



# QPR ProcessAnalyzer 2026.4

## New features

# MCP: Added input parameters, structured output and improved OAuth support



- MCP tools have input parameters to define what kind of data the agent needs to pass to the tool
  - Input parameters are defined as JSON schema
  - Tip: Use LLM to generate input parameters schema
- Structured output can be defined for MCP tools, to inform agent what kind of data to expect from the tool (defined as JSON schema)
  - Agents are good at interpreting free text tool output, so structured output is not required in most cases
- Metadata can be defined for tools: destructiveHint, idempotentHint, openWorldHint, readOnlyHint
- OAuth has been improved
  - Added support for OAuth client credentials flow which maps to specific QPR ProcessAnalyzer user
  - Refresh tokens allows agents to use MCP long time without reauthentication (refresh tokens remain valid event if server restarts)
  - MCP interface is rate limited with configurable threshold (default 60 calls per minute per user)
- Limited MCP tool names to first 50 characters of script name (required by Claude)

### Script Properties: GetModelDetails

GENERAL **INPUT** OUTPUT MCP DESCRIPTION

```
{
  "type": "object",
  "additionalProperties": false,
  "properties": {
    "modelId": {
      "type": "integer",
      "description": "Model identifier"
    }
  }
}
```

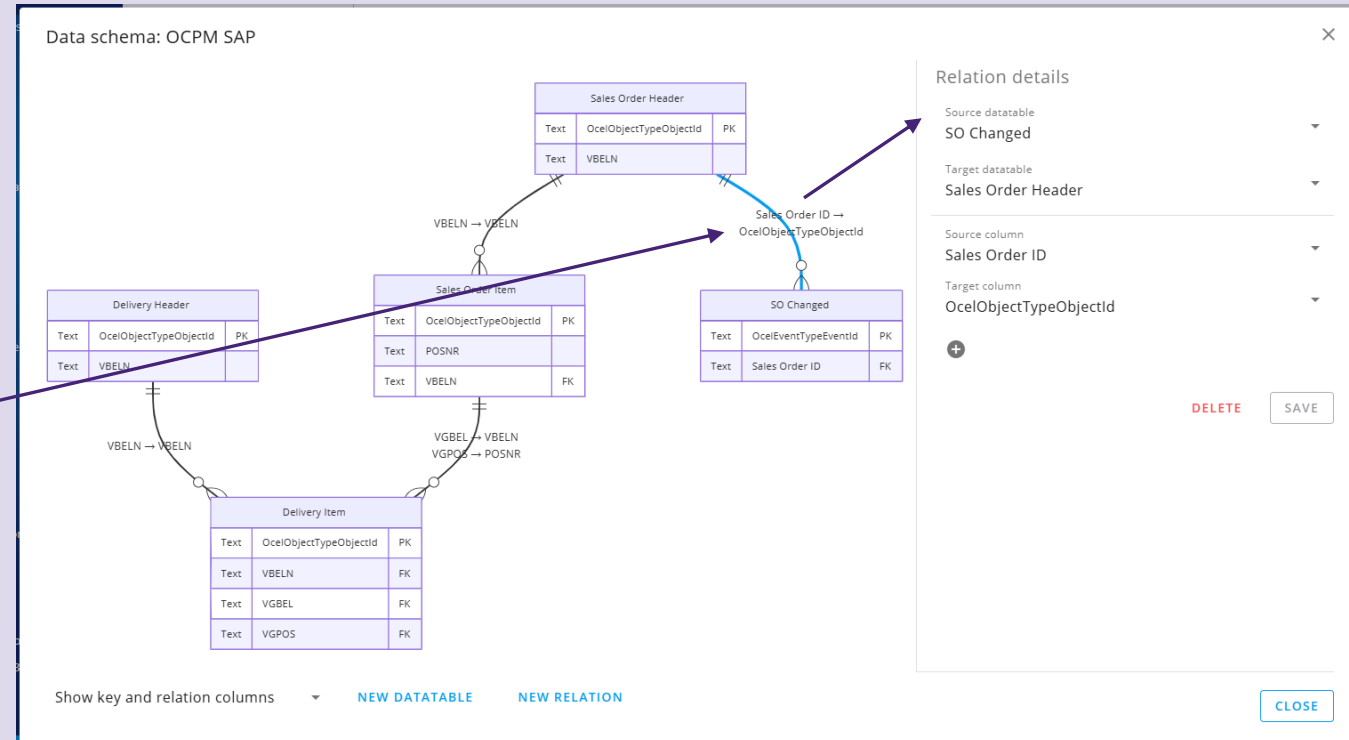
CLOSE

[wiki.onqpr.com/pa/index.php/QPR\\_ProcessAnalyzer\\_as\\_MCP\\_Server](https://wiki.onqpr.com/pa/index.php/QPR_ProcessAnalyzer_as_MCP_Server)

[wiki.onqpr.com/pa/index.php/Create\\_MCP\\_Tools](https://wiki.onqpr.com/pa/index.php/Create_MCP_Tools)

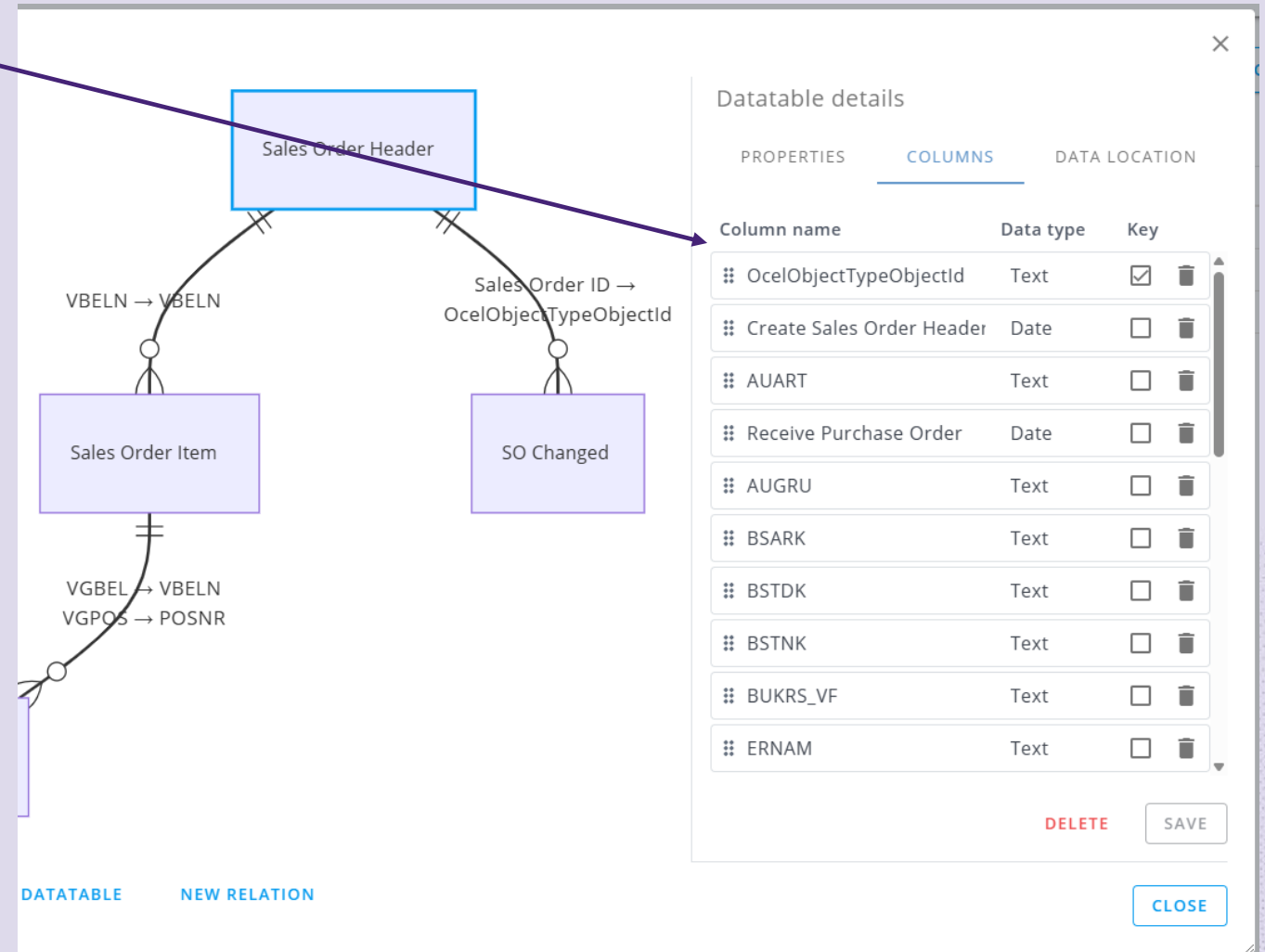
# Data schema dialog to visualize and modify datatables and relations

