

VANTIQ 1.44 Update

VANTIQ Engineering & Product Management Enablement

Agenda

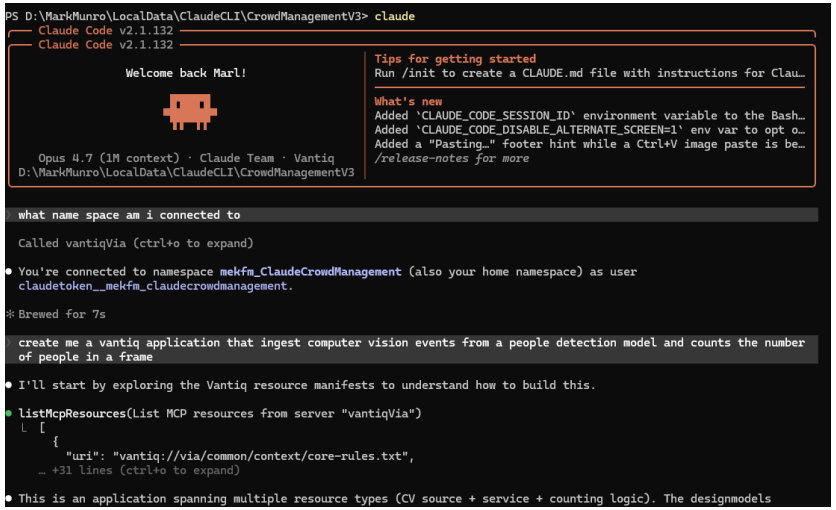
- Overview of New Features and Enhancements
- Claude & AI Development Tool Integration
- Discussions, A2A and GenAI Agents Enhancements & MCP Support
- IDE Enhancements
- Client Builder Enhancements
- Miscellaneous Enhancements
  - IDE
  - Server

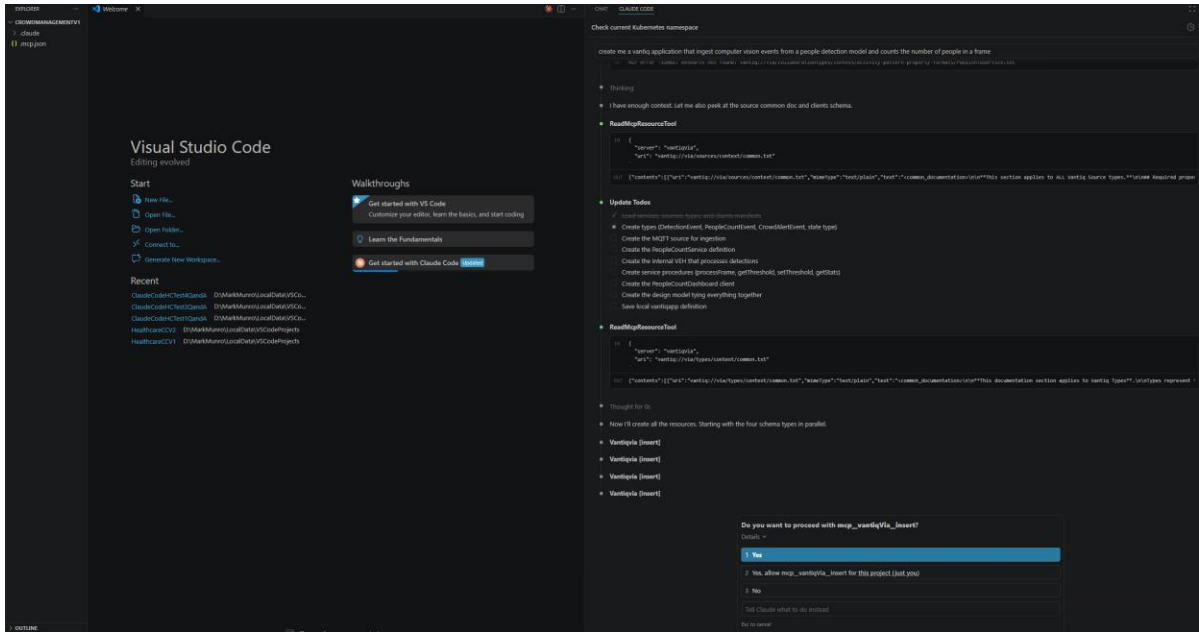
Overview of New Features and Enhancements

Usability Enhancements - Claude and AI Developer Assistant Integration

Increased developer productivity

- VIA integrates with Claude Code and other AI Development tools via a MCP server
  - io.vantiq.via.mcpServer
  - This server provide Vantiq development knowledge and skills to the AI Dev tools
- Development can be driven through CLI's as well as IDE such as VS Code





Usability Enhancements- Multi Agent Support

Increased developer productivity

- Multi Agent Framework Enhancements
  - Ability to send message to agents

    ```
    var task = io.vantiq.ai.A2A.messageSend("Agent", message,
    options)
    ```

  - Ability to stream Agent responses

    ```
    io.vantiq.ai.Agent.executePlanAsSequence(plan, userRequest,
    currentTask.contextId, options)
    ```

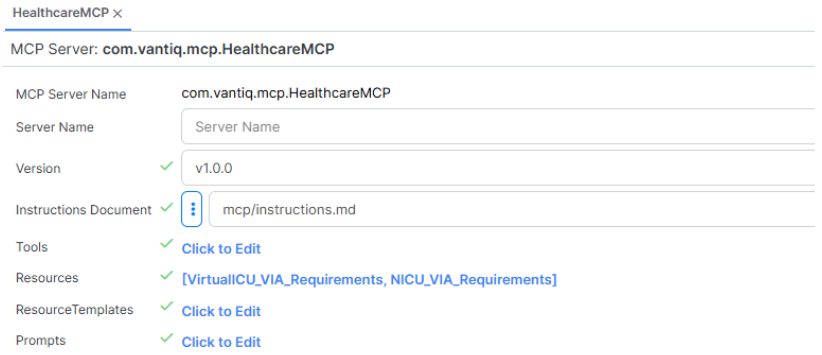
    ```
    io.vantiq.ai.A2A.messageStream("Agent", message, options)
    ```

  - Ability to publish status updates
    - `io.vantiq.ai.Agent.sendTaskStatusUpdate("message")`
- Client and Server support for Discussions

GenAI Enhancements – MCP Integration

Increased integration capabilities

- Model Context Protocol (MCP) support for Vantiq Platform
  - MCPServers will be a new resource within Vantiq
  - Support the Streamable-HTTP protocol
- External systems such as AI Agent/Claude will now be able to access Vantiq through MCP



Claude and AI Developer Tool Integration

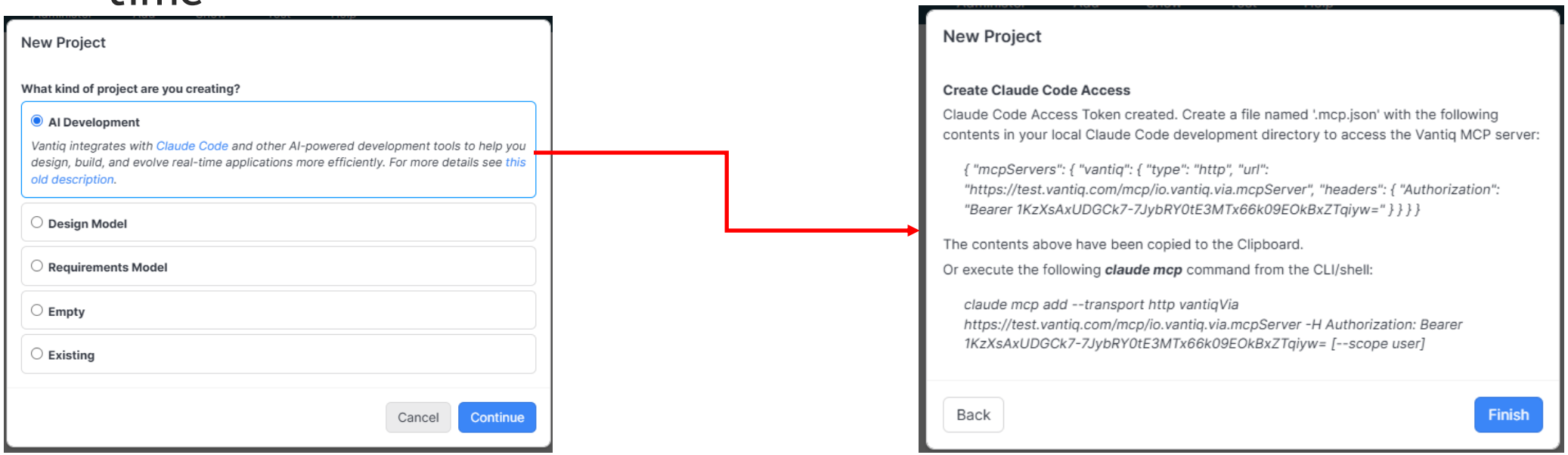
Claude and other AI Dev tools

- Integration of Vantiq and AI development tools such as Claude Code through a dedicated MCP server within the platform
  - Add the MCP server to Claude either via the
    - ‘claude mcp add --transport http vantiqVia https://test.vantiq.com/mcp/io.vantiq.via.mcpServer -H "Authorization: Bearer <accessToken>" [--scope user]’ command
  - Or
    - Adding a ‘.mcp.json’ file to your project directory

```
{
  "mcpServers": {
    "vantiqVia": {
      "type": "http",
      "url": "https://test.vantiq.com/mcp/io.vantiq.via.mcpServer",
      "headers": {
        "Authorization": "Bearer a[illegible]g8A="
      }}
  }
}
```

New Project Wizard - AI Dev Tools

- Create a project for AI Dev Tools – Generate a token and show mcp commands
  - New user will also get an announcement when they login for the first time



The ‘io.vantiq.via.mcpServer ’MCP server

- Provides resources and tools to allow Claude and other tools to generate Vantiq Applications
  - Resources describes the various capabilities
  - Tools such as select, insert allow the AI dev tool to access Vantiq resources
  - The resources are saved to you namespace as well as the local file system
  - It understand namespaces and can switch namespaces if using a org level access token
    - You can also ask Claude to always use a specific namespace for a project – it will save that fact away in its memory

Claude and namespaces

● You're connected to the **mmSandbox_root** namespace (also your home namespace), as user **claudetoken__mmsandbox_root**.

