

VANTIQ 1.45 Update

VANTIQ Engineering & Product Management Enablement

Agenda

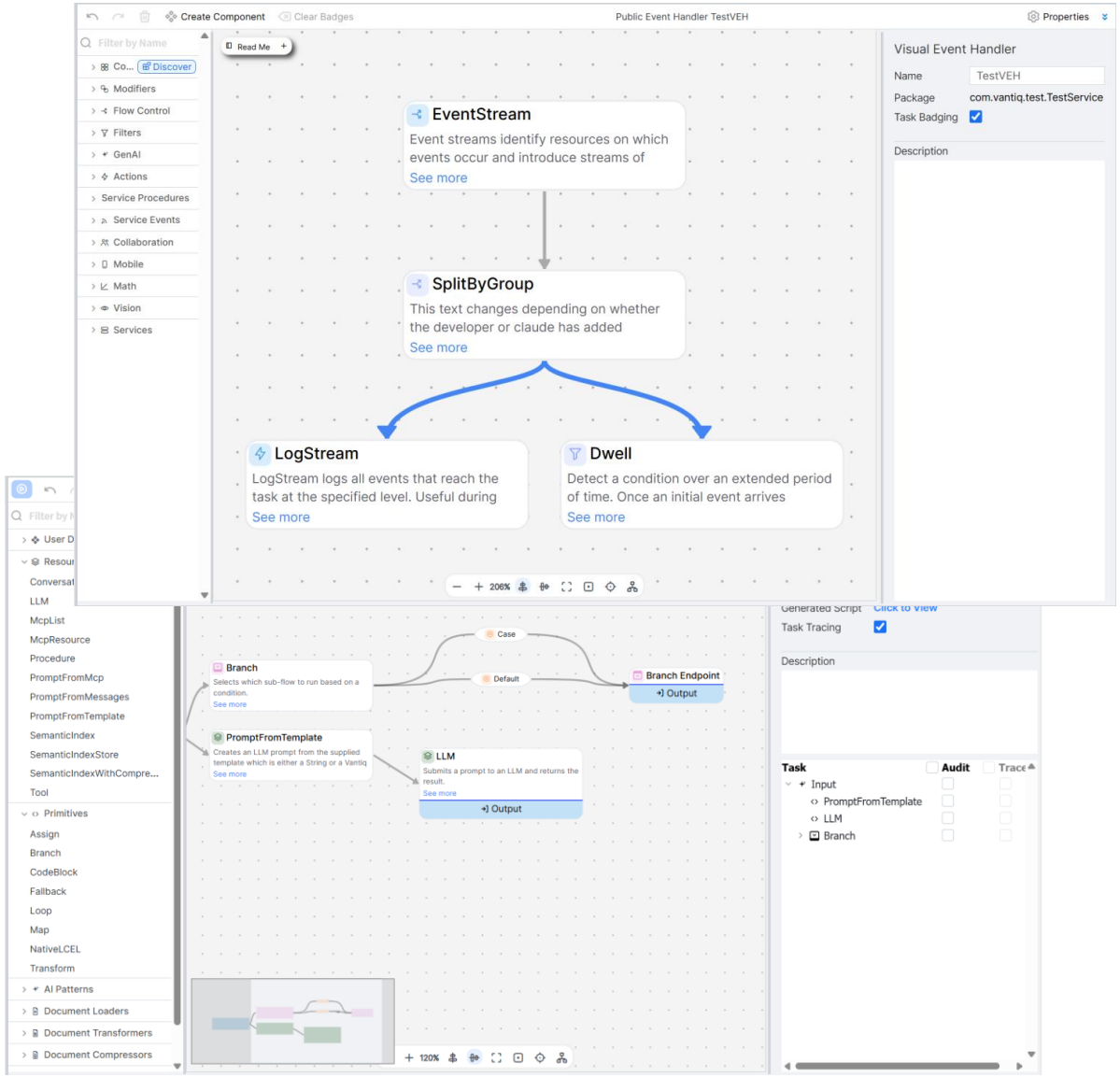
- Overview of New Features and Enhancements
- Visual Event Hander and GenAI Flow look and feel refresh
- New Prompt Editor
- New Python Services
- AI Development Tool/VIA Enhancements
- A2A, GenAI & MCP Support Enhancements
- IDE Enhancements
- Server and Operational Enhancements

Overview of New Features and Enhancements

Usability Enhancements - Refresh of the Visual Builders

Increased developer productivity

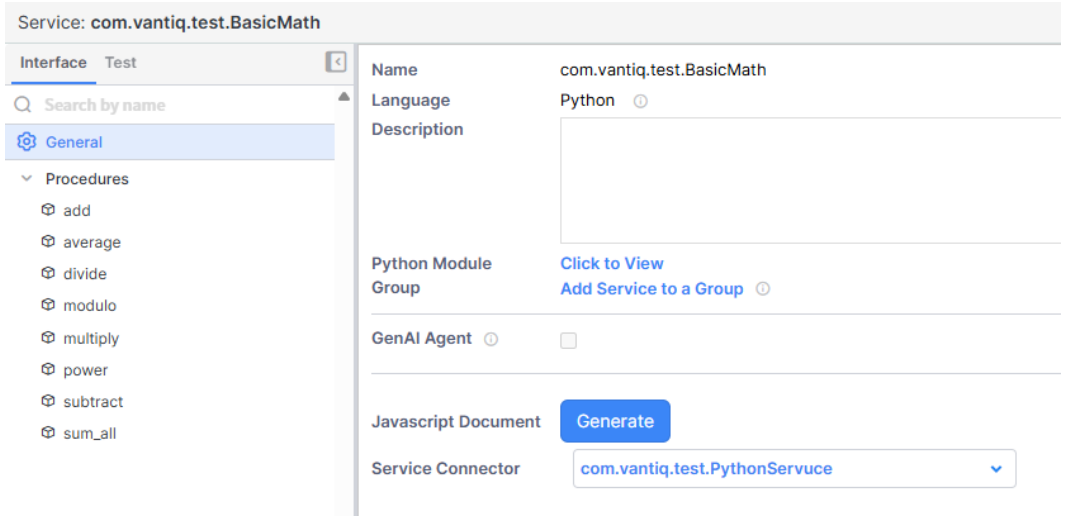
- New look and feel for the Visual Event Handlers and GenAI Flows
- Support horizontal and vertical layout
- Property sheet now always open by default
  - Old behavior still available as an option in user settings



Python Service - Python code running within the Vantiq platform

Increased integration capabilities

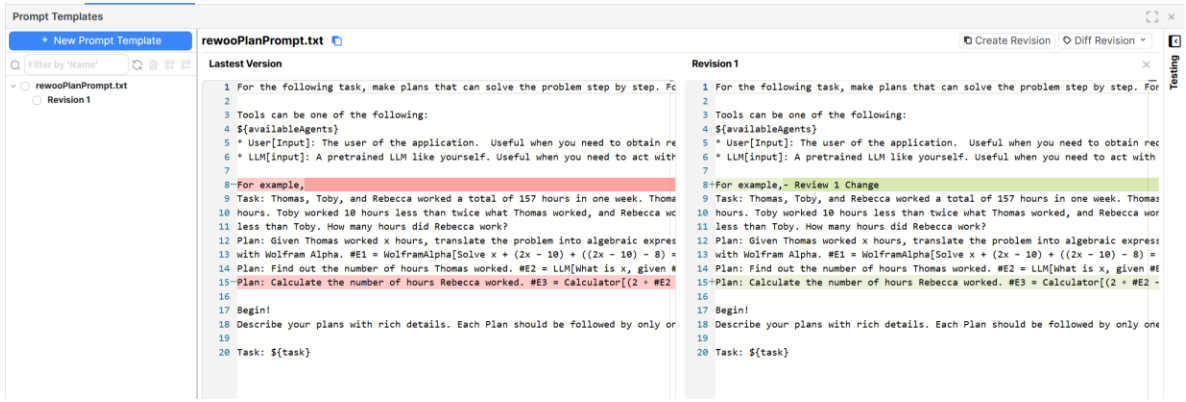
- New read-only Python Services
- Python code running within a new Python Service Connector
- All public functions exposed by a Python Module or Class visible
- The Service Connector image can be extended by building a new image with the required libraries

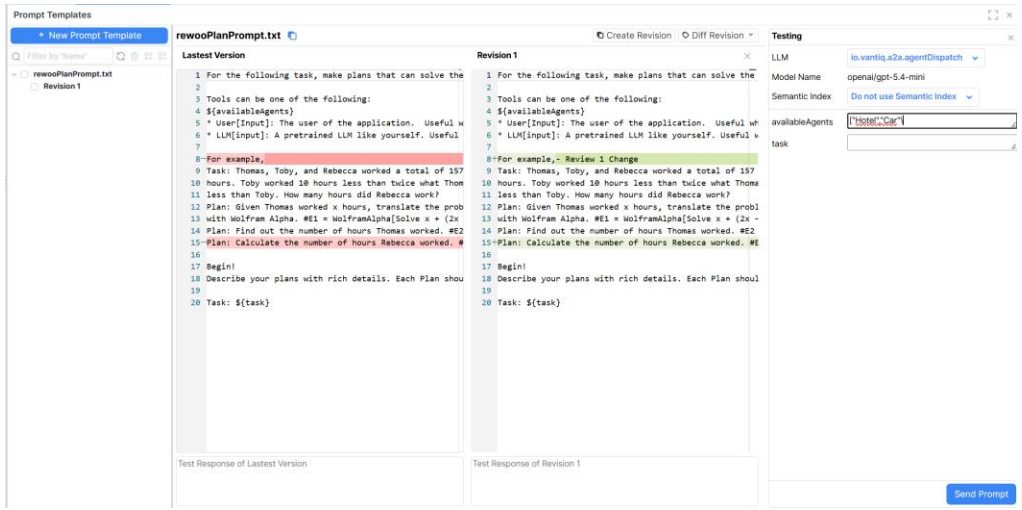


Prompt Management - New Prompt Template Editor

Increased developer productivity

- New tool to manage Prompts/Prompt templates
- Prompt editing
- Prompt revision support
- Prompt revision diffing
- Prompt testing and result comparison