- Data schema dialog shows project's datatables and relations (=foreign keys)
  - Only Snowflake datatables are shown
  - Object-centric models' Events, Objects, EventToObject and ObjectToObject datatables are not shown
- Relation panel shows selected relation properties and allows to modify and delete relation
- Datatable panel shows selected datatable properties and allows to modify and delete datatable
  - Snowflake's database, schema and table can be set to bind datatable to Snowflake table
  - **Refresh from datasource** button updates columns from Snowflake



# Configurable datatable columns in Data schema dialog

- Datatable columns can be added, modified, deleted, and reordered
- **Data type** can be set when adding column, but not changed for existing column
- **Key** checkbox defines which columns form the datatable primary key (i.e., columns that uniquely identifies rows)
- When columns are modified, relations are changed accordingly to keep them consistent



The screenshot displays the 'Data schema dialog' interface. On the left, a relationship diagram shows three entities: 'Sales Order Header' (top), 'Sales Order Item' (bottom left), and 'SO Changed' (bottom right). 'Sales Order Header' is connected to 'Sales Order Item' via a relationship labeled 'VBELN → VBELN'. 'Sales Order Header' is also connected to 'SO Changed' via a relationship labeled 'Sales Order ID → OcelObjectTypeObjectId'. 'Sales Order Item' is connected to an external entity (partially visible) via a relationship labeled 'VGBEL → VBELN' and 'VGPOS → POSNR'. The 'Sales Order Header' entity is highlighted with a blue border. A purple arrow points from the 'Sales Order Header' entity to the 'Datatable details' panel on the right.

The 'Datatable details' panel is titled 'Datatable details' and has three tabs: 'PROPERTIES', 'COLUMNS', and 'DATA LOCATION'. The 'COLUMNS' tab is active. It contains a table with the following columns: 'Column name', 'Data type', and 'Key'. The table lists several columns, each with a 'Key' checkbox and a delete icon.

Column name	Data type	Key
OcelObjectTypeObjectId	Text	<input checked="" type="checkbox"/>
Create Sales Order Header	Date	<input type="checkbox"/>
AUART	Text	<input type="checkbox"/>
Receive Purchase Order	Date	<input type="checkbox"/>
AUGRU	Text	<input type="checkbox"/>
BSARK	Text	<input type="checkbox"/>
BSTDK	Text	<input type="checkbox"/>
BSTNK	Text	<input type="checkbox"/>
BUKRS_VF	Text	<input type="checkbox"/>
ERNAM	Text	<input type="checkbox"/>