✻ Sautéed for 8s

› what namespaces do I have access to

Called vantiqVia (ctrl+o to expand)

● **readMcpResource**(Read resource "vantiq://via/common/context/namespaces.txt" from server "vantiqVia")
└ {
"contents": [
{
… +69 lines (ctrl+o to expand)

Called vantiqVia 10 times (ctrl+o to expand)

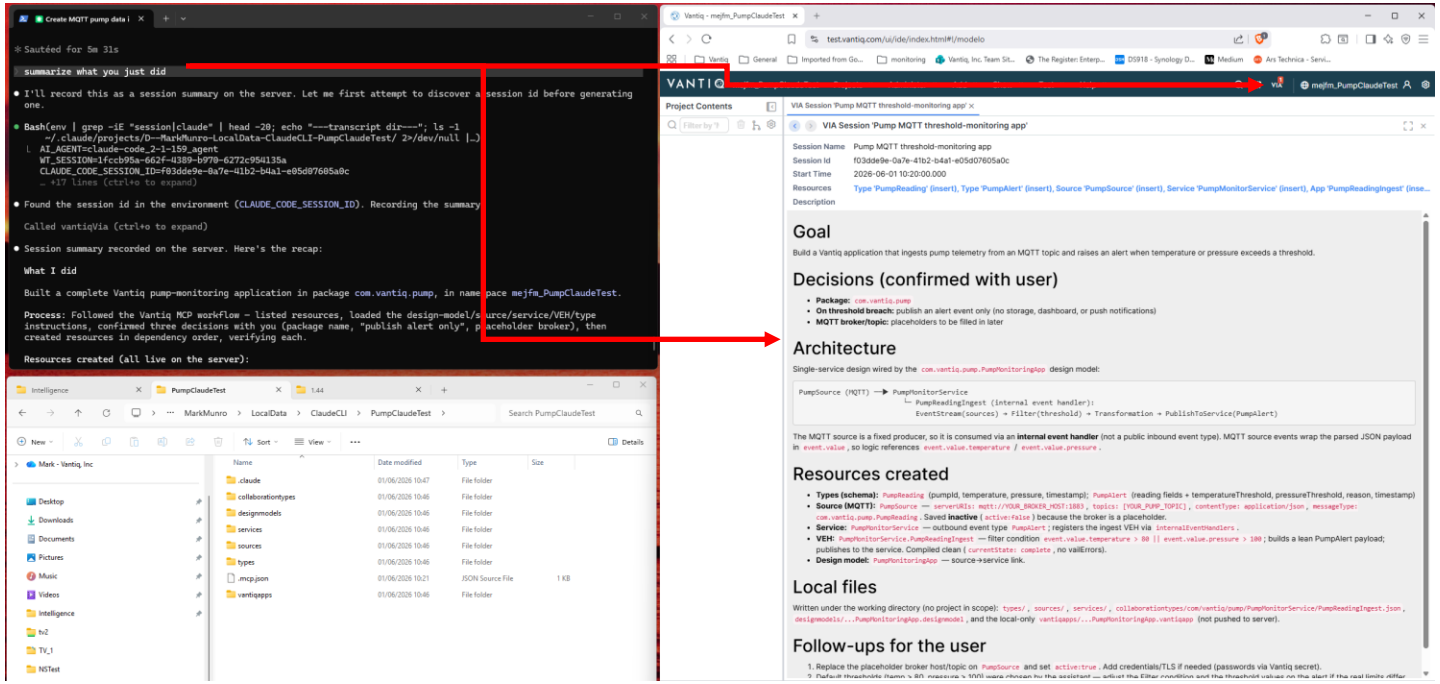
● You have access to 9 **namespaces** in the **mmSandbox** organization (you're an org admin, so you can act in any of them v
asOrgAdmin: true if needed):

Namespace	Role
mmSandbox_root	Organization root (your home)
mejfmSB_Intro	sub-namespace
mejfmSB_Deploy	sub-namespace
mejfm_DIYCat	sub-namespace
mejfm_DIYRigadoDev	sub-namespace
mejfm_HPE_RigadoDev	sub-namespace
mejgm_GSTesting	sub-namespace
mejgm_140Testinh	sub-namespace
mejfm_AIAgentTests	sub-namespace

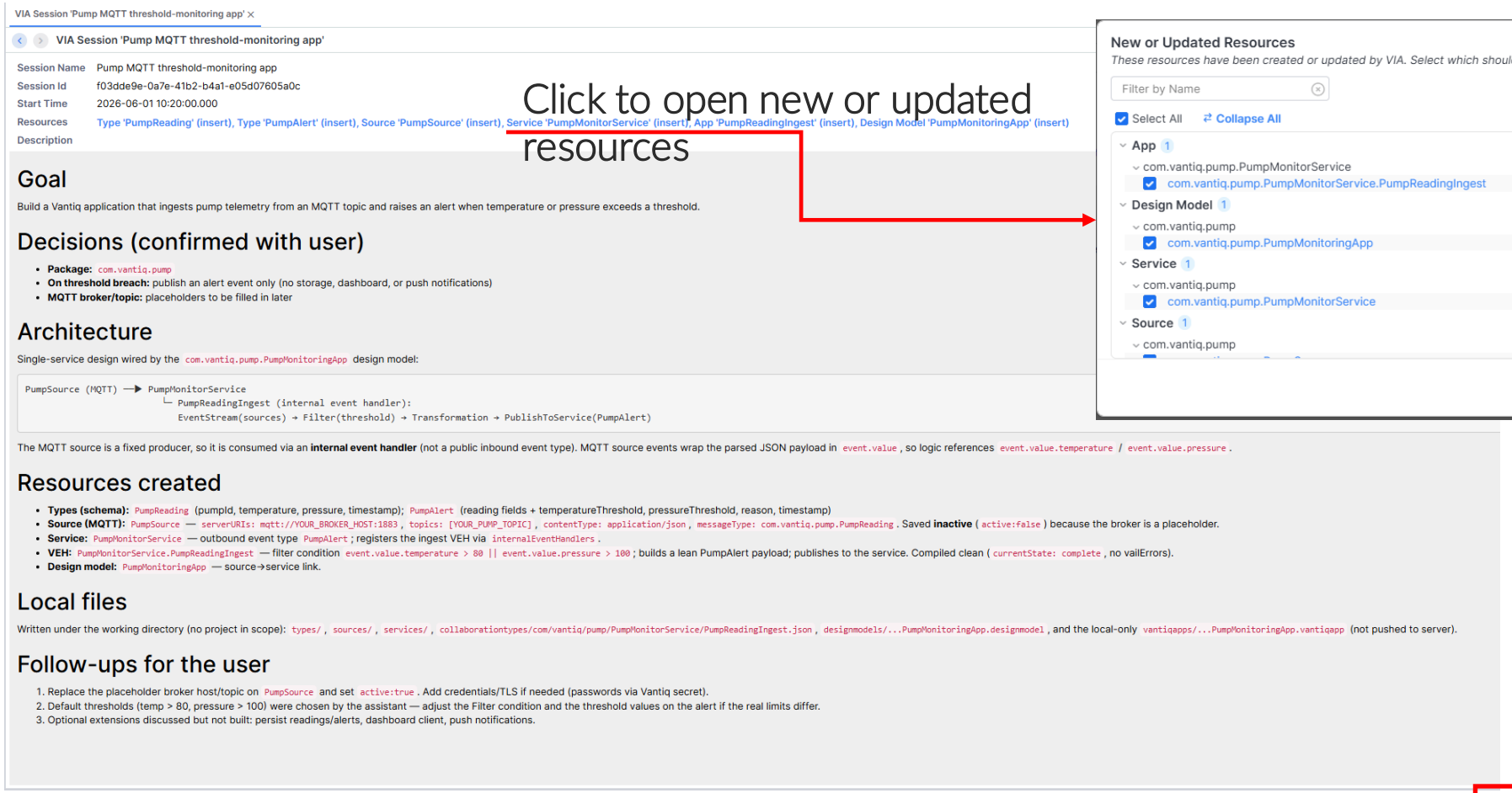
All belong to organization **mmSandbox**. To target one in a tool call, pass **namespace: "<name>"** (and **options: {**
asOrgAdmin: true } for any namespace where you don't hold an explicit authorization).

Claude and Vantiq IDE Integration

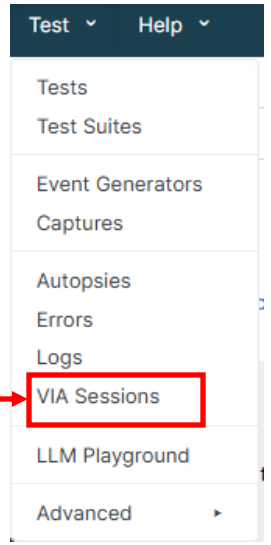
- Ask Claude to summarize that it just did will result in a recoding about your current Claude session being recording in your namespace and a new UI is available to visualize the information
  - ‘summarize what you did’
  - This will send summary to NS



Claude and Vantiq IDE Integration cont.



