phrase.



Select **Clear search and try again** to reset the search field and enter a new query.



Search query too short

Please enter at least 3 characters to search. Try a longer keyword or phrase.

[Clear search and try again](#)

While a search is running

If a search takes longer than a few seconds, an information banner appears:

Your search is still running
Results will appear as soon as they're ready.

Select **Stop** to cancel the in-progress search. Partial results are not kept after you stop.

Loading more results

Search results are returned in pages.

When more results are available, a **Load More** button appears at the bottom of the result list.

Select **Load More** to fetch the next page using the same search query, applied filters, and sort order as the current result set.

Load More does not apply staged filter changes; if you have edited filters but not yet selected **Apply Filters**, pagination continues with the previously applied filter set.

While additional results are loading, the button shows **Loading...** and is disabled.

Search Filters and Sorting

Use filters to narrow semantic search results after you enter a search query. Filters are built in the filter panel below the search bar; active filters appear as chips above the panel.

CAD

×

🔍 Search

🔍 is_current = true

×

🔍 item_type = Document

×

⌵ Add Filter

Clear All

Apply Filters

Sort by: Most Relevant ⌵

⌵

×

Cancel

Update Filter

Edit Filter

Property

Item Type

⌵

equals

Item Type

Document

⌵

📄

CAD-0004

⌵

Is Current: 1

Staged and applied filters

The search page keeps two filter states:

State	Meaning
Staged filters	Filters you have added, edited, or removed in the UI but not yet sent to the search service
Applied filters	Filters that were included in the last search request

Search results always reflect applied filters. Changes you make in the filter builder are staged until you apply them or run a new search.

Adding a filter



1. Enter a search query and run the search.
2. Select **Add Filter** to open the filter builder.
3. Select a **property** (field).
4. Choose a **value**. The control depends on the property type (see **Available filter properties below**).
5. Select **Add Filter** in the builder to save the filter to the staged list.

The filter appears as a chip above the filter panel. It does **not** change results until you apply it.

The screenshot shows the 'Add New Filter' dialog box. At the top, there are 'Cancel' and 'Add Filter' buttons. Below them, the text 'Add New Filter' is displayed. The main area contains a 'Property' dropdown menu, an 'equals' operator, and a 'Value' dropdown menu. The 'Property' dropdown is open, showing 'Item Type' and 'Is Current' as options. The 'Value' dropdown is set to 'Value'. Below the dialog, a filter chip 'Is Current: 1' is visible above a list of results.

Notes:

- Each property can be used **once** at a time.
After you add an **Item Type** filter, that property is no longer available in the dropdown until you remove that filter.
When both available properties are in use, **Add Filter** is disabled.
- Duplicate filters (same property and value) are rejected with a warning notification.
- Select **Cancel** to close the builder without saving.
- If the **Item Type** list fails to load, an inline error is shown with a **Retry** option.

Available filter properties

All filters use the **equals** operator.

Filter	Values
--------	--------



Item Type	Any ItemType configured for indexing in your instance
Is Current	True or False

The **Item Type** list is loaded from your instance's index configuration.

Filter chips

Staged filters are shown as chips in a row below the search bar (up to **8** visible; select **â€¦N More Filters** to expand or **Hide N Filters** to collapse).

Each chip displays: **Property = Value** (for example, **Item Type = Part** or **Is Current = True**).

Chip colors

Appearance	Meaning
Blue (primary) outline	Applied: this filter was included in the last search
Gray outline	Staged only: added or changed, but not yet applied to results

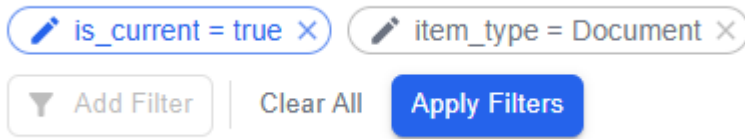
After you add, edit, or remove a filter, affected chips turn gray until you apply changes.

Chip actions

Action	How	Effect
Edit	Select the pencil icon on the chip	Opens the filter builder in edit mode with current values
Remove	Select the X on the chip	Removes that filter from the staged list only

Removing or editing a chip does not immediately update results. You must select **Apply Filters** or run the search again.





Applying filters

The **Apply Filters** button appears when staged filters differ from applied filters.

For example, after adding a filter, editing one, removing one, or selecting **Clear All**.