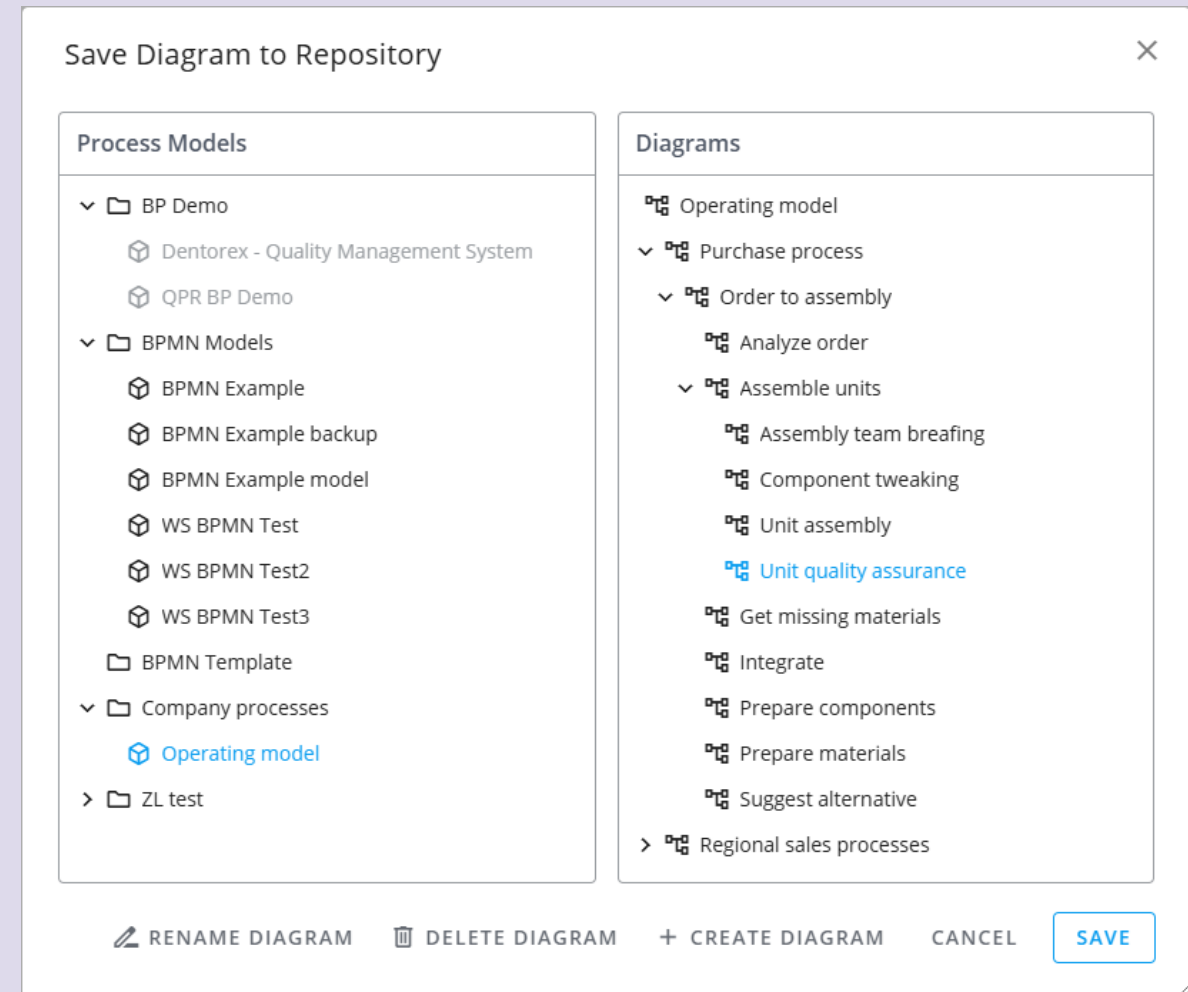
At the bottom of the 'Datatable details' panel, there are 'DELETE' and 'SAVE' buttons. Below the diagram, there are 'DATATABLE' and 'NEW RELATION' buttons. At the bottom right of the dialog, there is a 'CLOSE' button.

# BPMN diagrams can be stored and opened from EA repository



- In BPMN editor, diagrams can be opened from and stored to QPR EnterpriseArchitect repository
  - All accessible folders, EA/PD models and diagrams can be browsed
  - Non-BPMN models and diagrams are shown as non-selectable
- Diagrams in the repository can be created, deleted and renamed from QPR ProcessAnalyzer
- Opened/saved BPMN diagram contains a hierarchy of diagrams which all are opened/saved
  - When opening/saving, previously opened EA/PD diagram is preselected because diagram ID is stored to component settings
- MEA connection string is stored as project secret, allowing to use BPMN diagrams in project's models
  - MEA connection string contains MEA web service url, MEA username, and MEA password, example:

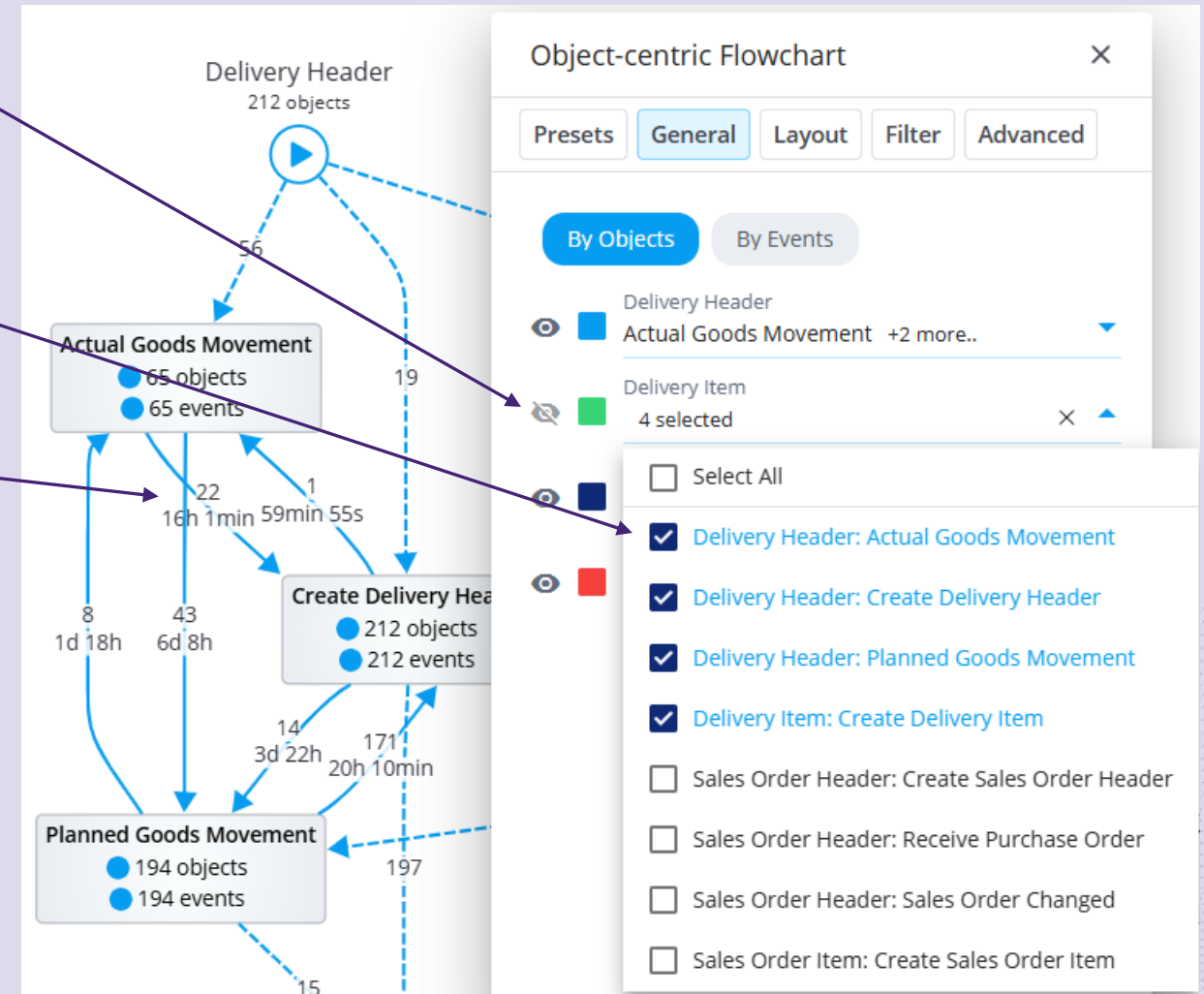
```
{
  "url": "https://hostname/QPR/Portal/QPR.Isapi.dll/wsforward/MainService.svc/webHttp",
  "logOnName": "qpr",
  "password": "demo"
}
```
- BPMN editor and selector button components store BPMN XML to their own settings
  - BPMN model can be linked to any dashboard variable with linked settings