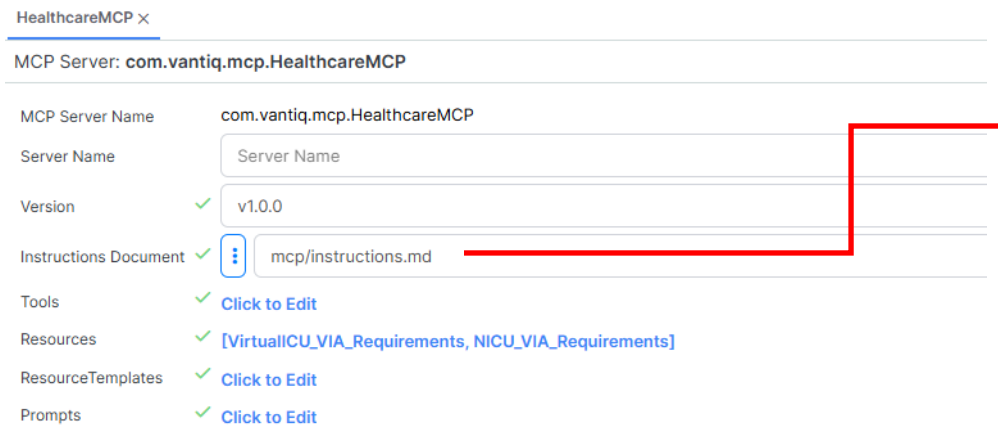
Click to open new or updated resources

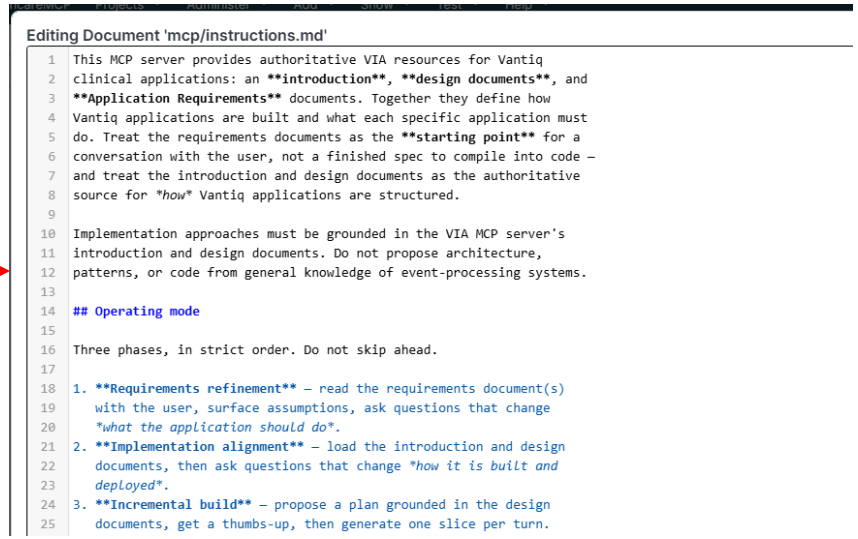


Test->VIA Sessions to see previous sessions

Enhance the AI Development experience

- Add additional MCP servers to allow you to fine tune how and what Claude or other AI development tools create for you
- MCP servers are now a standard resource in the platform. See section in GenAI Enhancements





Claude and the Healthcare ICU MCP server



GenAI Enhancements

Multi Agent vs Single Agent system

- 1.43 primarily supported single agent systems
- 1.43 started to introduce the multi-agent framework based on the Google A2A protocol – ***marked as a preview feature***
- 1.44 expands and completes the multi-agent framework and introduces several new features and concepts
  - Discussions
  - Enhancement to the framework
  - Discussion Widget

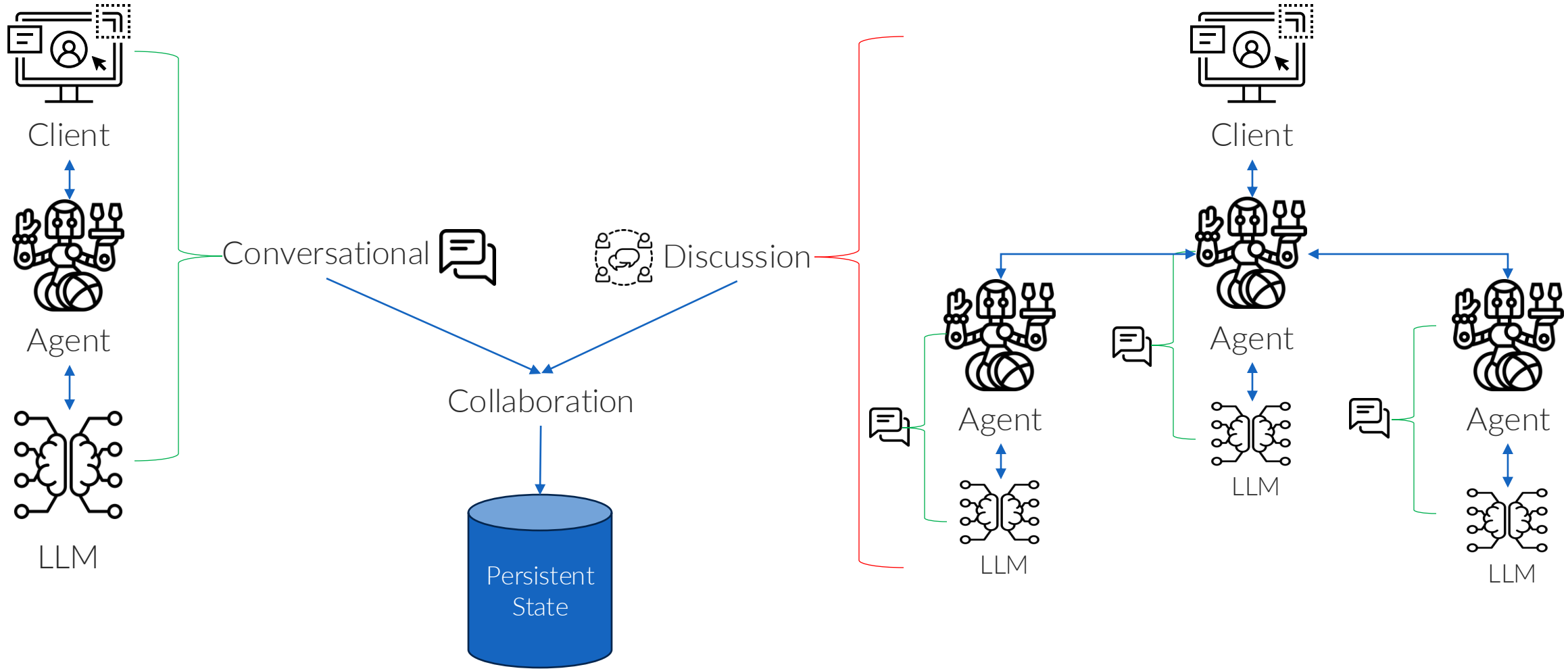
First – What is a Conversation

- A conversation manages the state of an interaction between an LLM and a client of the LLM (user or agent)
  - If you have more than one LLM and or client then you have multiple conversations
  - If you have multiple interactions with separate results then each is managed by multiple conversations
- In a single Agent environment (1.43) each client/agent->llm interaction is discrete and produces a single conclusion. In which case conversation manages this state.
- If the conversation needs to be long lived, they are persisted and managed through a Collaboration

Multi Agent Systems

- 1.44 Expands & rounds out the multi-agent framework
- When you have clients and multiple agents talking to multiple llms you can no longer manage the state solely through conversations
- This is what Discussions represent – a way of managing the interaction and state between multiple agents, multiple llms and clients (and as a consequence multiple Conversations)
  - Each Agent manages its own state (through Conversations and potentially collaborations)
  - The overall state is managed through a Discussion
  - Discussions have a ‘relatedResource’ property that can hold references to other metadata associated with the Discussion

Multi Agent vs Single Agent system

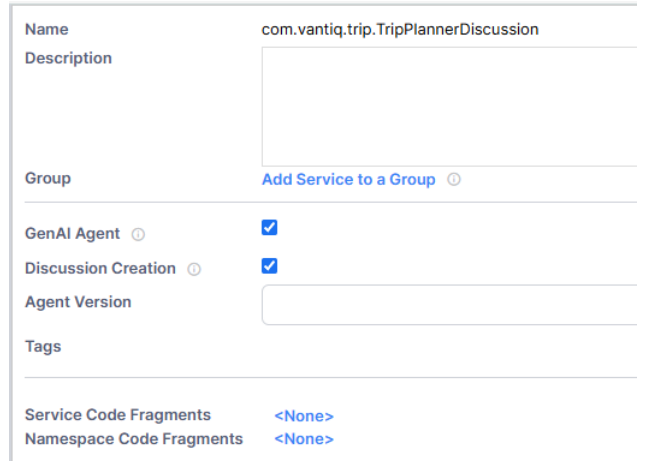


Discussions

- A Discussion not only records the messages exchanged between clients, agents, and llms, it also records the artifacts
  - Conversations on the other hand just record messages
- A Discussion is owned by a single Agent
  - The Managing/supervisor Agent updates the Discussion state based on responses from Agents
  - Individual Agent can manage and maintain their own state
    - They could also create their own Discussions