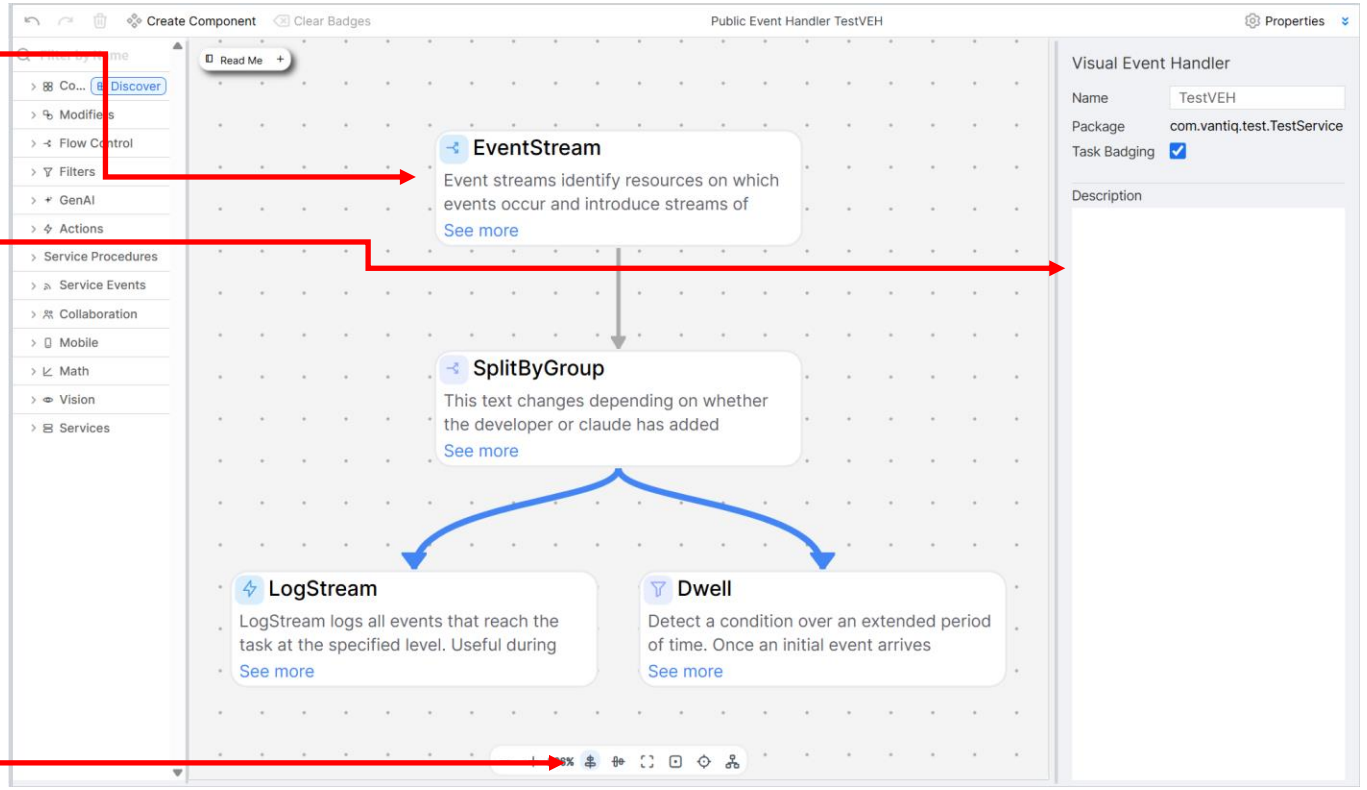


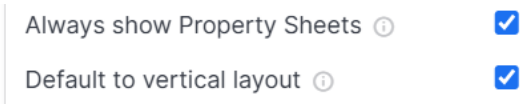


Visual Event Handler and GenAI Flow refresh

Visual Event Handler & GenAI Flows have new look and feel

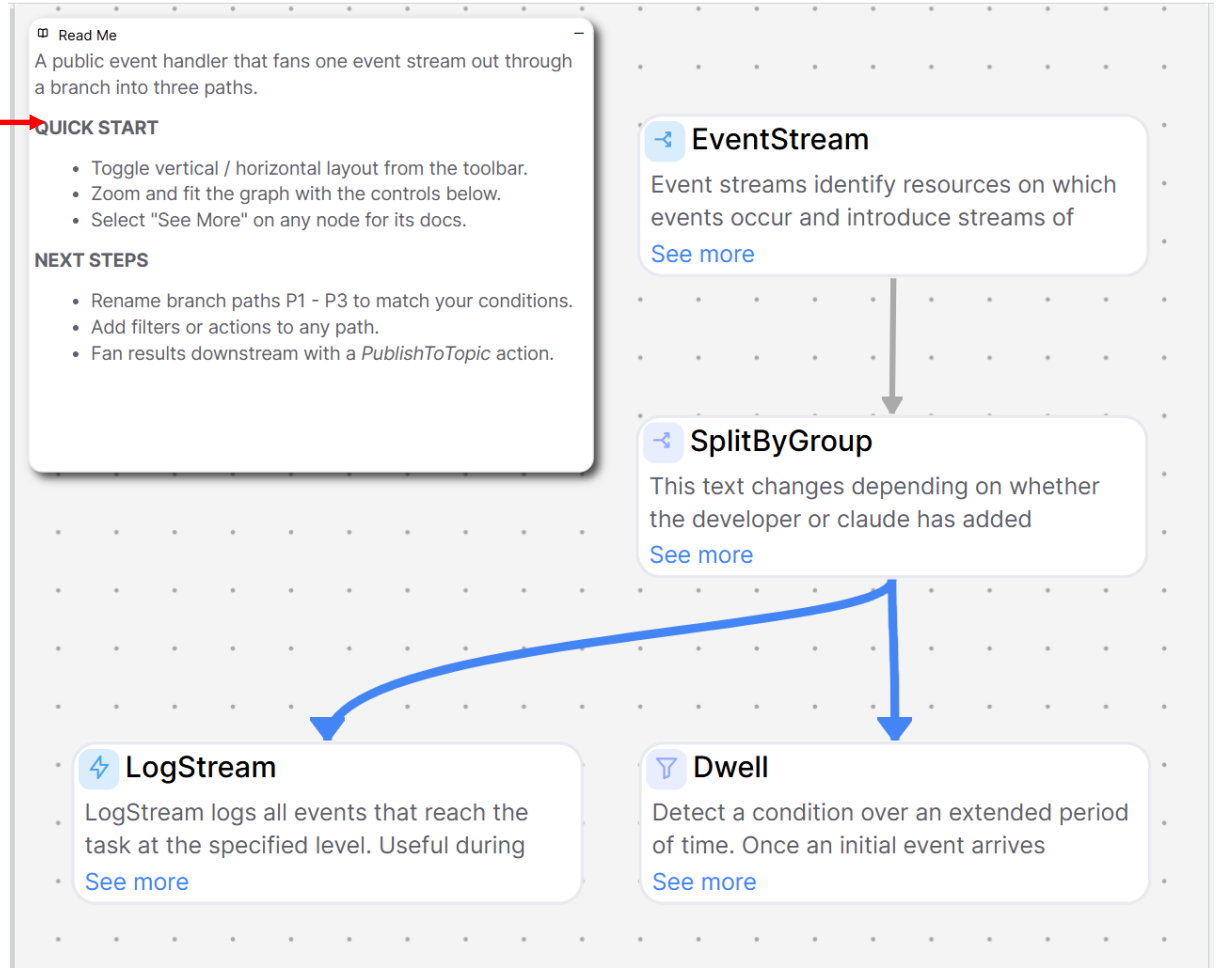
- New Look and Feel
- Property Sheet always open by default
- Vertical and Horizontal layout
- New IDE settings to change defaults





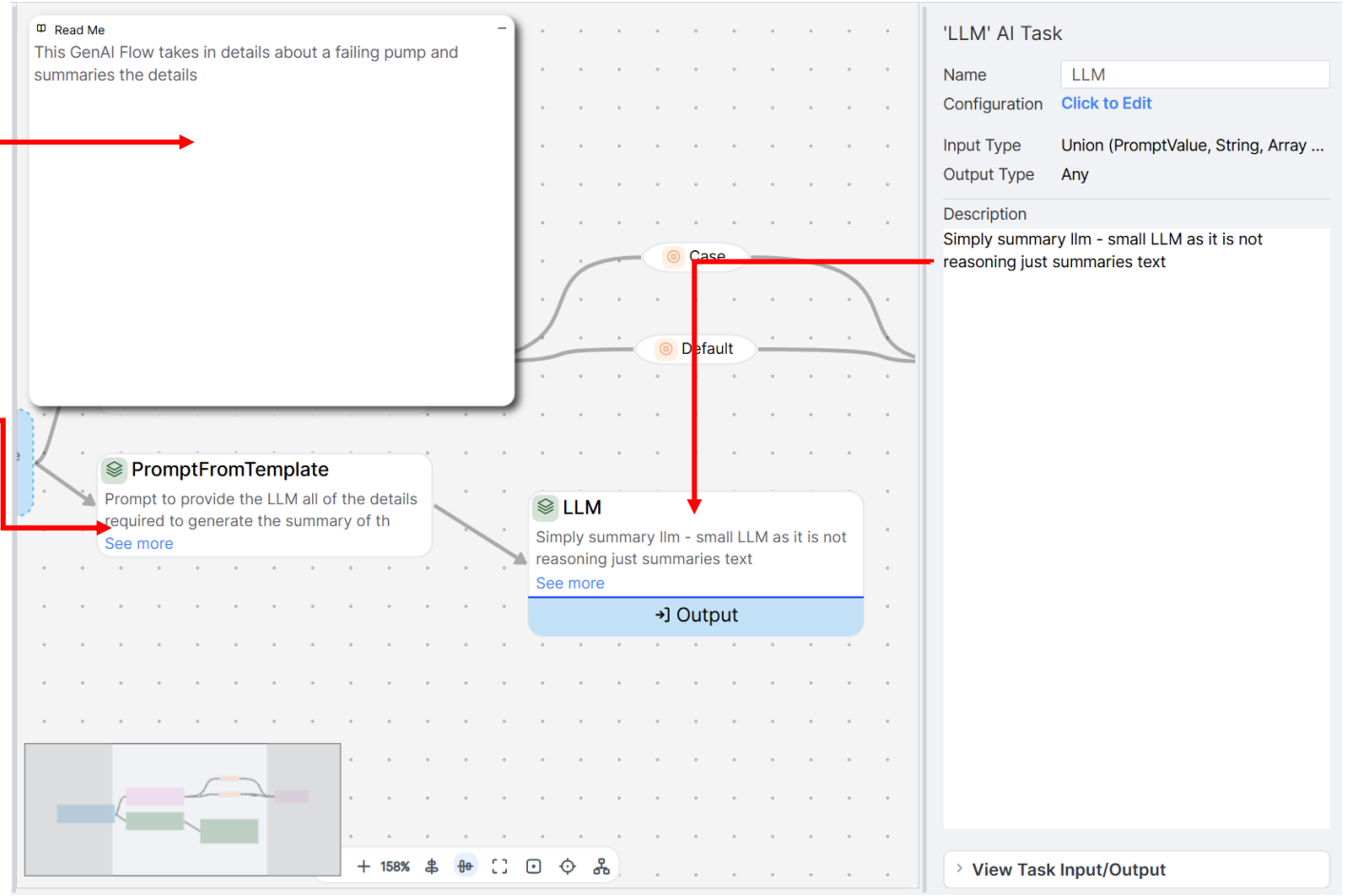
New user and information/help in graphs

- Read Me
  - Gives new users basic help
- Activity Pattern text
  - Provides new user an outline of each Activity Pattern's function



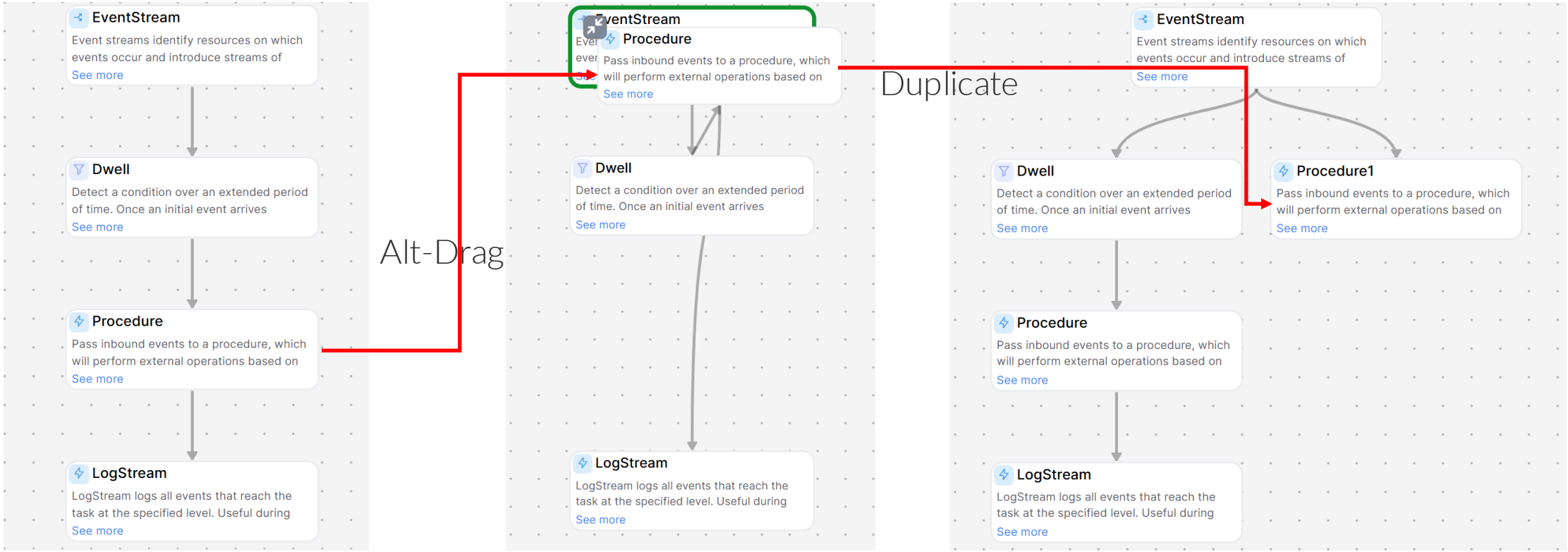
AI Assistants and diligent developers

- Read Me
  - Read me text comes from the Description for the VEH/GenAI Flow
- Task text
  - The Task text comes from the description for each task



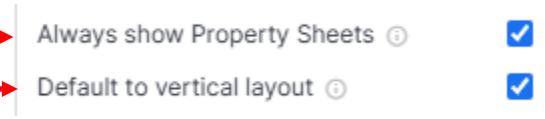
Copy nodes in graph

- Alt Drag & Drop will now copy/duplicate the node in the graph rather than reconnect it



Miscellaneous

- Renaming nodes in the graph doesn't force a re-layout
- New IDE Settings
  - Property sheet behavior can be set to 1.44 behavior
  - Default Layout style – Horizontal or Vertical



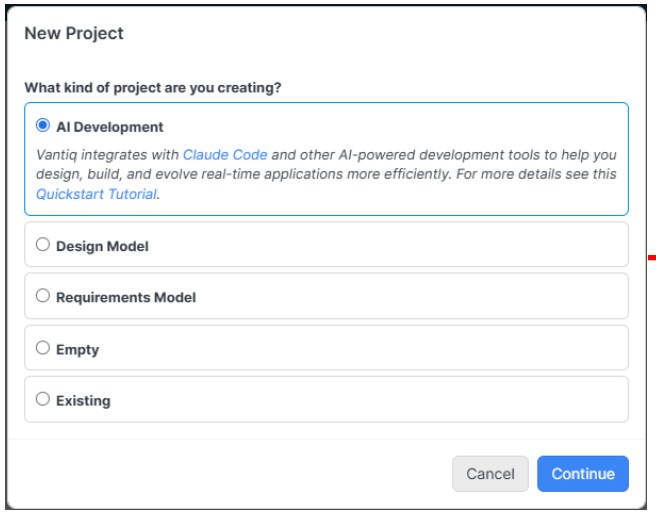
Claude and AI Developer Tool Integration