Select **Apply Filters** to copy staged filters to applied filters and re-run the search with the current query and filters.

Apply Filters is disabled while a search is already in progress.

Staged filters are also applied automatically when you press **Enter** in the search box or select the **Search** button.

In those cases, the latest staged filters are sent with the search request.

Clearing filters

- **Remove one filter:** Select **X** on a chip.
The filter is removed from the staged list.
If it was previously applied, results stay unchanged until you select **Apply Filters** or search again.
- **Clear all filters:** Select **Clear All** to remove all staged filters and close any open filter builder.
Clear All does not run a search by itself. If applied filters were active, **Apply Filters** appears so you can refresh results without those filters.

Sorting results

Use **Sort by** on the right side of the filter row to change result ordering.

Option	Behavior
Most Relevant	Default semantic ranking by search score. No sort direction control.
Item Type	Sorts by item type name. Use the arrow button to toggle ascending or descending order.



Changing the sort field or direction immediately re-runs the search.

This also **applies any staged filter changes** before searching, the same as selecting **Apply Filters**. The direction button is disabled for **Most Relevant**.

Filter and sort behavior summary

Action	Applies staged filters?	Updates results immediately?
Add / edit filter in builder	No (staged only)	No
Remove chip or Clear All	No (staged only)	No
Apply Filters	Yes	Yes
Search or Enter	Yes	Yes
Change sort	Yes	Yes
Load More	No (uses previously applied filters)	Yes (next page only)

Typical workflows

Narrow results after searching

1. Search for a term such as **battery housing**.
2. Select **Add Filter** → **Item Type** → choose a type → **Add Filter**.
3. The chip appears in gray (staged).
4. Select **Apply Filters**.
5. The chip turns blue; results show only matching items.

Change an existing filter

1. Select the pencil icon on a filter chip.
2. Change the value and select **Update Filter**.
3. The chip turns gray (staged).
4. Select **Apply Filters** to refresh results.

Remove all filters and refresh



1. Select **Clear All**.
2. Select **Apply Filters** to search again without filters.

Understanding results

Each result is presented as a card.

The card title is derived from the title template an administrator has configured for that ItemType, or from the item's keyed_name if no template is set.

The card subtitle follows the same pattern.

Each card shows an item type icon and a relevance indicator: a colored bar that reflects how closely the result matches the query:

Bar color	Relevance level
Green	High
Blue	Medium
Orange	Low
Gray	Lowest



×
🔍 Search

▼ Add Filter
Clear All
Sort by: Most Relevant ▼
↓


CAD-0008
^

Is Current: 1

Keyed Name: CAD-0008

Item Number: CAD-0008

Name: Electronics Bay Thermal CFD Model

Description: ANSYS Fluent 3D CFD model of the electronics bay at peak motor controller load. Predicts PCB and component junction temperatures under natural and forced convection with heatsink at 40°C ambient.

State: Preliminary


CAD-0002
▼

Is Current: 1

Keyed Name: CAD-0002

RELEVANCE


CAD-0004
▼

Is Current: 1

Keyed Name: CAD-0004

RELEVANCE


DOC-0005
▼

Is Current: 1

Keyed Name: DOC-0005

RELEVANCE


CAD-0010
▼

Is Current: 1

Keyed Name: CAD-0010

RELEVANCE


DOC-0015
▼

Is Current: 1

Keyed Name: DOC-0015

RELEVANCE

Expanding a result card

Select the expand control on a card to see additional detail.

Expanded cards show indexed property snippets that matched your query.

For example, a **Content** field for matched file text. Property labels are shown using administrator-configured names where available.

Selecting the card title or the expanded card header opens the item record directly in Innovator, where you can view, edit, and navigate the item using all standard Innovator functions.

File results and related items

When a result is a **file** item type, expanding the card may also show a **Related items**:

- section listing parent items linked to that file (for example, the CAD document or part the file belongs to).
- Each related item shows its item type icon and name.
- Select a related item to open it in Innovator.
- If related items cannot be loaded, a warning message **Could not load related items** is shown inside the expanded card.

Error and empty states

The table below lists the main messages shown in the results area, with the action available on each screen.

Screen title	Message	What to do
Ready to Search	Search across all CAD documents, parts, and engineering manuals. Example query chips are shown.	Enter a query or select an example chip to begin.
Search query too short	Please enter at least 3 characters to search. Try a longer keyword or phrase.	Select Clear search and try again , then enter a longer query.
No Matches Found	We couldn't find anything matching your query. Check your spelling or try including more general keywords.	Rephrase the query, adjust filters, or select Clear search and try again .