AI Agents and Discussions

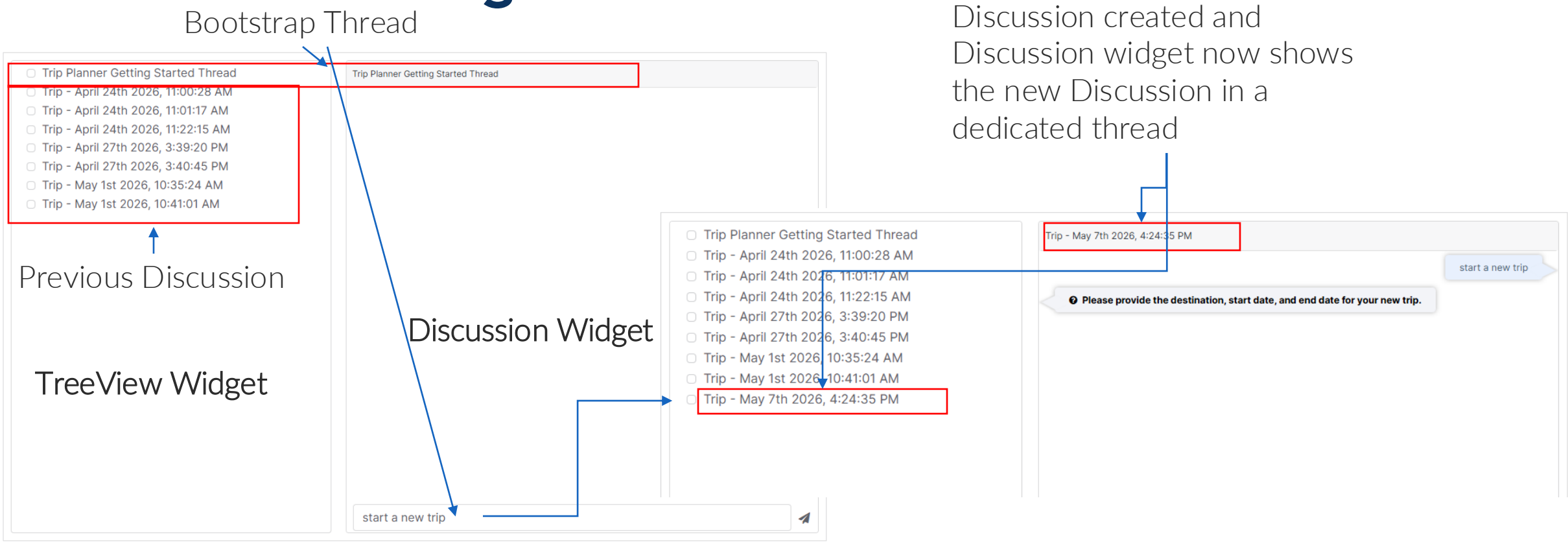
- An AI Agent has an additional property ‘Discussion Creation’ which is true by default
  - A client must use the A2A messageSend/Stream to interact with the Agent
  - The client must define an A2A extension in the messageSend/Stream to cause the Agent to request the Agent creates a Discussion
  - The Discussion will automatically have a collaboration associated with it to manage its state
  - The Discussion Widget handles this automatically
    - *see next slide*



AI Agents/Discussion and the Discussion Widget

- The Discussion widget can only be associated with an AI Agent
- If the Agent has ‘Discussion Creation’ as true (the default) then a Discussion and associated collaboration will be created automatically and returned to the Discussion Widget
- A Discussion is created when the user sends an initial message from the Discussion widget to the server (via messageSend)
  - This is a ‘Discussion’less interaction and is used to bootstrap a Discussion
  - Once the bootstrap message is sent and the Discussion is created the Discussion Widget switches so displaying the ‘Real’ Discussion
  - The Discussion Widget is able to be switched between Discussions and will load the state of the Discussion from the server

Discussion Widget



Demo - New Trip Planner

Additional Framework Built-in Functions

- io.vantiq.ai.Agent…
  - `sendTaskArtifactUpdate`
  - `getDispatchPlan`
  - `getAgentsMd`
  - `messageSend`
  - `messageStream`
  - `executePlanAsSequence`
- io.vantiq.ai.A2A…
  - `discussionGet`
  - `Message.forValue / build`
  - `Artifact.forValue / build`
  - `Part.forValue / buildText / buildData / buildFileFromBytes / buildFileFromUri`

Dynamic Planner

- Added a new nodeType in the planExecute structure ‘planner’
- This allows a plan to dynamically generate a plan when executing that step
- The node must define a tool (procedure) that will return a plan
- The generated plan will be dynamically injected into the parent plan

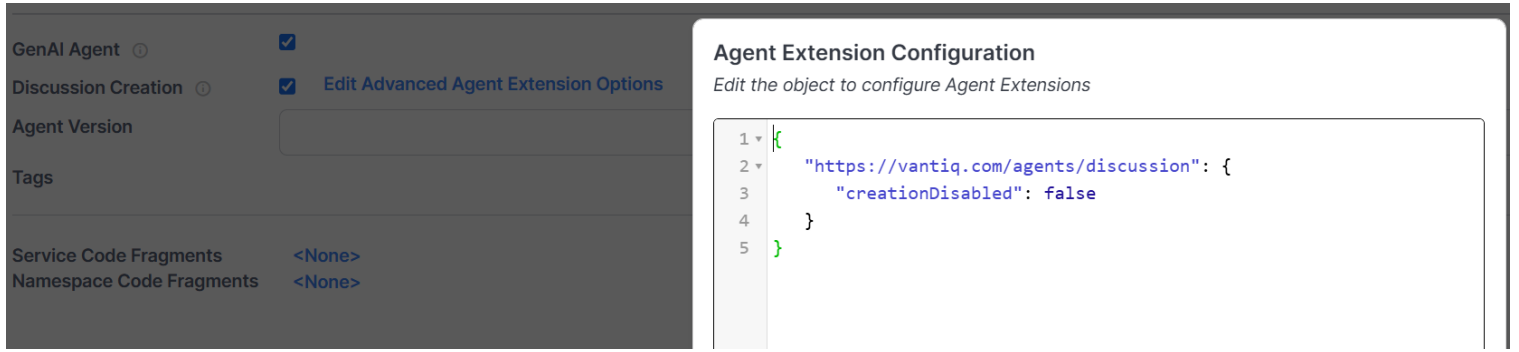
Better image model support within GenAI Flows

- Support for passing documents and TempBlob references into a GenAI Flow
  - These can then be passed to model that support images or documents
- Support for OpenAI's multi-turn editing feature for images
  - LLM configurations can declare '`extra_request_params`' which is can be used to pass the OpenAI image '`previous_response_id`' that OpenAI multi-turn editing option requires
- Updated LangChain to V1.0 and now make use of their model agnostic message structures

Custom LLM and Dispatch mechanism

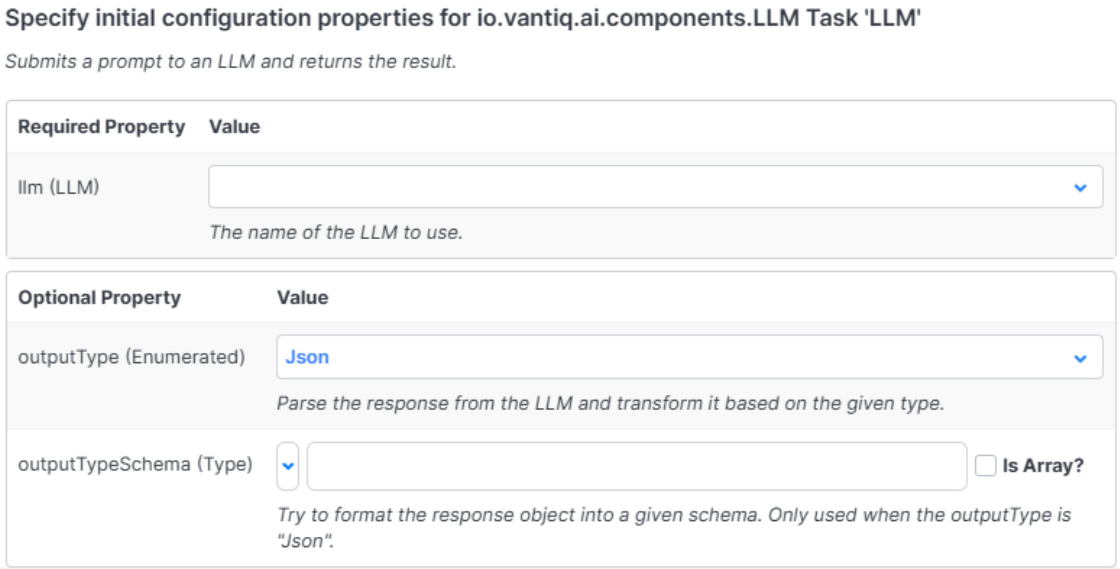
- At a resource level it is possible to customize the LLM and Dispatch mechanism that the multi-agent framework uses

```
{
"https://vantiq.com/agents/requestDispatch" : {
      "algorithm" : "tools"
    },
"https://vantiq.com/agents/requestDispatch" : {
      " llm" : “llm name"
    },
"https://vantiq.com/agents/requestDispatch" : {
      "prompts" : {
        "planner" : “/plan/DispatchPlanPrompt.txt"
      }
}
```



Miscellaneous

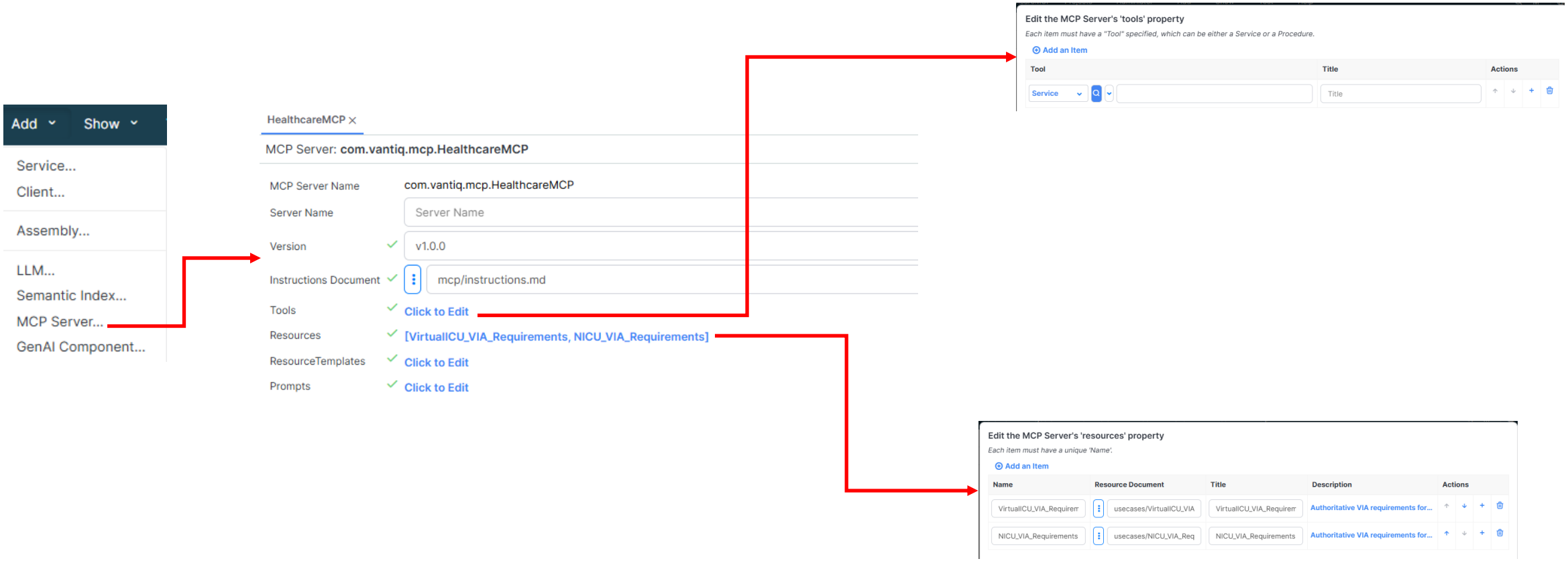
- Support for nested types when specify an output type of Json and defining an outputTypeScema
- ‘Is Array’ means the LLM will return an array of outputTypeScema