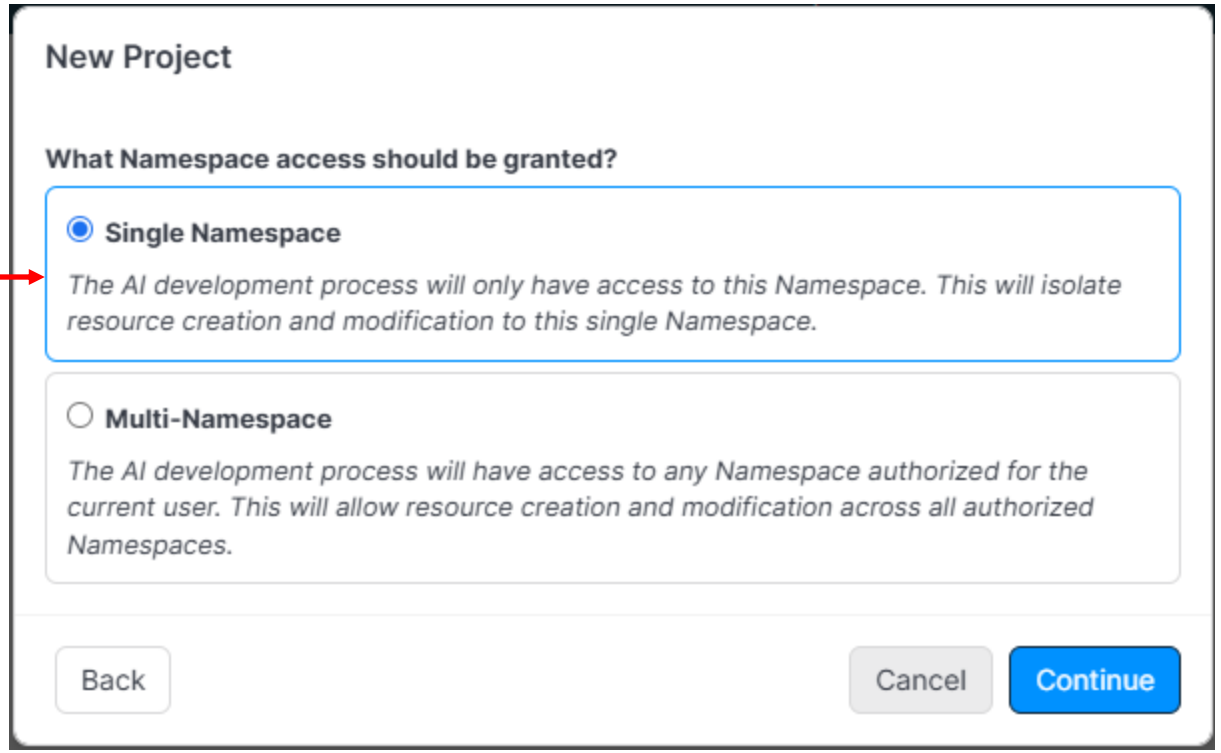
VIA upgrades

- VIA now supports for additional development resources
  - Python Services
  - Catalogs
  - Assemblies
  - Extension Sources
  - App and GenAI Components
  - Developer namespace creation

Multi or Single Namespace Tokens for VIA

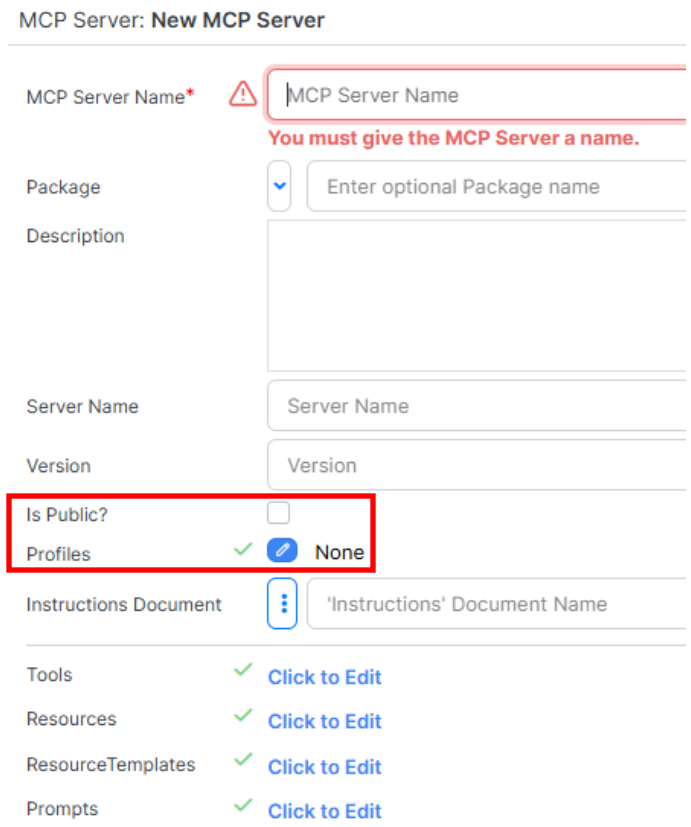
- When creating a new project that supports AI development (**VIA**), the IDE now asks whether you want a multi- or single-namespace token





Public MCP Servers

- It is now possible to mark an MCP server as ‘Public’
- Public MCP servers have the following capabilities/restrictions
  - Do not require an access token to communicate with them
  - By default, they run as the Public user
    - Can only serve ‘public’ documents as resources
    - Tools will run as the ‘Public’ user
  - Can associate them with a profile in which case the restrictions above are related to the profile set
- URI is <vantiqServer>/mcp/<namespace>/<mcpServerName>



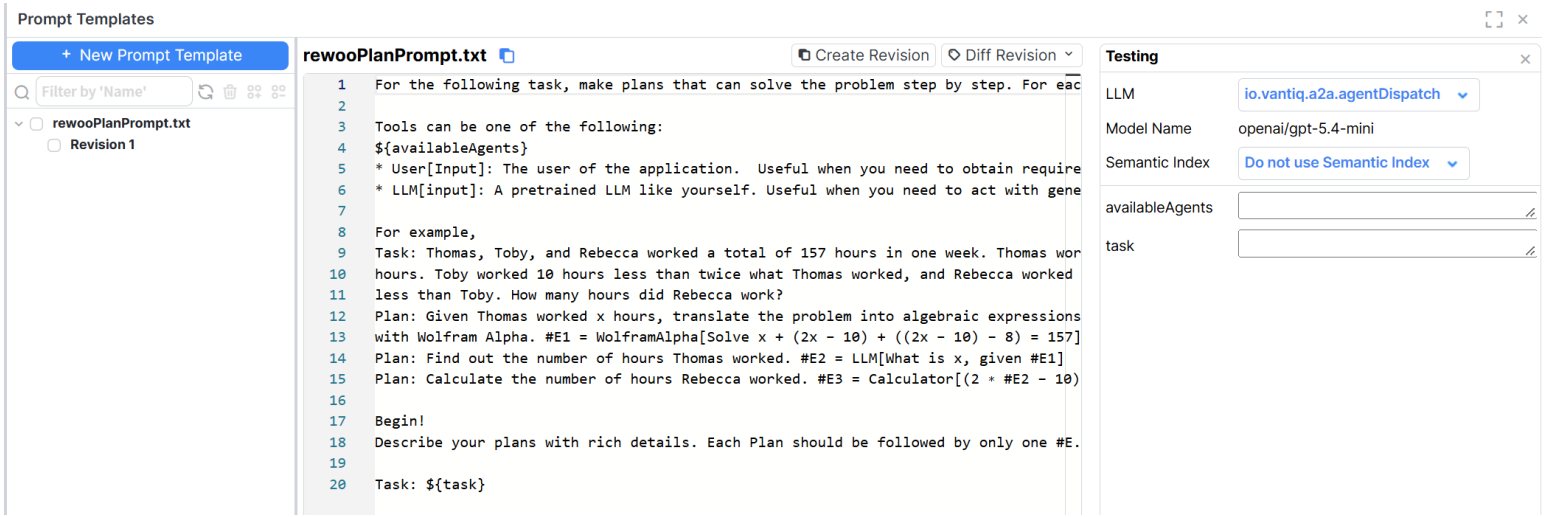
Miscellaneous VIA enhancements

- VIA now exports resource definitions using the same format as the IDE import/export rather than the raw resource format
  - VIA will reconcile the change in format
- VIA supports load/dump of SI and Types using TUS and the .dmp format
- VIA will now use project snapshots – equivalent to Project export zip files to bulk export the project to disk

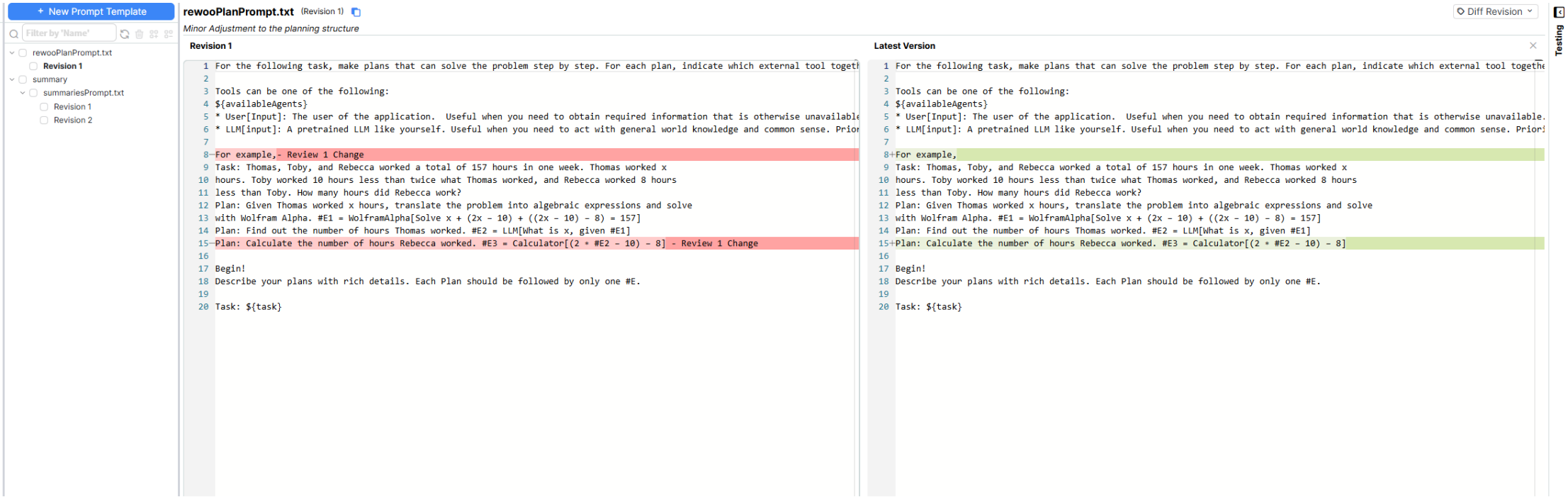
Prompt Template Editor

New Prompt Template Editor tool

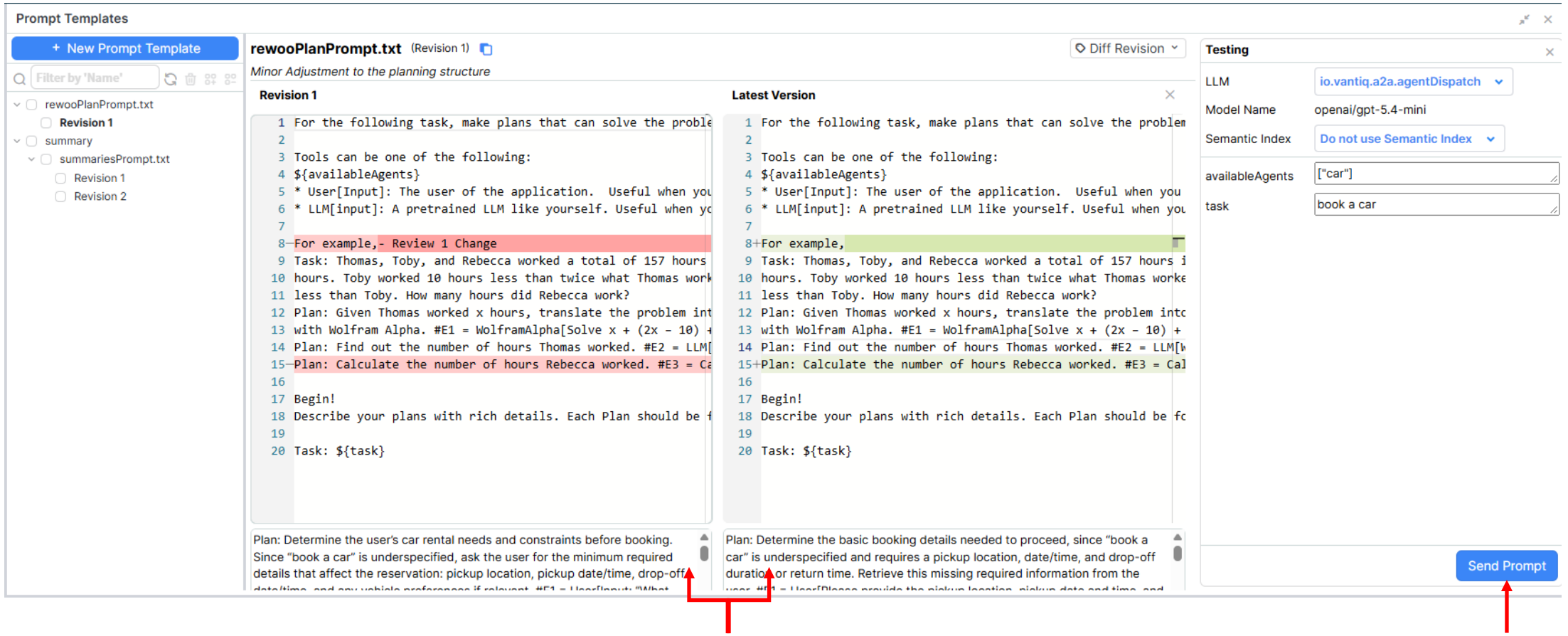
- Allows editing of prompts
- Supports multiple revisions of a prompt
- Supports diffing prompts
- Supports testing and comparison of prompt results - one or two revisions