No results found (with filter guidance)	Try using more specific keywords or applying filters to narrow your search.	Shown when the result limit is reached with no visible matches. Refine the query or add filters.
The Index Is Not Ready Yet or Is Currently Unavailable	The centralized search index is currently unavailable, rebuilding, or temporarily locked by an administrator. Your search cannot return live results at this time.	Wait and try again later, or contact your Innovator administrator. This indicates the search index itself is not ready, not a temporary network error.
Search service is temporarily unavailable	We could not complete your search right now. This is usually a temporary issue.	Select Search again to retry. This indicates a transient service error, distinct from the index-not-ready state.
Invalid search syntax	The search query contains invalid characters or syntax. Please revise your search and try again.	Remove or correct special characters or syntax in the query.
Could not load related items	Shown inside an expanded file result card when parent items fail to load.	Expand another result or open the file item directly in Innovator.

Important Guidelines

AI Search results never exceed your existing Innovator permissions. You only see items and files you are already authorized to see in standard Innovator search. AI Search does not grant any new access.

Customer data is not used to train AI models. Content indexed by AI Search and queries submitted to it are not used to train, retrain, or improve any general purpose or shared AI model. See the Aras Innovator Edge Product Specific Terms for full details.

Semantic Search via Edge API Query-on-the-fly

AI Search results can also be consumed through Edge API's Query-on-the-fly capability, allowing search to be combined with other governed Edge API data access in the same request flow. See the Edge API Query-on-the-fly documentation for details.



Appendix: Basic Terminology

Essential terms and their meaning in the context of AI Search.

Term	Definition
AI Search	The Innovator Edge service that delivers semantic search across Innovator items and attached files.
Indexing Scope	The set of ItemTypes and file types an administrator has configured to be searchable. Defined in the AI Search Configurator.
Indexable Property	A property included in the semantic index for a given ItemType. The property selector in the configurator shows a consistent set of Innovator properties regardless of ItemType. Custom properties specific to individual ItemTypes, such as <code>item_number</code> or <code>description</code> , do not appear in the selector but can be added by specifying their internal name in the JSON schema and importing it. The indexing engine reads from the actual Innovator table, so any valid property name on that table can be indexed.
File Handling	The configuration that determines which file types attached to an ItemType's items are extracted and included in search. Available only for ItemTypes that contain a property of type Item. The selectable file types are sourced live from your Innovator instance's FileType table, not from a fixed product list.
File Properties	A section within each ItemType row in the AI Search Configurator, separate from File Handling Settings. File Properties controls which file-related fields are used for text extraction, vectorization, and search. Each 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. At least one file property must be enabled when file content indexing is required for an ItemType that has file handling configured.
Title Template	A configurable template controlling how a result card's title is displayed, using <code>{{ fieldName }}</code> placeholders. Falls back to the item's <code>keyed_name</code> if not set.
Subtitle Template	A configurable template controlling how a result card's subtitle is displayed. Supports item property placeholders and <code>{{score}}</code> .



Relevance	A measure of how closely a result matches a search query, shown on each result card as a colored bar: green (High), blue (Medium), orange (Low), or gray (Lowest).
Staged filters	Filters that have been added, edited, or removed in the search UI but not yet sent to the search service. Staged filters appear as gray chips. They do not change search results until applied.
Applied filters	Filters that were included in the most recent search request. Applied filters appear as blue chips. Search results always reflect applied filters.
Indexing Status	The live-refreshing section of the AI Search Configurator showing the current state of each indexer in the active configuration, alongside timestamps, progress counts, and service health indicators. Refreshes every 30 seconds. Each indexer shows one of five states: Pending, Running, Completed, Partially Completed, or Failed.
Service Health Indicators	A row of status dots in the Indexing Status section showing the health of the six backend services AI Search depends on: sql-search, sql-data, innovator, embeddings, temporal, and cdc-configurator. A green dot indicates the service is healthy; a red dot indicates it is degraded or unhealthy.
Indexing Errors dialog	A dialog opened by selecting the N errors link on a Failed or Partially Completed indexer row. The dialog shows a summary of the run, a list of workflow errors with their timestamp, activity name, error category, message, and cause, expandable technical details, and a Copy error details action for support diagnostics. Error categories include Timeout, Activity Failure, and Application Failure.
Re-indexing	The process of rebuilding the index for an ItemType after its indexing scope configuration changes. Required whenever properties, file handling, file properties, or the set of indexed ItemTypes is changed on an instance that has already been indexed. Re-indexing behavior is determined by the backend automatically when Apply is submitted.
Change Data Capture (CDC)	The component that detects changes to indexed items and files in Innovator and triggers incremental re-indexing, keeping the index current after initial indexing completes.
Embedding	A vector representation of text or file content used by AI Search to match queries to relevant items by meaning rather than by exact keyword.
Semantic Search API	The API endpoint that returns ranked search results for a given query. Consumed by the AI Search web interface and available to external developers through Edge API.