MCPServer Resource Details

- MCPServers will support the following MCP capabilities
  - ***instructions*** - Reference to a Document containing the MCP server instructions
  - ***resources*** - a list of the MCP resources and resource templates provided by the MCP server – Document references
  - ***prompts*** - a list of the MCP prompts provided by the MCP server – Document References
  - ***tools*** - a list of the MCP tools provided by the MCP server – Service or Procedure references
- MCPServers can be per-namespace or organization wide, if they are org based then they can access all namespaces within an organization
- Your MCP Servers will be accessible via the *http(s)://servername/mcp/<mcpServerName*

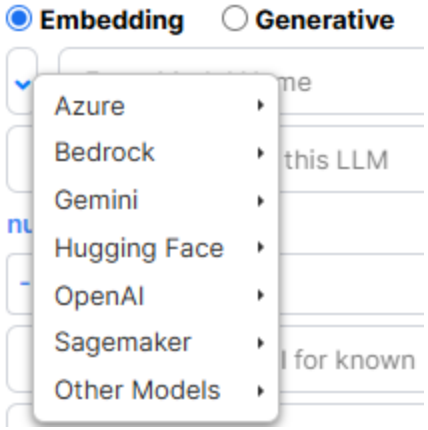
Adding an MCP Server

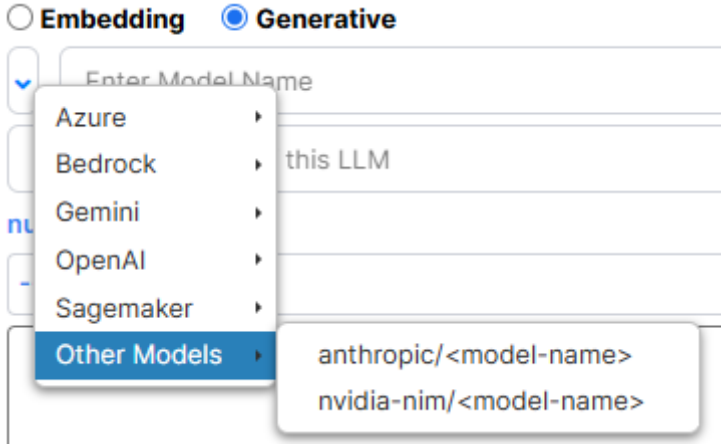


IDE Usability Enhancements

Change to LLM list

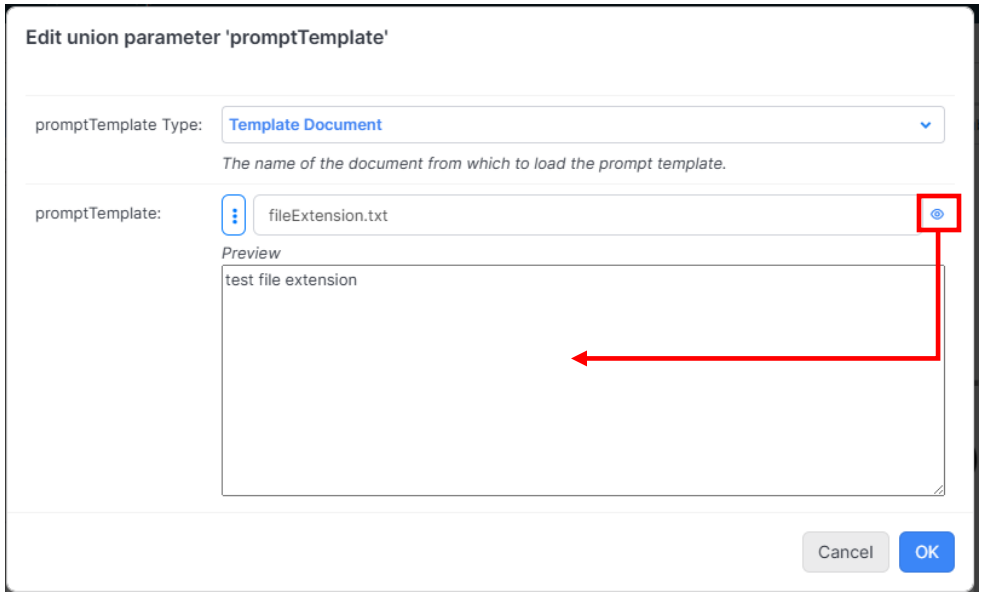
- Reorganised the LLM drop list and added additional options





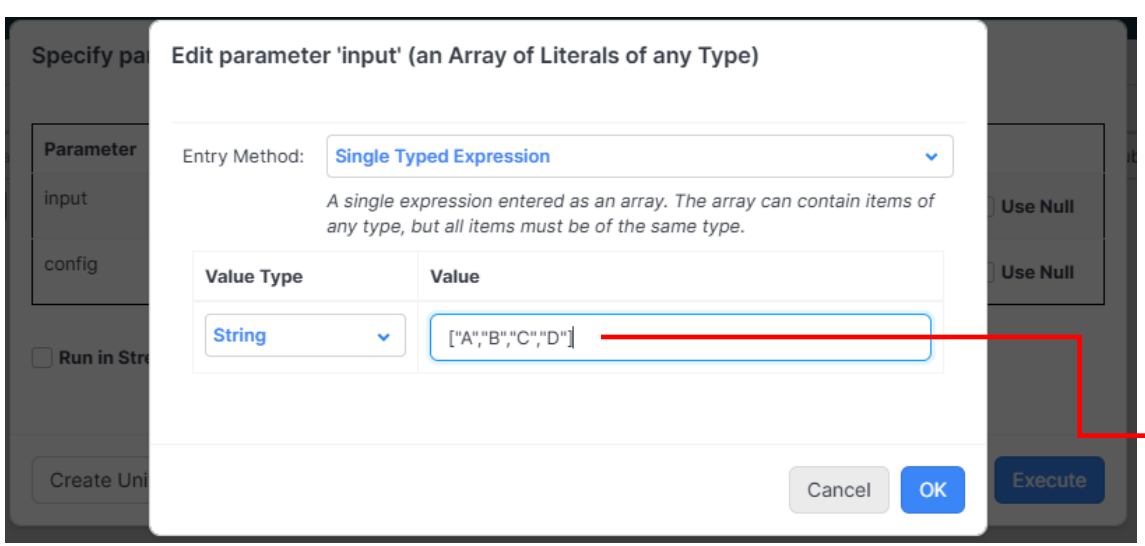
GenAI Builder Enhancements

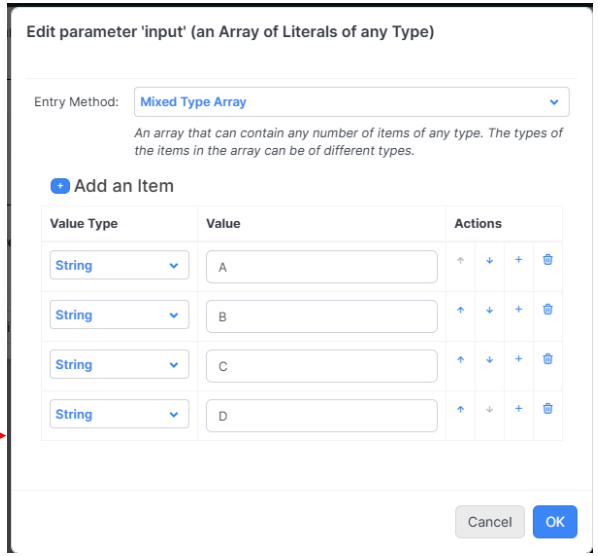
- ‘Eyeball’ added to the PromptFromTemplate/Message
  - Allows you to preview the prompt file