Diffing two revisions



Testing one or more revisions

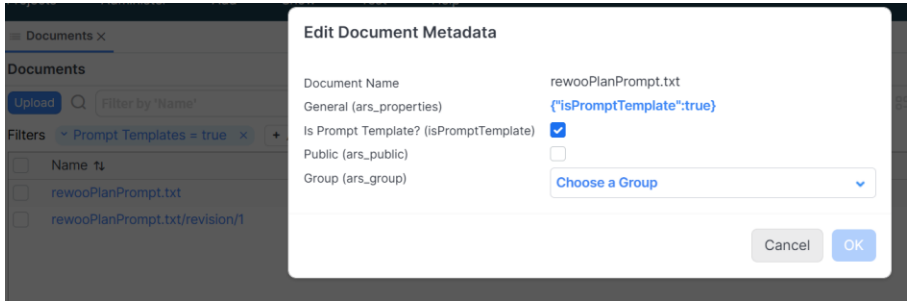


Result from each revision

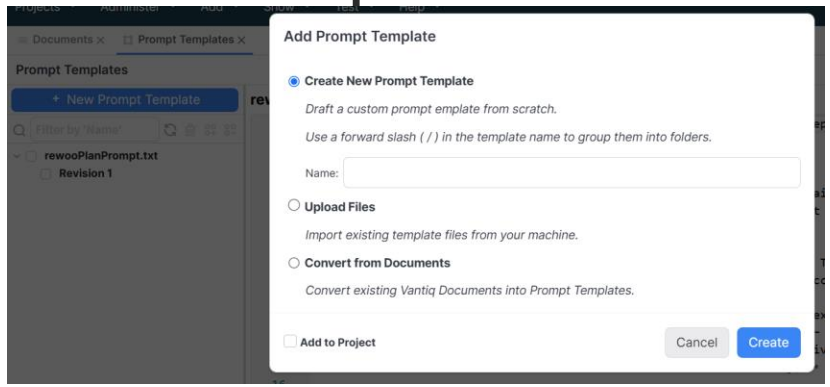
Send each revision with availableAgents & task as input

Prompt Management and revisions

- Prompts are still documents. Prompt documents have a new ars_properties field to identify which documents are prompts

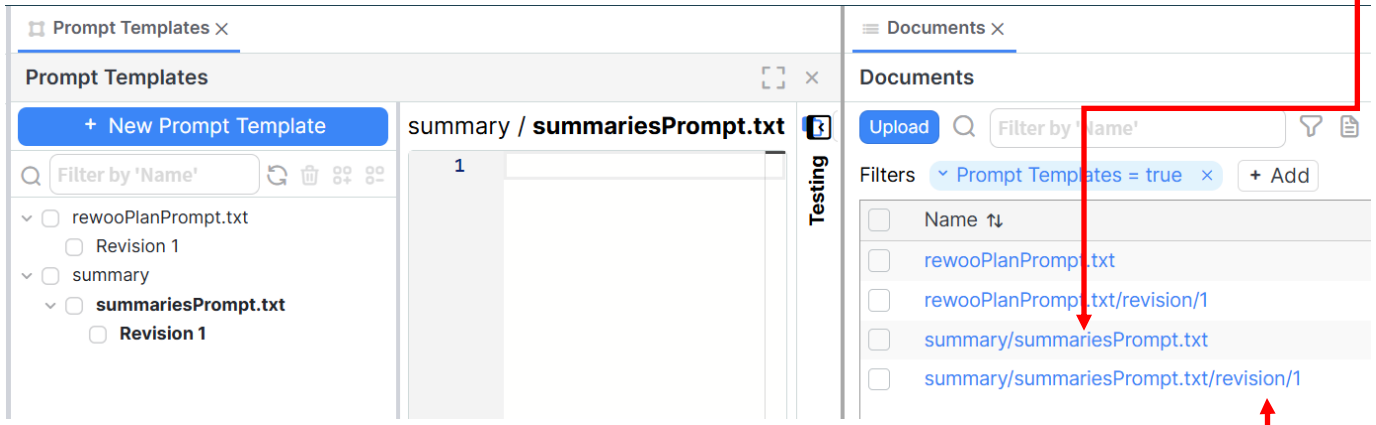


- When adding a Prompt in the editor the tools will update this field



Prompt Revisions - structure

- Live instance remains the same – e.g. ./docName.txt or md
- GenAI Flows and documentReferences do not have to change

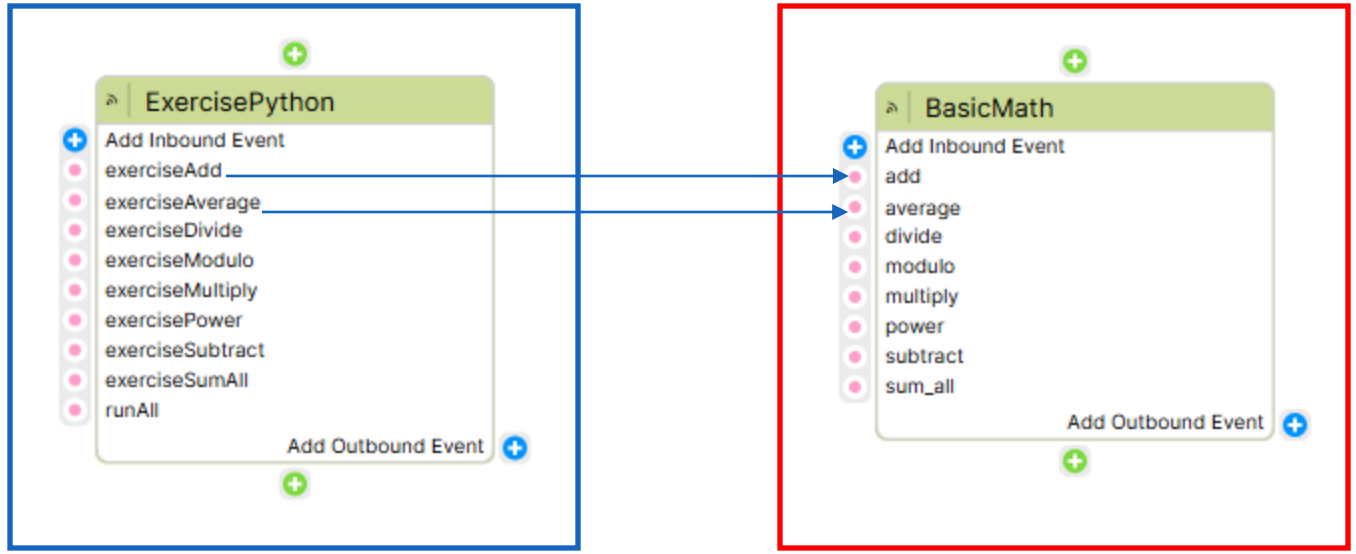


- Revisions are separate documents with the structure ‘revisions/{no}’

Python Service

Python Service

- The Python Service is a combination of a Python Service Connector and a read-only Vantiq Service

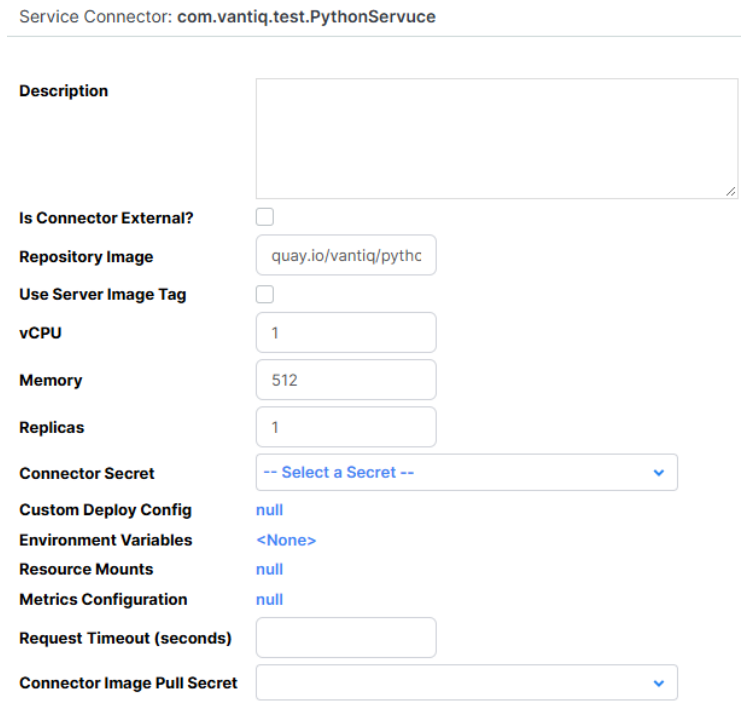


Std Vantiq Service

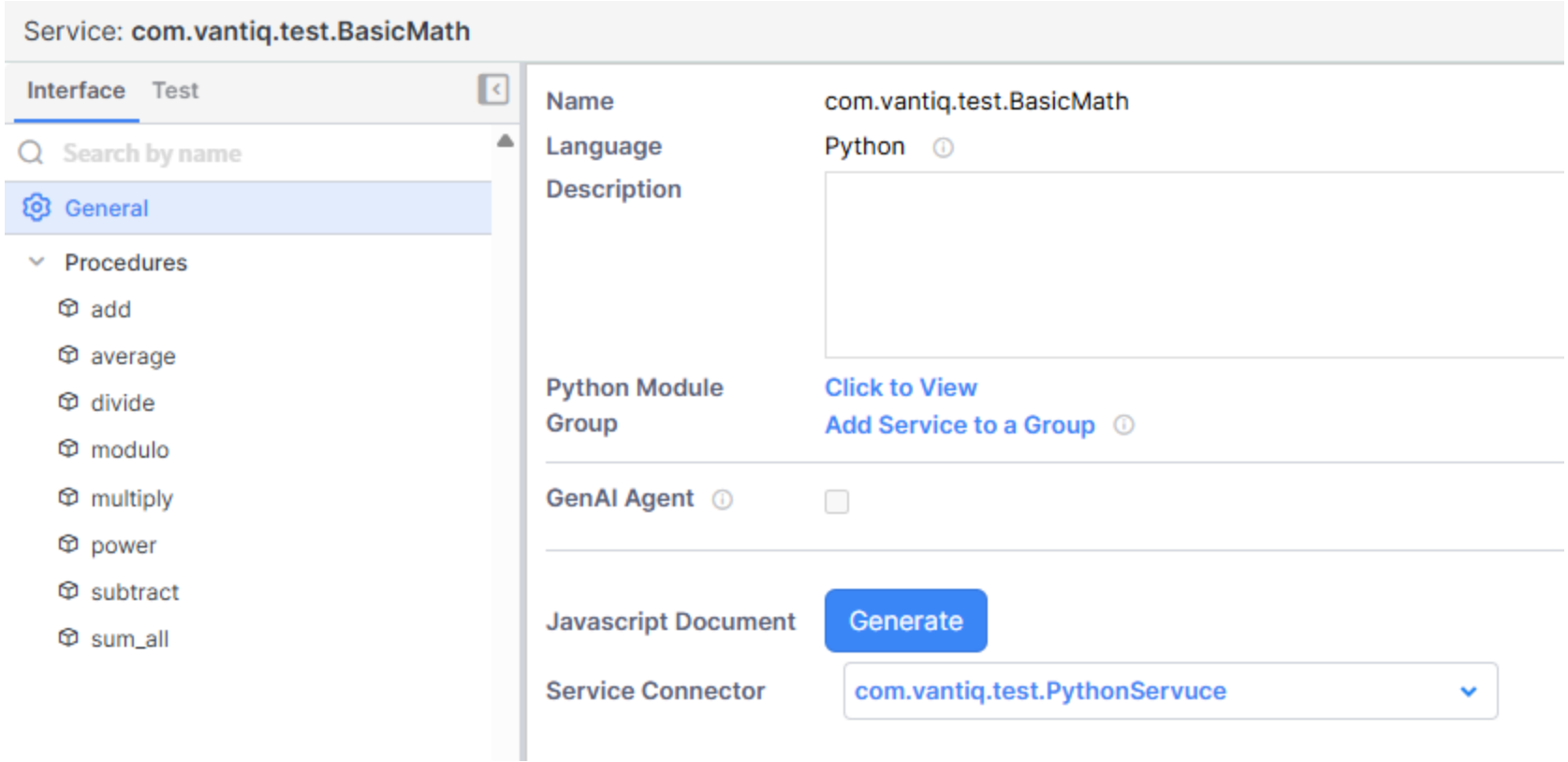
Actual Python code running in Service connector

Service Connector

- Add a service connector to your namespace or organization
- With the image quay.io/vantiq/pythonservice:1.45.0