[wiki.onqpr.com/pa/index.php/QPR\\_ProcessAnalyzer\\_BPMN\\_Editor#Opening\\_and\\_Saving\\_a\\_Diagram\\_in\\_the\\_Repository](https://wiki.onqpr.com/pa/index.php/QPR_ProcessAnalyzer_BPMN_Editor#Opening_and_Saving_a_Diagram_in_the_Repository)

# Object types can be hidden and shown easily in object-centric flowchart

- Each object type can be toggled between shown and hidden using button next to each list
  - Flowchart shows event types, flows, and start/end symbols only for enabled object types
- If every event type is linked to only one object type, event type dropdown list items are organized by object types
- Flow routing has been improved to avoid labels from overlapping



# New tool: QPR Task recorder for web applications



- QPR Task recorder for web applications records user interactions in selected web pages and turns them into event data for QPR ProcessAnalyzer
- Core purpose: Helps analyze how real tasks are performed as process instances, using case IDs, timestamps, event types, page data, element data, browser tab IDs, and user IDs
- Key captured events: Clicks, text input, selection changes, page navigation, form submissions, file selections (without file contents), and recognized keyboard shortcuts
- How it works: Runs as a Chrome or Edge browser extension and records actions on pages matching configured target URL patterns
- User control: Toolbar popup lets users enable or disable recording, see recording status, preview recent events, manage sessions, and export captured data
- Privacy features: Captured data stays local in the browser until export; it is not sent to external servers
- QPR ProcessAnalyzer ready: Exports events as CSV compatible with QPR ProcessAnalyzer datatable import and eventlog mapping
- Enterprise ready: Supports local settings, configuration import/export, and managed policy deployment

[wiki.onqpr.com/pa/index.php/QPR\\_Task\\_Recorder\\_for\\_Web\\_Apps](https://wiki.onqpr.com/pa/index.php/QPR_Task_Recorder_for_Web_Apps)

# Configurable LLM for AI Agent, expression language support for scripts, and more table header spacing



- Added system setting **AIAgentDefaultModelName** to define initial LLM for AI Agent
  - Used in Process Discovery and when adding AI Agent to dashboard
- Updated GPT-5.4 as default LLM for Open AI API (*OpenAIDefaultModelName*)
  - Used by AI Assistant and AI Agent when LLM name is empty
- Expression language API improvements
  - Added function **CreateScript** for project
  - Script objects have functions **Modify** and **DeletePermanently**
  - Filter object can be moved to different model (i.e., change *ModelId* property)
- For tables, column filter menu icon is shown when hovering header
  - Table header spacing have been adjusted to fit more text

## Create script:

```
ProjectById(1).CreateScript({  
  "Name": "My script",  
  "Code": "WriteLog(\"Hello world!\");"  
});
```

## Change script:

```
ScriptById(1).Modify({  
  "Name": "Updated script"  
});
```

## Delete script:

```
ScriptById(1).DeletePermanently();
```

[wiki.onqpr.com/pa/index.php/QPR\\_ProcessAnalyzer\\_Objects\\_in\\_Expression\\_Language#:~:text=CreateScript](https://wiki.onqpr.com/pa/index.php/QPR_ProcessAnalyzer_Objects_in_Expression_Language#:~:text=CreateScript)

**Founded**

1991

**Corporate headquarters**

Helsinki, Finland

**Customers**

Over 2000

**Stock symbol**

QPR1V:  
Nasdaq, Helsinki

**Licenses sold**

Over 1 million  
worldwide

**Products**

The Process Intelligence Suite:  
QPR ProcessAnalyzer  
QPR EnterpriseArchitect  
QPR ProcessDesigner  
QPR Metrics

**Industry recognitions**

Gartner® Magic Quadrant™ for Process Mining Platforms  
2023, 2024 & 2025  
Everest Group: Process Mining Products PEAK Matrix®  
Quadrant Knowledge Solutions:  
SPARK Matrix for Process Mining  
Forrester Research, Ventana Research, Palladium