GenAI Builder Enhancements

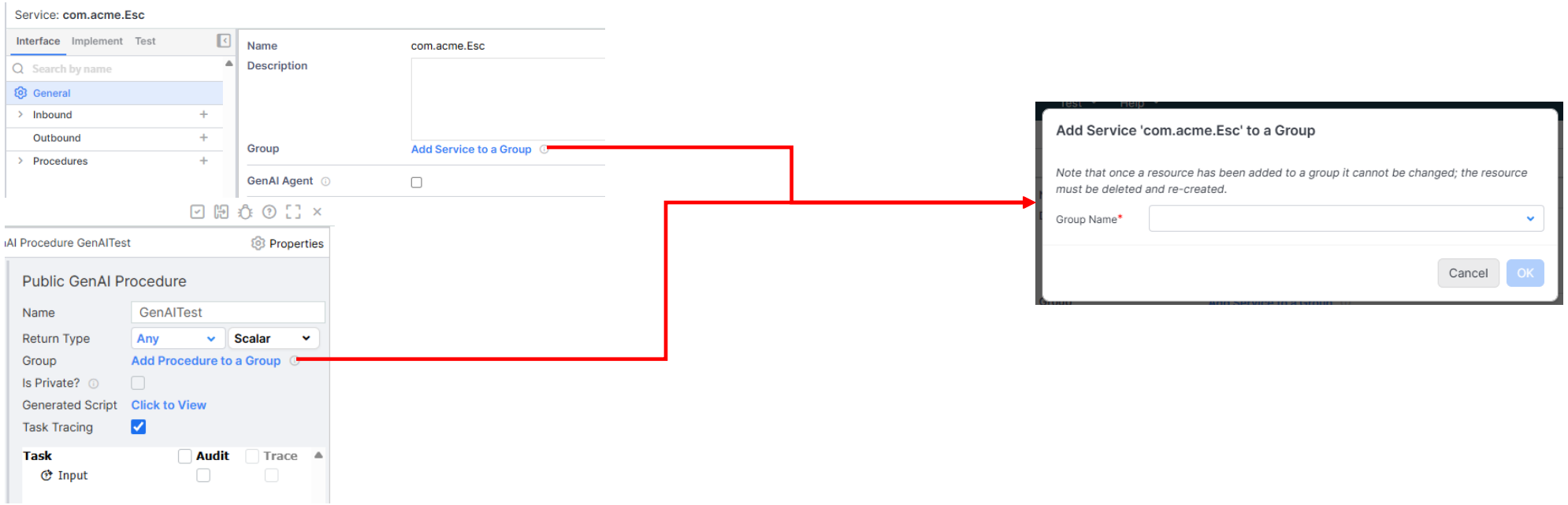
- Calling GenAI Procedure with Array parameter previously force you to manually entry each array element. Now can paste in a string that will be converted to an array input





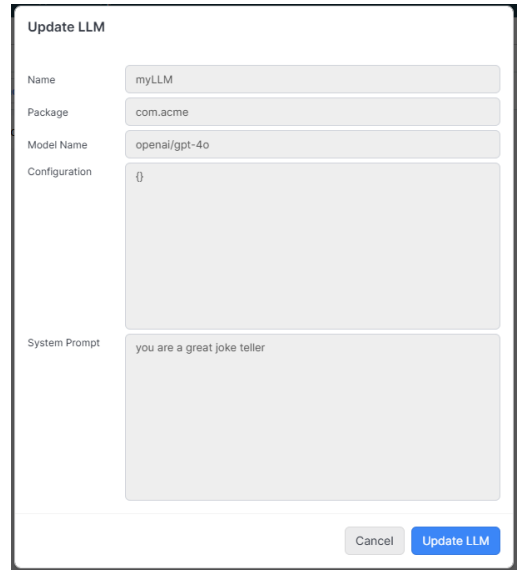
Group Support for Service & GenAI Procedures

- Service and GenAI Procedure can now be Group enabled (features was back ported to 1.43)
  - Procedures already had the ars_group = '<groupName>' modifier

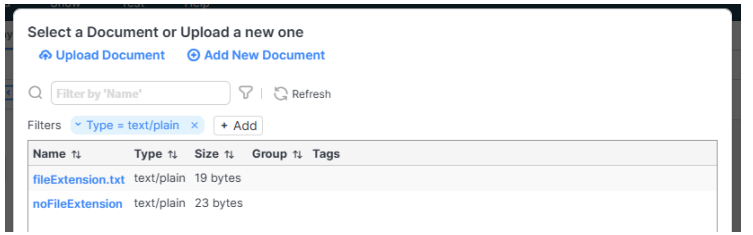


LLM Playground Enhancements

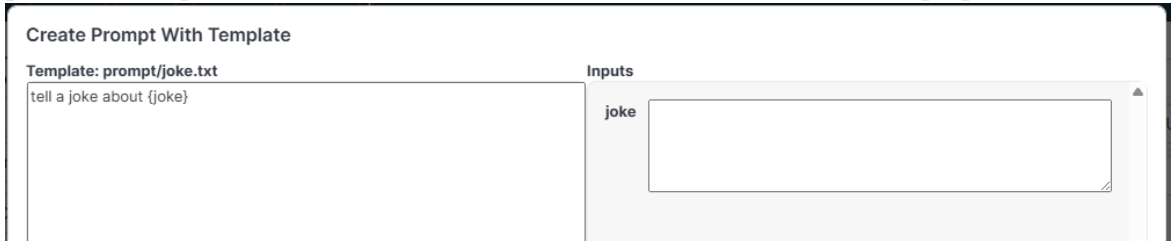
- Update LLM and System Prompt – Click Update LLM



- Load Prompt – Adds text filter automatically



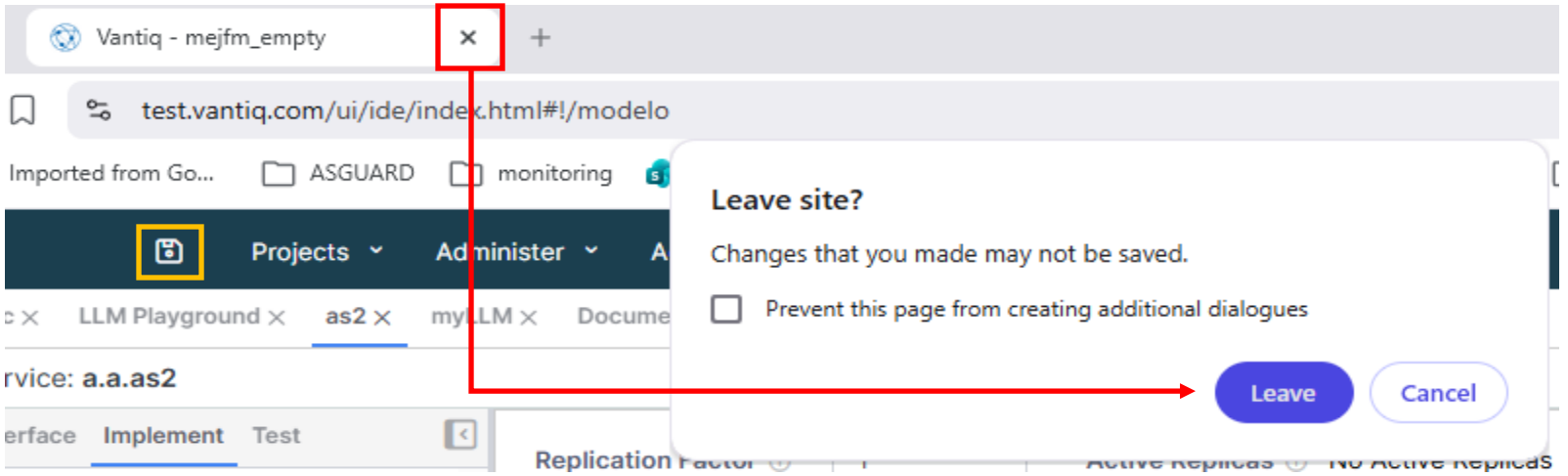
- {} as well as ${} format for prompt substitution now supported



Unsaved Changes

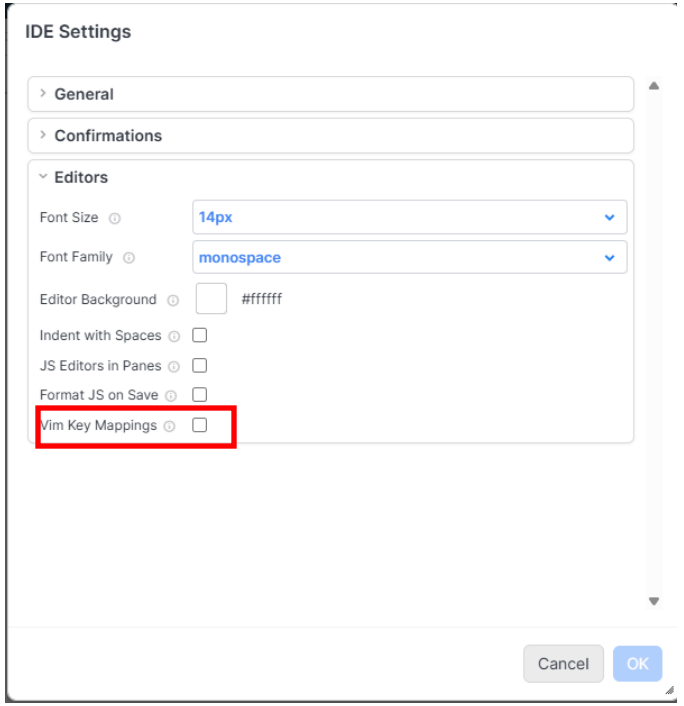
- New IDE Setting – Unsaved Changes
  - NS Change or Logout displays a confirmation dialog
  - Close(X) Tab results in a browser dialog to confirm or discard close





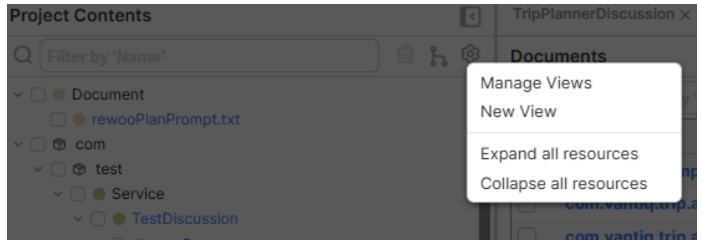
Code Mirror Editor Vim Key mapping

- Code Mirror based editors – VAIL, Documents etc. now support Vim keyboard mapping