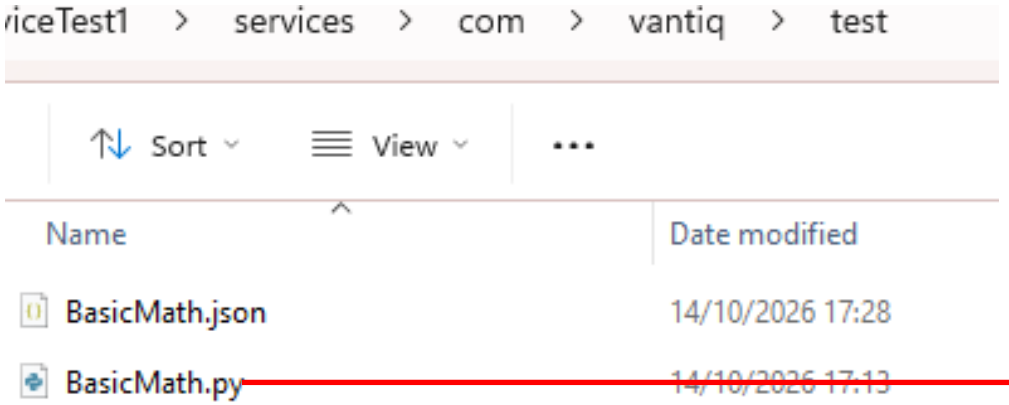


Use Claude and VIA to create the python service



Claude & VIA

- User: ‘create me a python service in Vantiq that defines a series of functions for performing basic maths like add and subtracting numbers’




```
D: > LocalData > ClaudeCLI > PythonServiceTest1 > services > com > vantiq > test > BasicMath.py
"""Basic arithmetic operations exposed as Vantiq procedures.

Vantiq service: com.vantiq.test.BasicMath
Connector:      com.vantiq.test.PythonServuce
"""


def add(a: float, b: float) -> float:
    """Return the sum of a and b."""
    return a + b


def subtract(a: float, b: float) -> float:
    """Return a minus b."""
    return a - b


def multiply(a: float, b: float) -> float:
    """Return the product of a and b."""
    return a * b


def divide(a: float, b: float) -> float:
    """Return a divided by b. Raises an error if b is zero."""
    if b == 0:
        raise ValueError("Cannot divide by zero")
    return a / b


def power(base: float, exponent: float) -> float:
    """Return base raised to the given exponent."""
    return base ** exponent


def modulo(a: float, b: float) -> float:
    """Return the remainder of a divided by b. Raises an error if b is zero."""
    if b == 0:
        raise ValueError("Cannot take modulo by zero")
    return a % b


def sum_all(values: list[float]) -> float:
    """Return the sum of a list of numbers."""
    return sum(values)


def average(values: list[float]) -> float:
    """Return the arithmetic mean of a list of numbers. Raises an error if the list is empty."""
    if not values:
        raise ValueError("Cannot average an empty list")
    return sum(values) / len(values)
```

Python Code

- The Python code must be either a Python module or a Python class
  - For Python modules each public top-level function becomes a public procedure of the service
  - For Python classes each public method becomes a public procedure of the service
- Parameter and return types come from Python type hints
- Python code can call back into vantiq using client.context
  - Use `client_context.get_vantiq_client()` to obtain a Vantiq client, then use its asynchronous operations - `select`, `select_one`, `insert`, `update`, `upsert`, `delete`, `count`, `execute`, `publish`

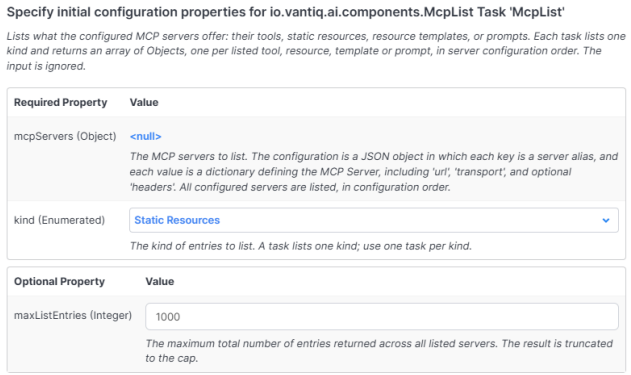
Python Service

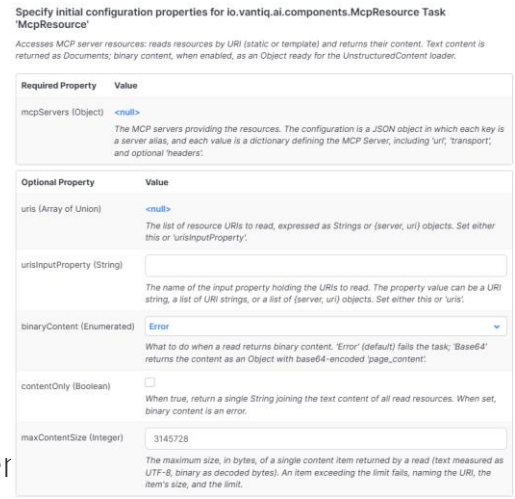
- Call the procedures/functions just like you would VAIL Procedures
  - Vail - Execute
  - Visual Event Handler – Procedure
  - GenAI Procedures – Tools
- Deployed just like other services
- The Service Connector must exist before creating and deploying
- If you need custom libraries define your own Python Service Connector image

GenAI Enhancements

MCP Resource Components

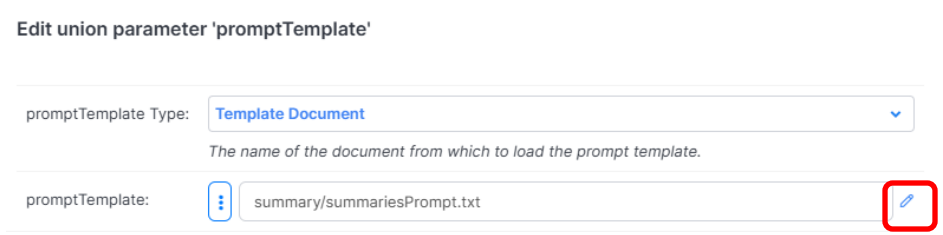
- Two new components added to the GenAI Builder to allow for access to MCP artifacts
  - McpList – Looks up the list of MCP artifacts – resources, resourceTemplates, tools and prompts
  - McpResource – Looks up a particular resource

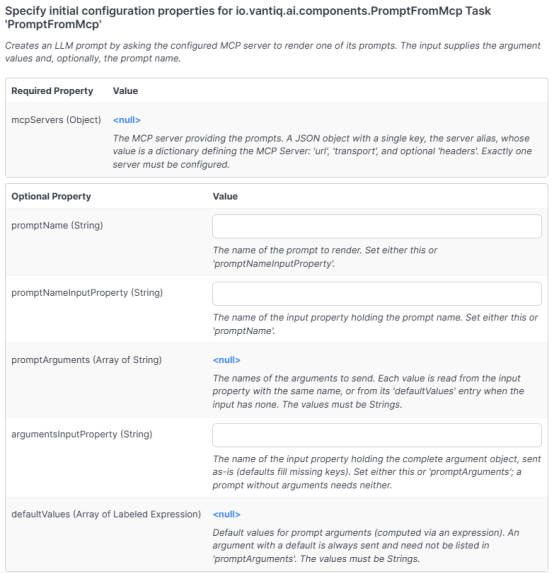




Prompt Management

- Launch basic prompt editor from within GenAI
  - PromptFromTemplate
  - PromptFromMessage
  - PromptFromMCP
- PromptFromMCP now allows lookup of Prompt from MCP server





Prompts referencing TempBlobs and Documents

- Prompts can now reference any content type, not only an image: a PDF or another file is sent to the model as a file

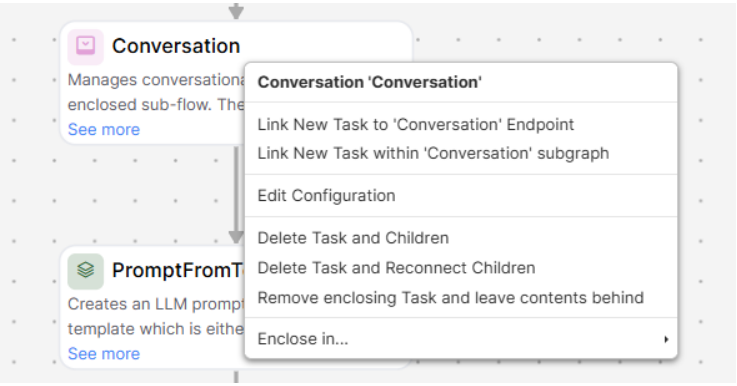
```
var msg = io.vantiq.ai.ChatMessage.buildHumanMessage([
    {type: "text", text: "Summarize this report in three sentences."},
    {type: "file", url: "system.documents/reports/q3.pdf"}
])
```

Fallback Component

- The Fallback component has a new configuration property - `exceptionKey` which holds the Python Exception string

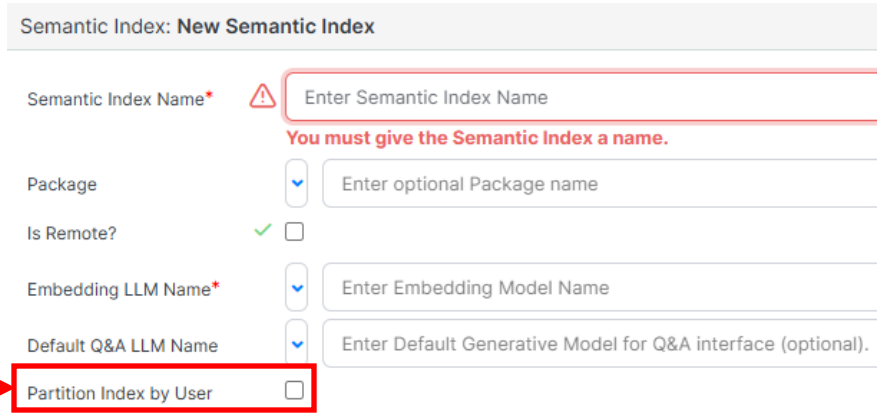
GenAI Builder

- Panning/zooming will not force a save.
  - If you want to save changes to the layout then make a material change for the Save button to appear
  - Behaviour same as 1.44 Visual Event Handler
- New Context menu on simple sub graphs – ‘Remove enclosing task and leave contents behind’ – reverse of Enclose in



Semantic Index - partition by user

- When creating a Semantic Index it can now be configured to partition entries by ‘user’
- Entries will only be visible to that user
- VEH etc that do not run as a user can use the entryOwner parameter to set the ownership of an entry
- The SemanticSearch functions now have a ‘entryOwner’ parameter
  - `answerQuestion`
  - `similaritySearch`
  - `retrieveChunks`