Relationship ItemType	An ItemType that represents a relationship between two items, for example a BOM relationship. Relationship ItemTypes can be added directly to the indexing scope, or referenced as a source of file properties under a regular ItemType in the File Properties section.
Is Current	A search filter that limits results to the current version of an item. AI Search only indexes the current version of items and files; earlier versions are never included in the index.
Work Repository	The Aras DevOps / BDS source control repository from which a SaaS Innovator instance is built and deployed. For SaaS deployments, AI Search is enabled by adding files to the Work repository and running the deployment pipeline.

AI Search: Frequently Asked Questions

Setup & Prerequisites

Do I need Edge AI to use AI Search? No. AI Search is a component of the Edge AI platform, but it does not require an active Edge AI agent or chat configuration to be used. AI Search runs independently once enabled.

Is AI Search available on-premises? Yes. AI Search supports full on-premises deployment using Docker containers, in addition to SaaS. See Installation of AI Search for OnPrem for the full procedure.

How do I enable AI Search for a SaaS Innovator instance? For Aras-hosted Innovator SaaS instances, enabling AI Search requires adding two files to your Work repository and running your deployment pipeline. See Installation of AI Search for SaaS for the full procedure.

Which Aras Innovator versions are supported? Aras Innovator 14.0.15 or later is required.

What SQL Server version is required for on-premises deployment? SQL Server 2025 or later is required for on-premises deployment. AI Search uses the native vector type introduced in SQL Server 2025. Azure SQL Database with vector support is also supported. If your organization is on SQL Server 2019 or 2022, contact Aras for the compatibility upgrade path before starting installation.

What other infrastructure is required before installing AI Search on-premises? The following must be in place before deploying AI Search on-premises: the Aras InnovatorEdge CDC pipeline (edgeapi-eventing-configurator, edgeapi-eventing-exporter, and a message broker, either RabbitMQ or Azure Service Bus); a Temporal cluster (version 1.24 or later) reachable from the Docker host; and an OpenAI-compatible embeddings API endpoint. AI Search auto-provisions its own durable queues and bindings on the message broker when CDC configuration is applied; no manual queue setup is needed on the RabbitMQ or Azure Service Bus side. See Installation of AI Search for the complete prerequisites list.

Configuration



Can I add a Relationship ItemType to the indexing scope? Yes. Relationship ItemTypes, such as a BOM relationship, can be added directly to the indexing scope in the AI Search Configurator. They can also be used as a source of file properties under a regular ItemType in the File Properties section.

What properties can I select for indexing? The property selector shows a consistent set of Innovator properties regardless of which ItemType is selected. ItemType-specific custom properties, such as `item_number`, `description`, and `classification` on Part, do not appear in the selector. Properties available through the selector include system properties common to all items, plus relationship-level properties for Relationship ItemTypes such as `quantity`, `source_id`, `related_id`, and `sort_order`.

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

What is the File Properties section? File Properties is a section within each ItemType row in the configurator, separate from File Handling Settings. While File Handling Settings selects which file types are included, File Properties controls which file-related fields are used for text extraction, vectorization, and search. Each file property row has an enable/disable switch. Enable at least one file property when file content indexing is required for an ItemType that has file handling configured.

What file types can I include in file handling? 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. The list is searchable; type any part of a format name to filter it. If an expected file type is missing, check whether two types in your database share the same extension, as only the first is shown. The list is cached for one hour when the configurator opens; reopening the configurator refreshes it.