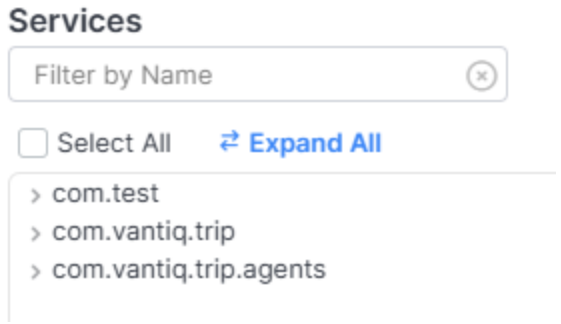


Miscellaneous

- When editing CSS document auto correct is off
- Document pane now supports sorting by name and date. Also supports double click to open a document
- Collapse & Expand All in Project Tree

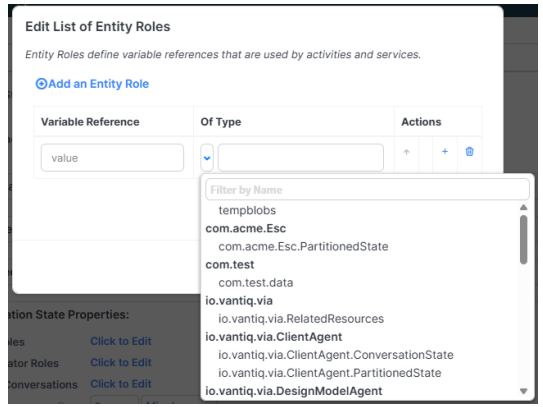


- Collapse & Expand All in Add Resource popup

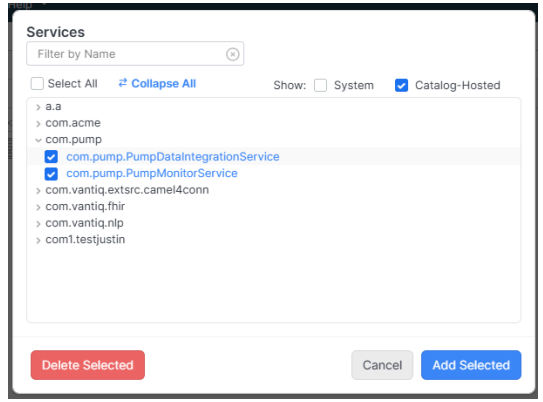


Miscellaneous cont.

- Entity Roles Droplist now shows the new filterable/hierarchical list of types



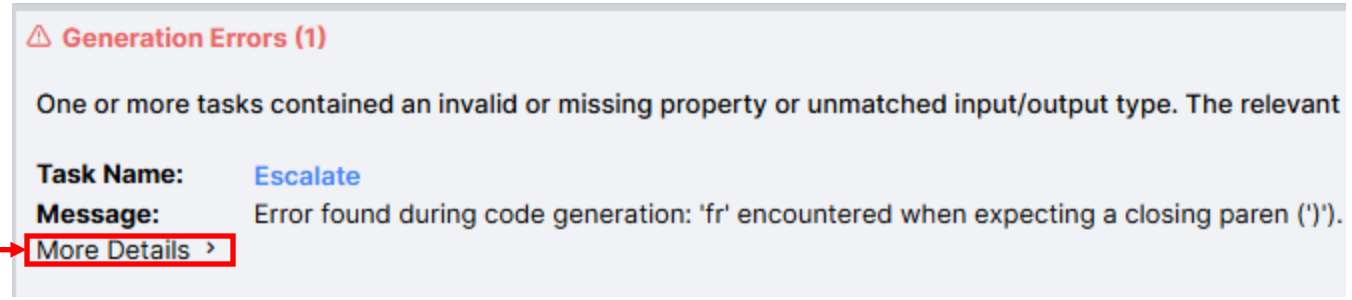
- Design Model – Add Service/Client allows multi selection



- Project Description Editor popup moveable

Miscellaneous cont.

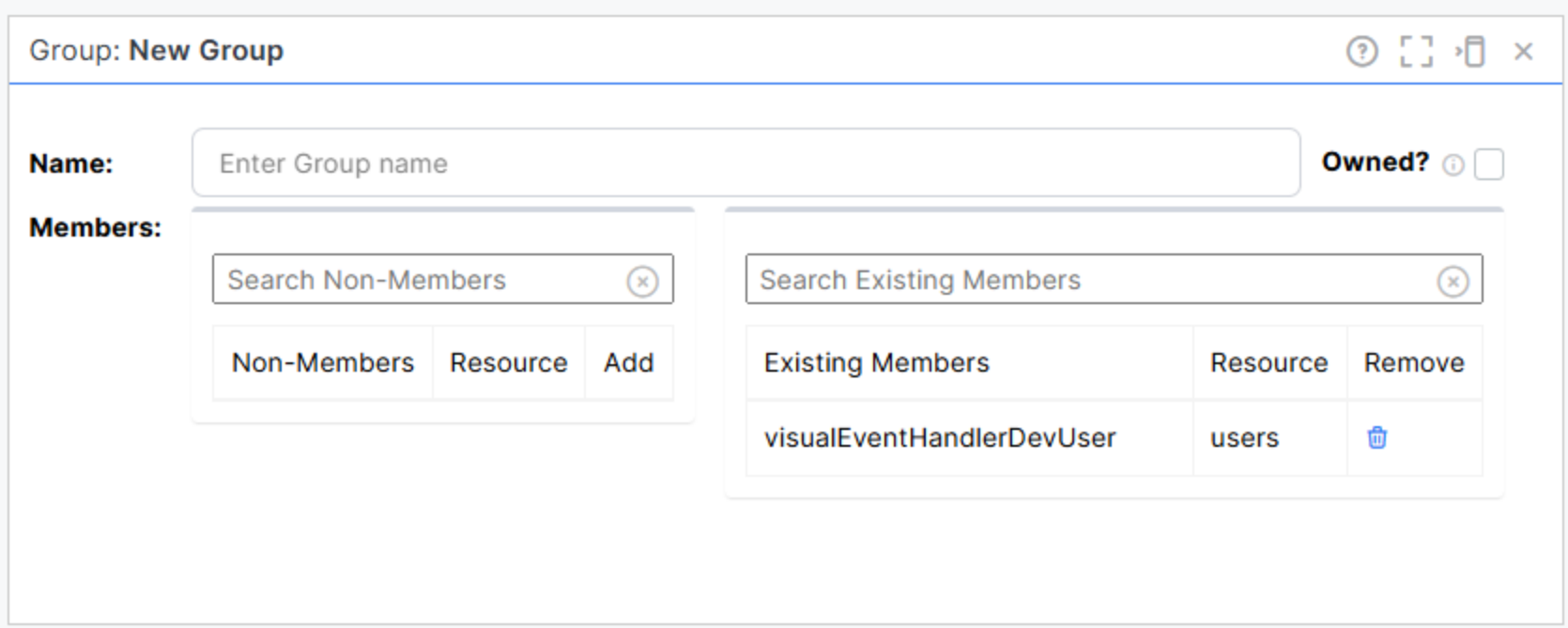
- More detailed compilation/generate errors in Visual Event Handler



- When Deleting a resource that is in another project it will no longer be deleted just removed from the current project

Owned/Un-owned Groups

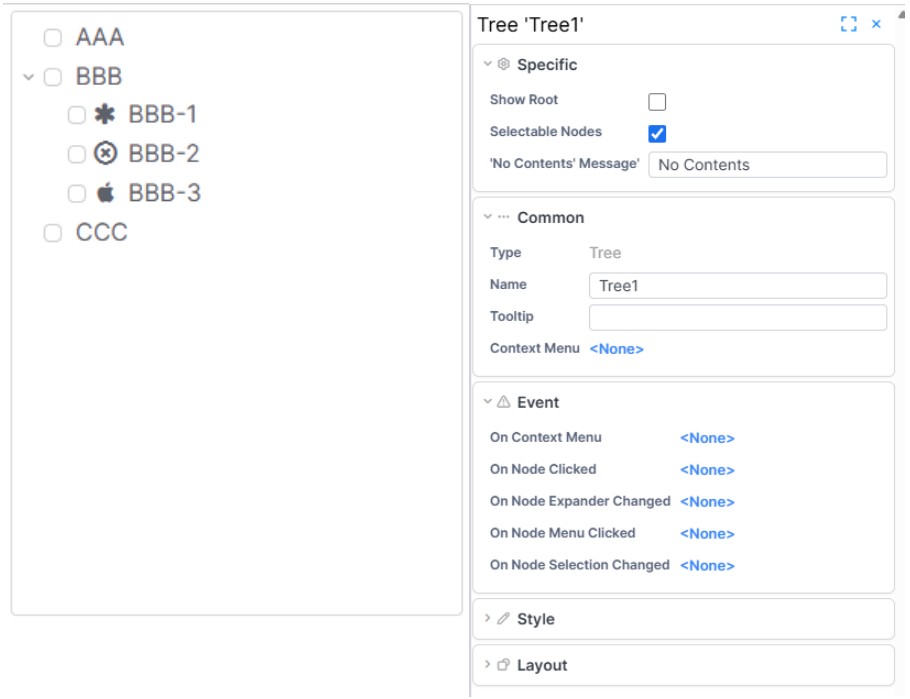
- IDE now support the creation of un-owed groups – defaults to unowned