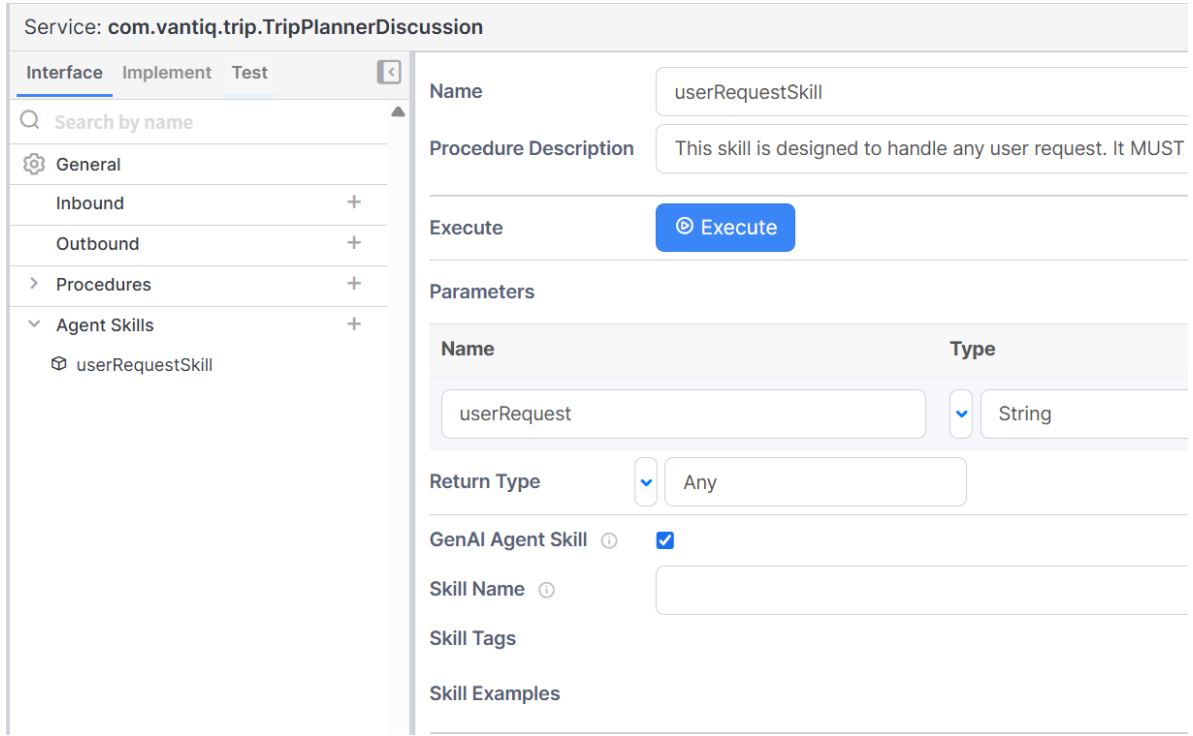


A2A Enhancements

Service Builder - Skills new separate accordion in Interface

- For GenAI Services Skills are now in a separate accordion on the interface tab

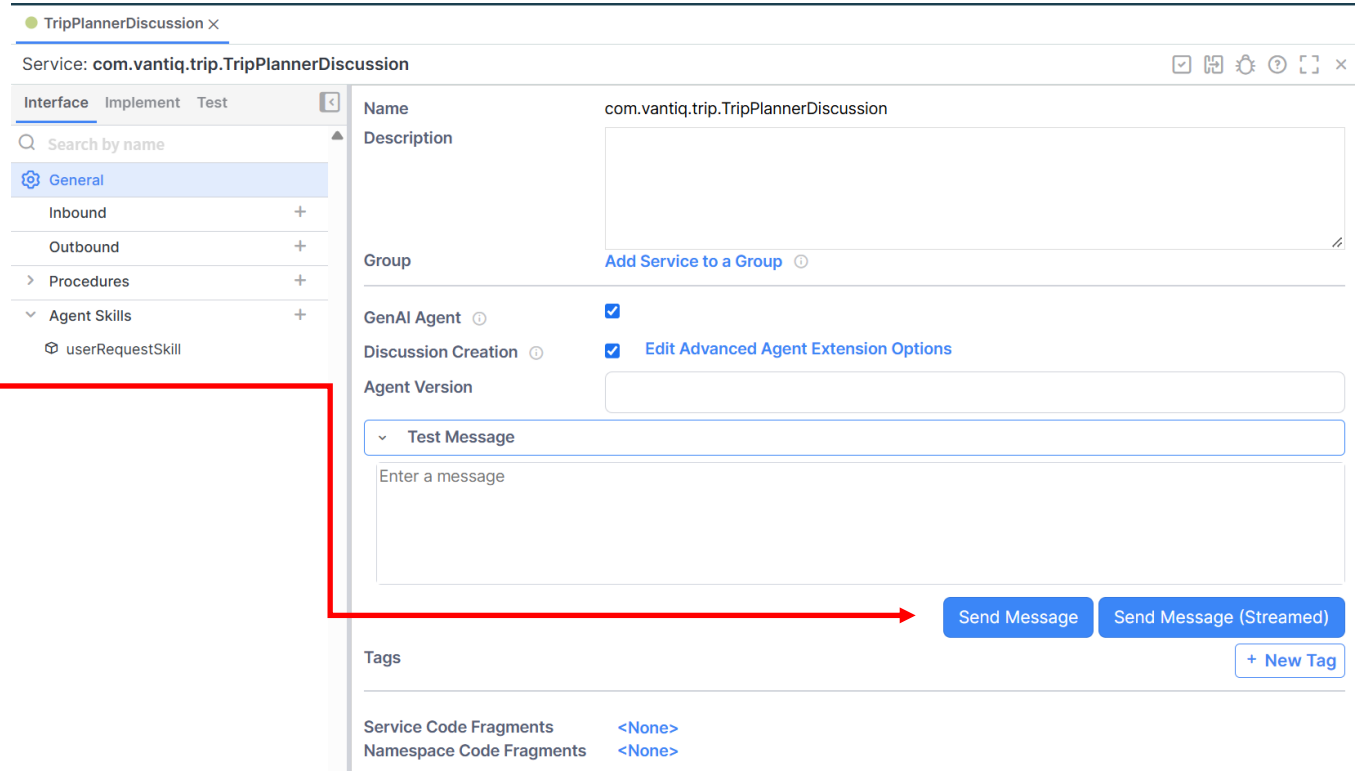
Agent Skills



Service Builder - GenAI Agents - sendMessage

- For GenAI Services it is now possible to execute a sendMessage/Stream from the Service Builder

Execute an A2A sendMessage/Stream to the Agent



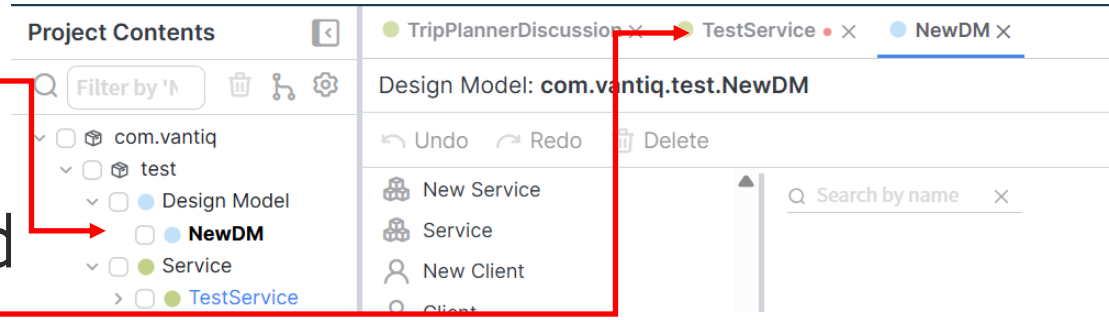
Miscellaneous Enhancements

- A2A implementation is now A2A V1.0 compliant
- HTTP+JSON Binding support for GenAI Agents
  - Use the following URL `https://dev.vantiq.com/a2a/<agentName>/message:send` or `/message:stream`
  - See the A2A spec for full set of capabilities
- messageSend/Stream now support new option – `returnImmediately`
- A2A Tasks now support pagination when retrieving them via HTTP+JSON binding

General Usability Enhancements

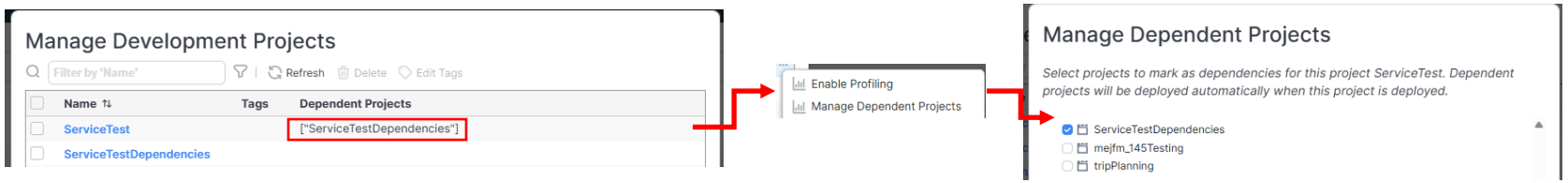
Project Enhancements

- The selected resource is now highlighted
- Each resource tab now has a colored dot or icon to indicate the type



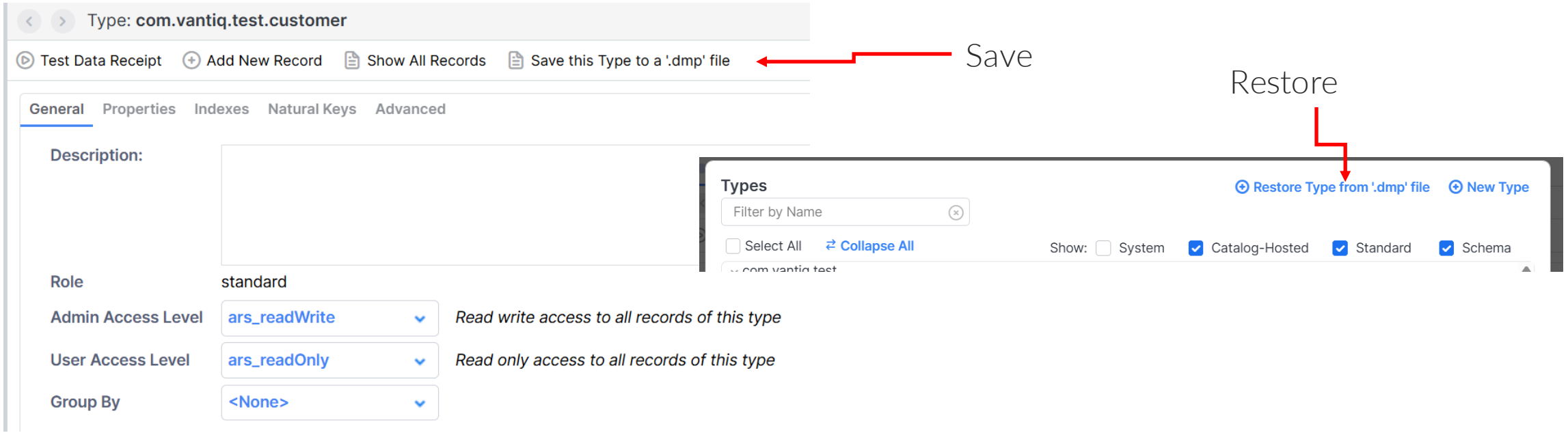
Dependent Projects

- Dependency Management changes
  - Project will no longer automatically includes dependent resources
  - This removes an issue with Git etc. integration when resources in the current project referenced shared resources in another project
    - Prior to 1.45 the dependent resources were auto-imported, which involved manual changes before executing git commands
- Dependent Project allow you to define the dependencies relationships between projects – this is used by the deployment tool to include these resource during deployment



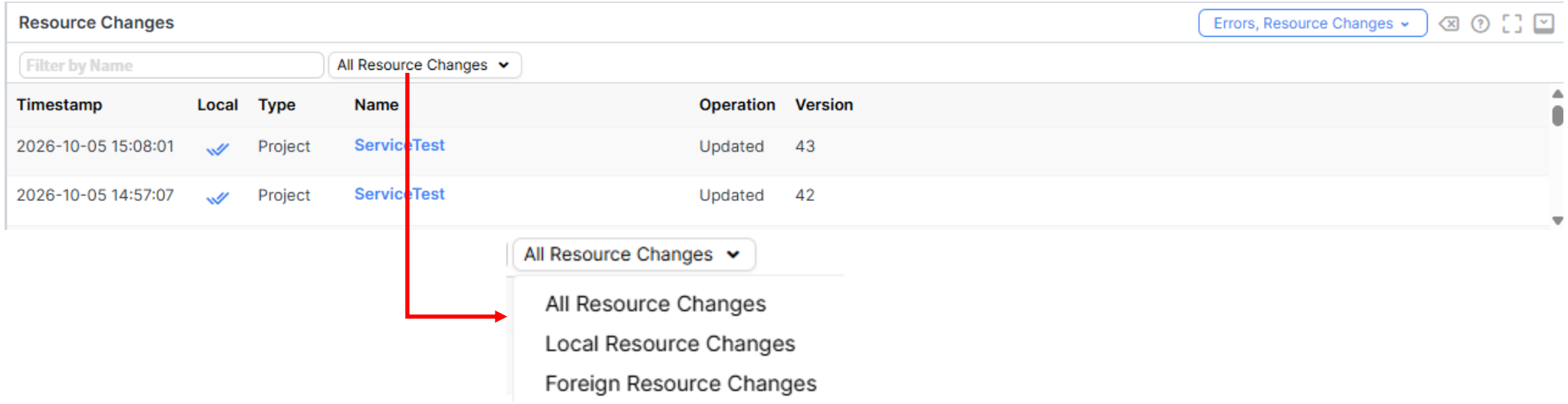
Dump and Restore Types

- There is now an option in the Type resource and Add Type popup to Dump and Restore types to/from a dmp file - These dmp files have same format as CLI dumps



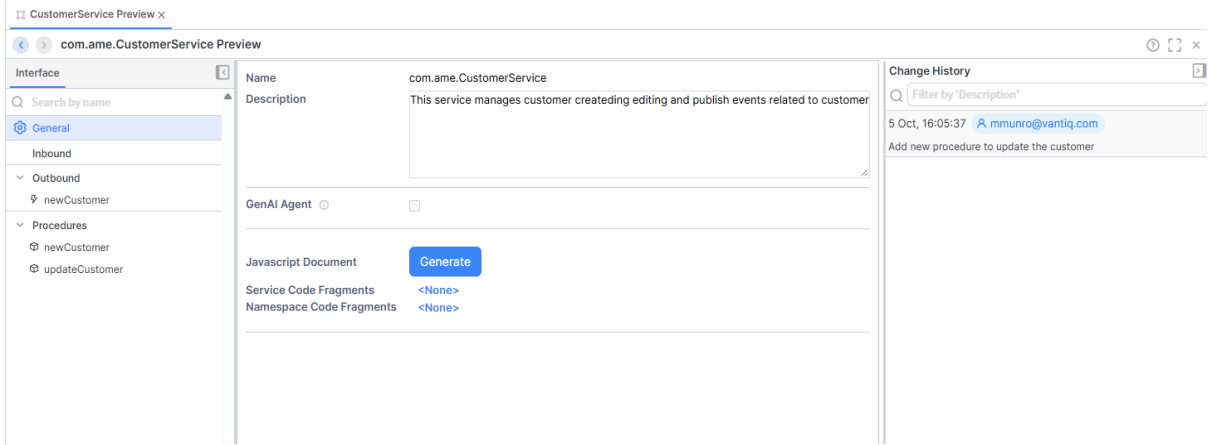
View Changes made to resource in project

- New option in Error/Log/Autopsy pane to show Resource Changes
  - Show all changes or changes just made by AI Assistants
  - Primarily added to allow developers to see what changes AI Assistants have made