What happens if I change the indexing scope after the instance is already indexed? A warning is shown before the change is applied, indicating that changing the indexed scope will require re-indexing and that this may take time and temporarily reduce search freshness. Re-indexing behavior is determined by the backend automatically when Apply is submitted.



Why does Apply fail with "At least one indexable property is required"? Every ItemType in the indexing scope needs at least one configured property, file-indexable property, or file relationship. Expand the ItemType row in the configurator and add at least one property, enable file handling with file types, and/or enable file properties in the File Properties section before applying.

Can I import and export my configuration? Yes. The AI Search Configurator supports exporting the current configuration as a JSON schema file and importing a JSON schema file to load it into the editor. The imported configuration is not saved until you select Apply. Use Verify to validate a configuration before applying it.

What does the Indexing Errors dialog show? When an indexing run ends with a Failed or Partially Completed status, the status row shows an N errors link. Select it to open the Indexing Errors dialog, which shows a summary of the run, a list of workflow errors with their timestamp, activity name, error category, message, and cause, expandable technical details, and a Copy error details action for use when contacting Aras Technical Support.

Data Access & Security

Can a user see AI Search results they are not authorized to see in Innovator? No. AI Search results are filtered by the requesting user's existing Innovator permissions. AI Search does not grant any new access beyond what a user already has.

Is my data used to train AI models? No. Content indexed by AI Search and queries submitted to it are not used to train, retrain, or improve any general purpose or shared AI model. See the **Aras Innovator Edge Product Specific Terms** for full details.

Does AI Search send my data outside Innovator? During indexing, the AI Search service reads in-scope item and file content from the Innovator database and sends it to an OpenAI-compatible embeddings API to generate vector representations. These vector representations are stored in the `[ais]` schema within the Innovator database; no separate database is created. The embeddings endpoint is configurable: on-premises customers can use an Aras-provided hosted embeddings service by obtaining the endpoint and credentials from Aras, or use any compatible provider. The endpoint is set via the `EMBEDDINGS_ENDPOINT` environment variable in the AI Search configuration.

Using AI Search



Where do I access AI Search? AI Search is available both from the search field in the Innovator header and from a dedicated AI Search page. Administrators additionally see a gear icon in the top-right corner of the dedicated AI Search page that opens the Admin panel.

What is the "Ready to Search" state? When you first open the dedicated AI Search page without having run a search yet, the results area shows a Ready to Search state with example query chips such as "123 module description," "Body parts with Preliminary state," "Cylinder head manual," and "Latest modified CAD documents." Select any chip to run that search immediately.

Is there a minimum query length? Yes. Queries must be at least 3 characters long. Submitting a shorter query returns the message: "Search query too short â€" Please enter at least 3 characters to search. Try a longer keyword or phrase."

What filters are available? The two available filter properties are Item Type and Is Current. Both use an equals operator. The Item Type list shows only the ItemTypes currently configured for indexing in your instance.

Filters follow a staged and applied model. Adding, editing, or removing a filter stages the change without immediately updating results. Staged filters appear as gray chips. Select **Apply Filters** to send the staged filters to the search service; applied filters appear as blue chips. Entering a new search query or pressing Enter also applies any staged filters.

Can I use more than one filter of the same type? No. Each filter property can be used once at a time. Once an Item Type filter is active, that property is not available in the Add Filter dropdown until the existing filter is removed.

What does the Load More button do? When more results are available beyond the first page, a Load More button appears at the bottom of the result list. Selecting it fetches the next page using the same query, applied filters, and sort order. Load More does not apply staged filter changes.

Why am I getting no results at all? This can happen for several reasons. If no ItemTypes have been configured for indexing yet, or the index is temporarily unavailable, you may see "The Index Is Not Ready Yet or Is Currently Unavailable" â€" wait and try again, or contact your Innovator administrator. If AI Search is running normally but your query returned nothing, you will see "No



Matches Found â€” We couldn't find anything matching your query." Rephrase using different or broader terms. If you see "Search query too short," enter a query of at least 3 characters.

What does "Search service is temporarily unavailable" mean? This message indicates a transient service error distinct from the index-not-ready state. Select Search again to retry.

What does "Invalid search syntax" mean? The query contains invalid characters or syntax. Remove or correct special characters and try again.

Why don't older versions of an item show up in search? AI Search only indexes the current version of an item and the current version of any indexed file attached to it. Earlier versions are not included in the index.



AI Search Release Notes

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

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.
-