Client Builder Enhancements

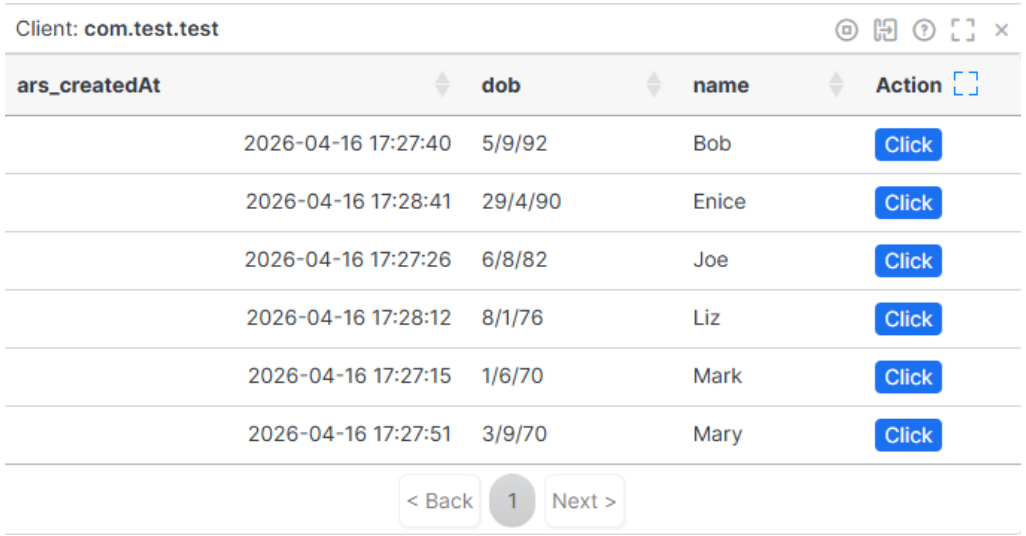
New Widgets

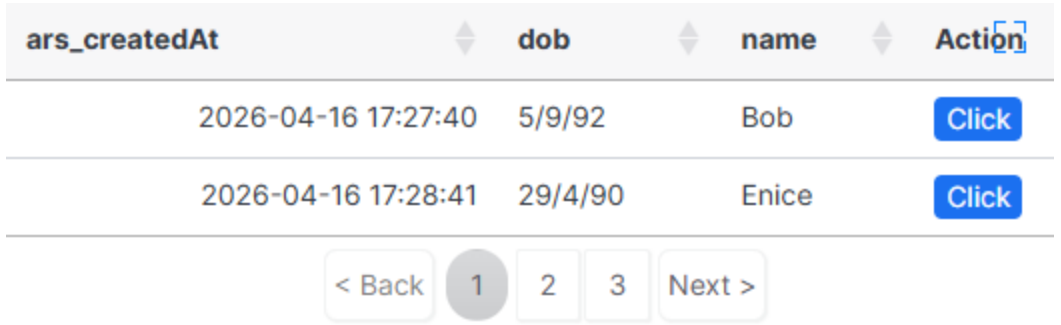
- Discussion Widget – discussed above
- Tree Widget – based on the Fancy Tree control used in the IDE
  - Display hierarchical data



Datatable Changes

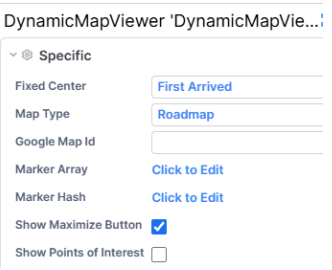
- Datatable widget now supports Size to Parent on height



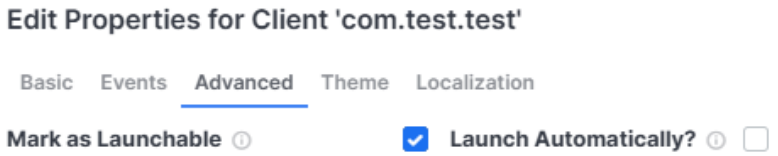


Miscellaneous

- Update JSHint to 2.13.6 & Update Default ES Version ES11
- Google Maps widget now supports – Hide/show POI



- Clients can be marked to auto launch

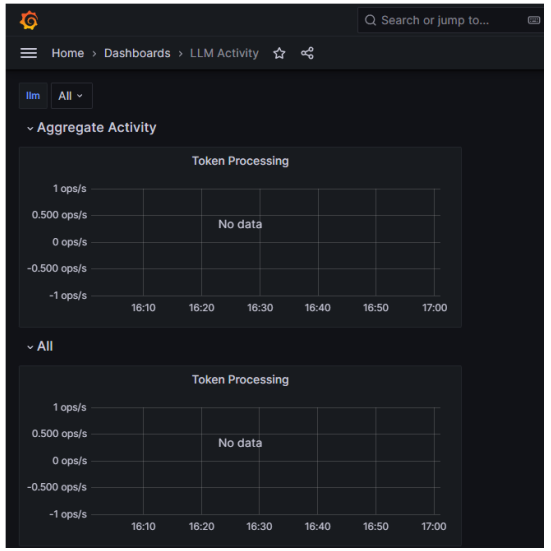


- Position pop-up in ClientBuilder – client.popupPage() new param
  - pagePosition:PagePosition - Center, NW, N, NE, W, E, SW, S, or SE
  - xOffset:number – pixel offset
  - yOffset:number – pixel offset

Miscellaneous Enhancements

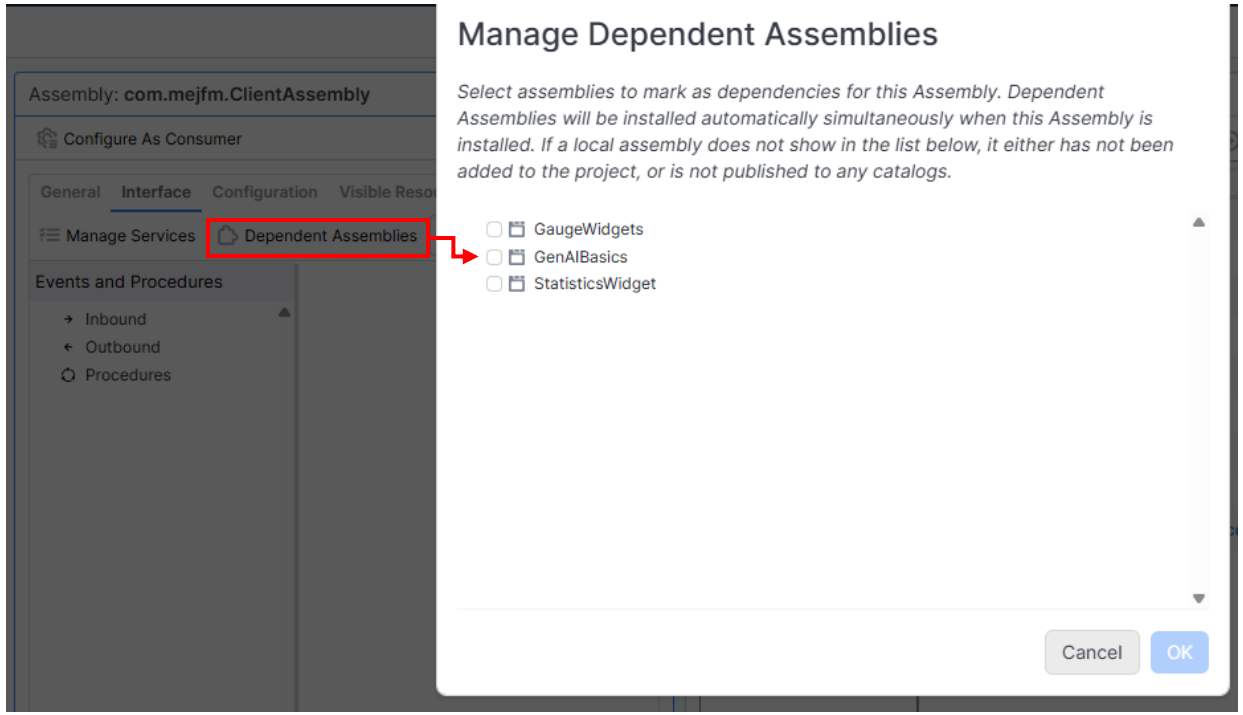
Server Enhancements

- Provide checksum function for TempBlobs
- New built-in merge – that will merge multi sequences
- Add @secret support to a Remote source payload body
- New generated collaboration procedure xxxGetById
- Grafana updates
  - LLM



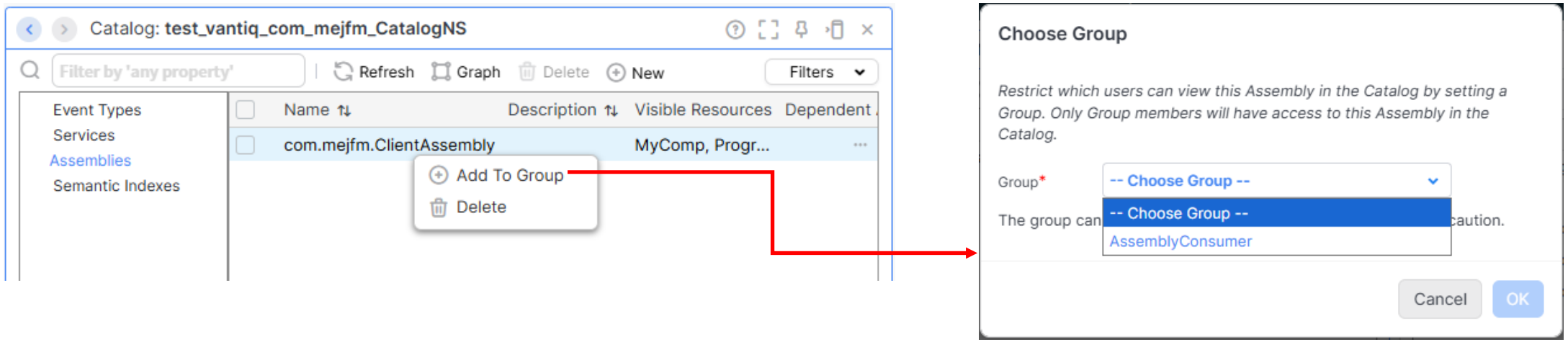
Assembly Enhancements

- Dependent Assemblies
  - An Assembly can now have multi dependency
  - When the Assembly is installed all of the dependent assemblies will also be installed



Catalog Enhancements

- Assemblies and Services now support ‘Groups’ to be able to restrict who can access a given assembly or service registered in the catalog



Deprecated Features

GuardrailsAI pattern

- The GuardrailsAI pattern in the GenAI Builder will be removed in 1.44

Thank You