New Catalog Interface

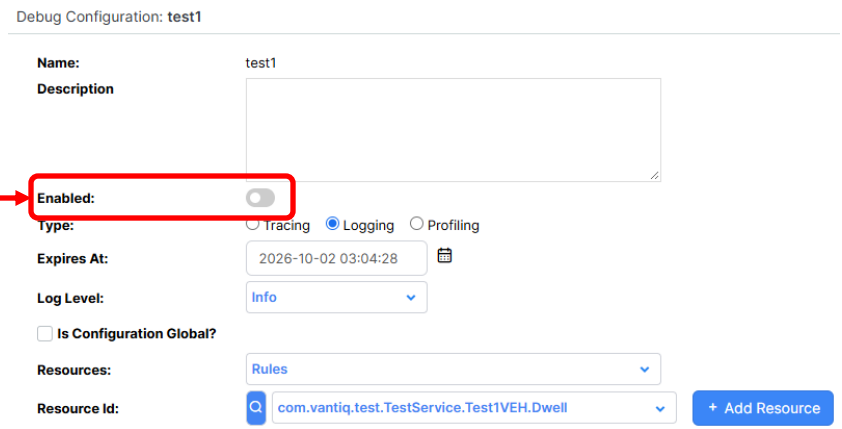
- The Catalog interface has been rationalized and harmonized
  - Each catalog entry type now opens a ‘next’ pane
  - Opens the the Service/Event/Assembly/Semantic index in read only view (does not import it)
  - Show Change History





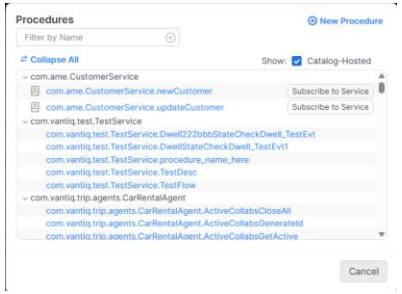
Miscellaneous Enhancements

- Debug Configurations can now be enabled/disabled – previously you had to delete them
- Show->Find Records doesn't lose JSON Where clause when switching between simple and {} Where Clause
- Project import – when you click on the drop area it now opens the Finder or Explorer to find the zip file

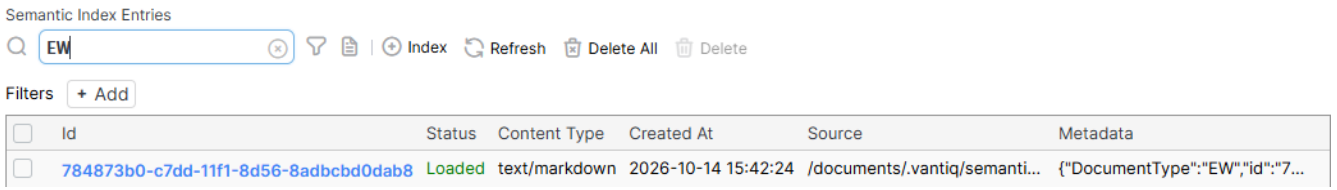


Miscellaneous Enhancements cont.

- All resources now have a Description and this can now be set from within the IDE
- Procedure list in Procedure Editor when using ‘execute’ now shows a popup rather than the built-in editor list
  - Provides more flexibility in filtering and organizing the list of procedures

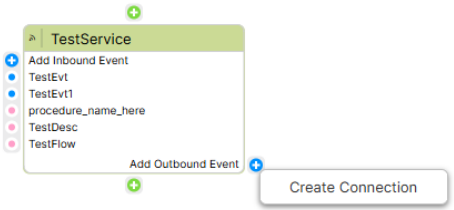


- Semantic Index – Filter by Metadata – does a text search on the Metadata and filters results

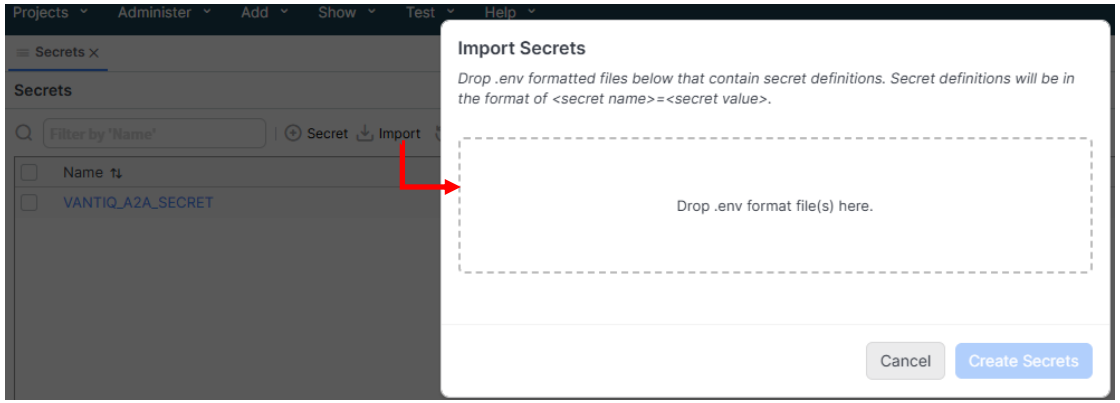


Miscellaneous Enhancements cont.

- Design Model – Routes can now be created through right-click and ‘Create Connection’ menu option rather than just drag and drop

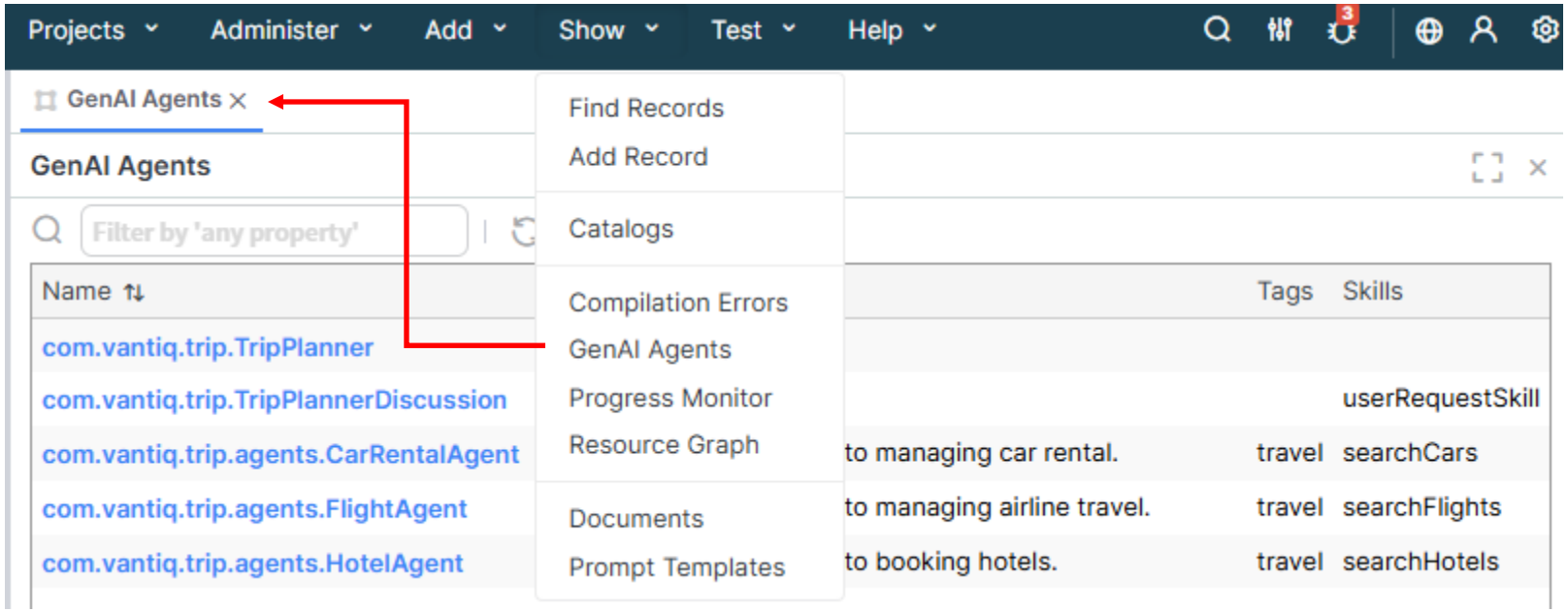


- Import secrets from a .env style text file



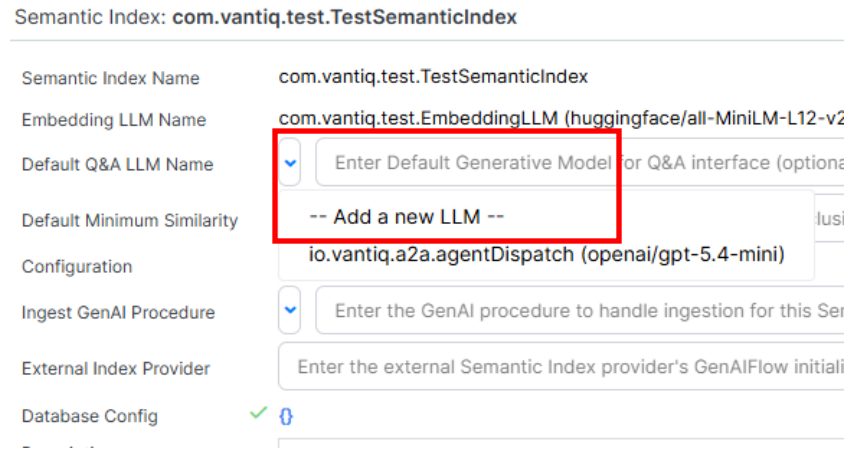
Miscellaneous Enhancements cont.

- List Gen AI Agents



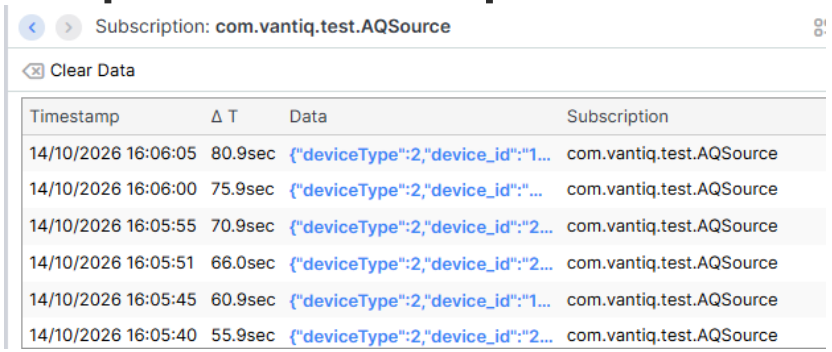
Miscellaneous Enhancements cont.

- Semantic Index – Add a new LLM



Miscellaneous Enhancements cont.

- Column order in Subscription/Test Data Receipt – Subscription now last

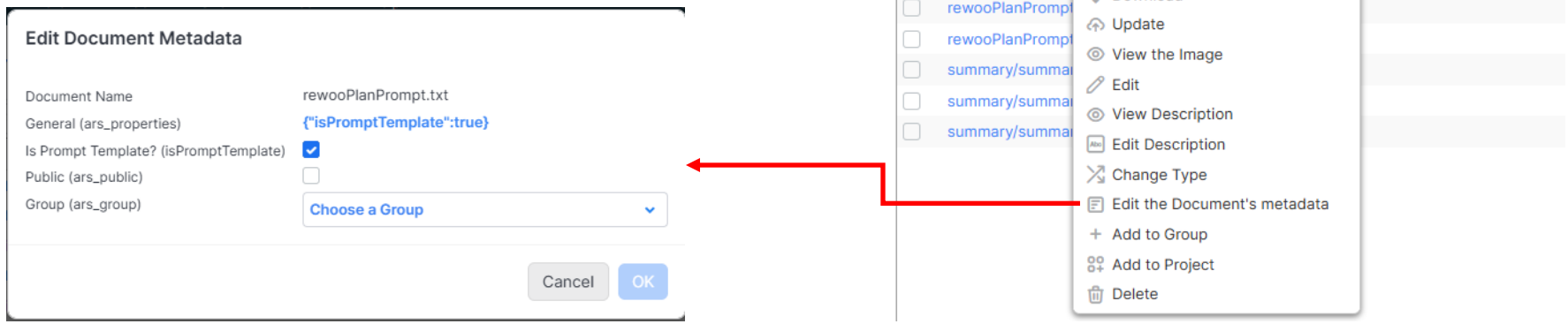


- MCP Server pane now provides a button to generate the .mcp.json for your MCP servers



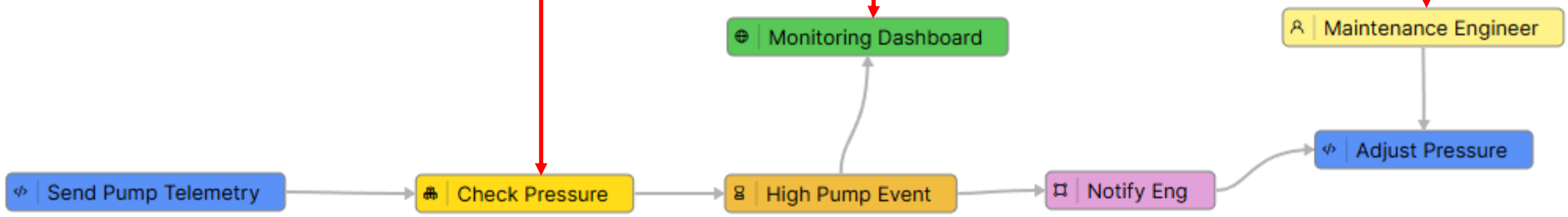
Miscellaneous Enhancements cont.

- Show Documents list now has additional context menus
  - View and Edit Descriptions
  - Edit the Document's metadata



System Modeler

- System Modeler now support 3 additional Event Storming artifacts
  - Actors
  - Read Models
  - Aggregates



Email Source

Email source - Sending attachments

- Send attachments using the attachment property

```
PUBLISH { text: "This month's report is attached.",
          attachments: [ { ref: "/system.documents/report.pdf" },
                         { content: "id,name\n1,widget", name: "data.csv", contentType: "text/csv" } ] }
  TO SOURCE MyEmailSource
  USING { from: "me@mydomain.com", to: "user@domain.com", subject: "Monthly report" }
```

Email source - Receiving email

- Add an IMAP property to the Email source to read emails

```
"imap": {
   "host": "imap.domain.com",
   "port": 993,
   "username": "support@domain.com",
   "password": "/system.secrets/MailPassword",
   "passwordType": "secret",
   "pollingInterval": 60,
   "properties": {
      "mail.debug": "true",
      "connectiontimeout": "20000"
   }
}
```

- Receive email via an event on the source

```
RULE onInboundMail
WHEN EVENT OCCURS ON "/sources/MyEmailSource" AS event

var message = event.value
log.info("Received {} from {}", [message.subject, message.from[0]])
```

Email Source - Support for OAuth

- Many email servers require OAuth
- Define an OAuth Remote Source

```
"name": "GmailTokens",
"type": "REMOTE",
"config": {
   "uri": "https://oauth2.googleapis.com/token",
   "oauthGrantType": "refresh_token",
   "oauthClientId": "<client id>.apps.googleusercontent.com",
   "oauthClientSecret": "@secrets(gmailClientSecret)",
   "oauthRefreshToken": "@secrets(gmailRefreshToken)"
}
```

- Configure the Email source to authenticate via Remote Source

```
"name": "MyGmailSource",
"type": "EMAIL",
"config": {
   "host": "smtp.gmail.com",
   "port": 587,
   "username": "user@gmail.com",
   "from": "user@gmail.com",
   "oauthSourceName": "GmailTokens",
   "imap": {
      "host": "imap.gmail.com",
      "port": 993,
      "username": "user@gmail.com",
      "oauthSourceName": "GmailTokens",
      "pollingInterval": 60
   }
}
```

Server/VAIL Enhancements

New or enhanced built-in procedures for object and array manipulation

- Add several new built-in procedures
  - Object
    - findAll(obj, lambda)
    - remove (obj, lambda)
  - Array Procedures
    - find(arr, lambda)
    - findAll(arr, lambda)
    - remove (arr, lambda)
    - replaceElements(arr, conditionLambda, transformationLambda)

New or enhanced built-in procedures for object and array manipulation - example

com.vantiq.test.TestService.findAllTest | Insert Code Fragment

Create Test | Quiet Execute | Tracing | Log Level: Info | View: Errors | Autopsies

```
1  package com.vantiq.test
2  stateless PROCEDURE TestService.findAllTest()
3
4
5  var testArray = ["abc", "abcd", "def"]
6
7  var findRes = findAll(testArray, (val) => {
8      var ret = false
9      if (startsWith(val, "ab")) {
10         ret = true
11     }
12     return ret
13 })
14
15 return findRes
```

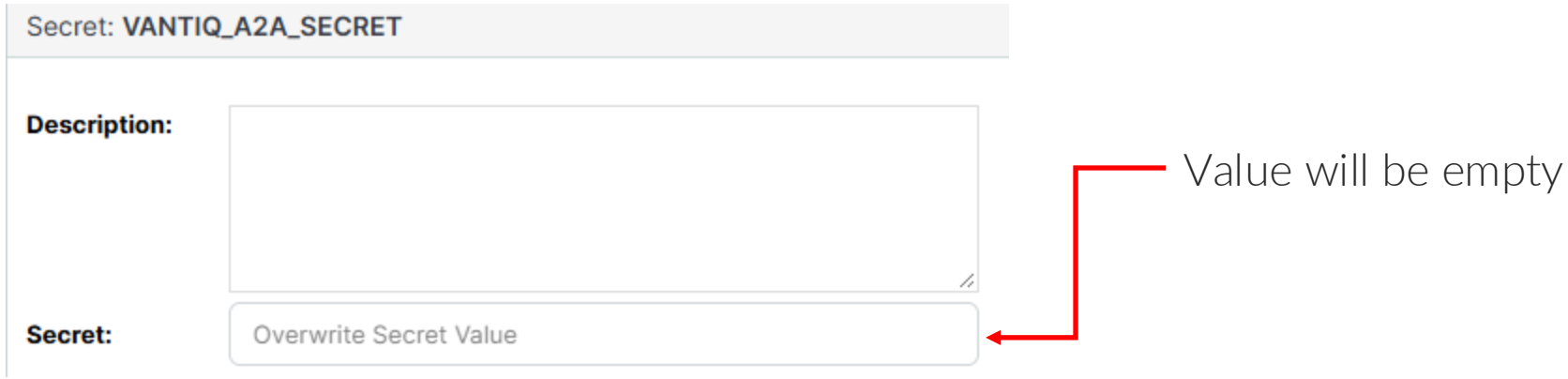
Lamba Function that check the value of the element in the array and returns true/false

Execution of

```
[
  "abc",
  "abcd"
]
```

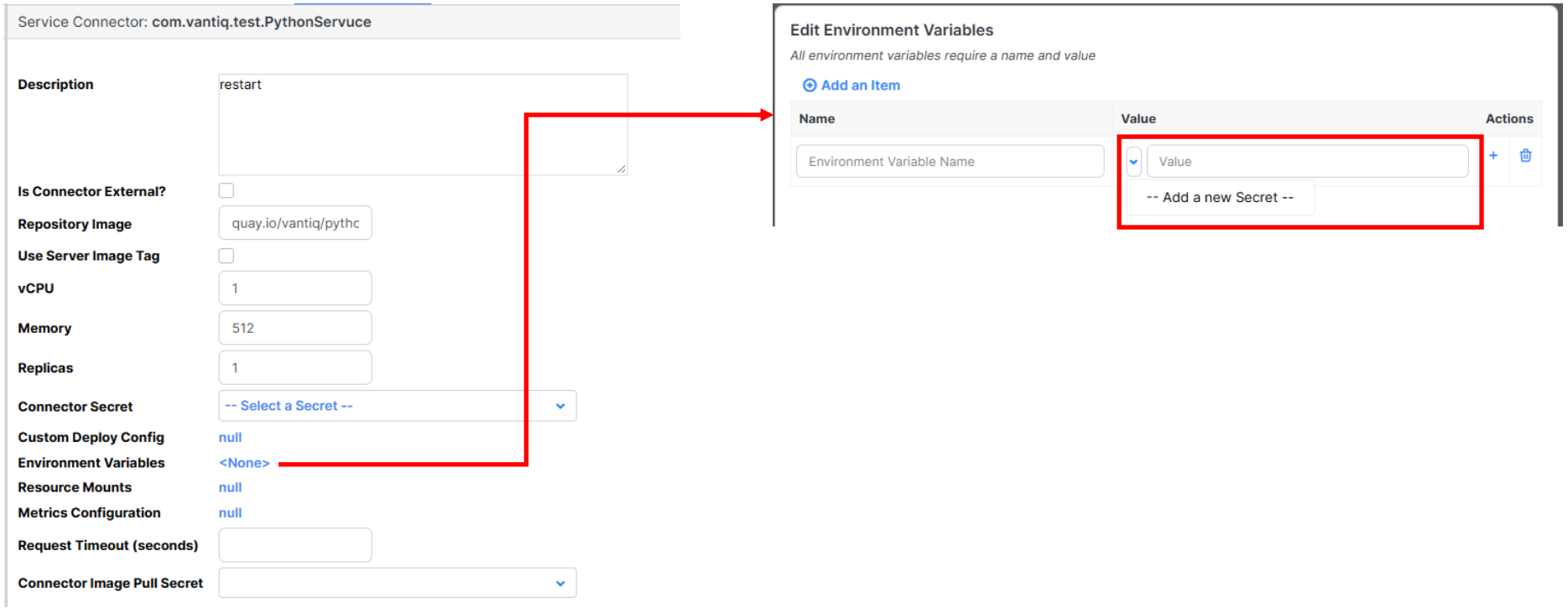
Secret import/export

- Secrets can now be added to a project and exported
  - The secret value will not be exported only the secret resource
- Secrets can now be imported
  - they will not include a value. That will have to be set in the IDE



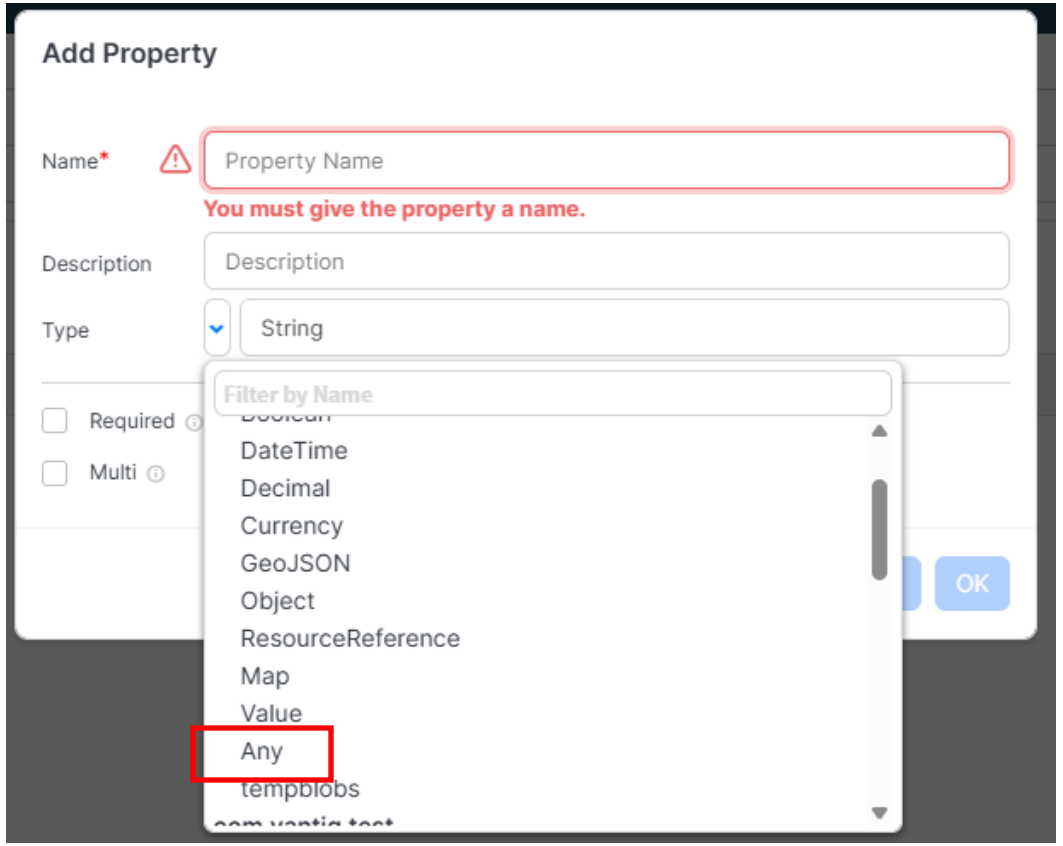
Service Connectors and Secrets

- Secrets can now be passed to service connectors via environment variables



Miscellaneous Enhancements

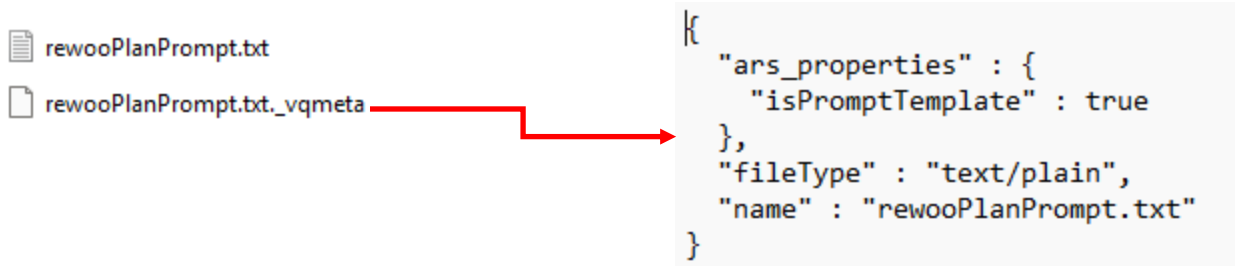
- Schema types now support ‘any’ as a property type



Operational Enhancements

CLI Enhancements

- Dump and Load now support dumping and loading a type from a .dmp file
  - More efficient approach to exporting large types as it is a simple compressed file rather than a series of JSON structures
- Document exports (and IDE exports) now include metadata as a sidecar file



Arm64 Based Edge Image

- A new Arm64 Edge docker image is now published in quay.io along with the amd64
- Arm64 version requirements
  - ARMv8.2 or later hardware supports the full stack (Mongo 7 only)
  - ARMv8.0 based hardware only supports Vantiq Edge server and MongoDB 4.4.18 (not GenAI Flow, Video Assistant or Unstructured)

Updates to Vantiq Infrastructure

- Runtime
  - Java 17, Groovy 5
  - Vert.x 5.1
- Databases
  - Mongo 7 (for edge or private clouds)
  - Qdrant 1.18.2
- The AI Assistant has been removed and its functionality merged into the GenAIFlow services (submitPrompt and answerQuestion)

Deprecated Features

Analytics Sources

- Analytics sources have been removed, as the ML Studio capability they relied on has been deprecated by Microsoft.

Thank You