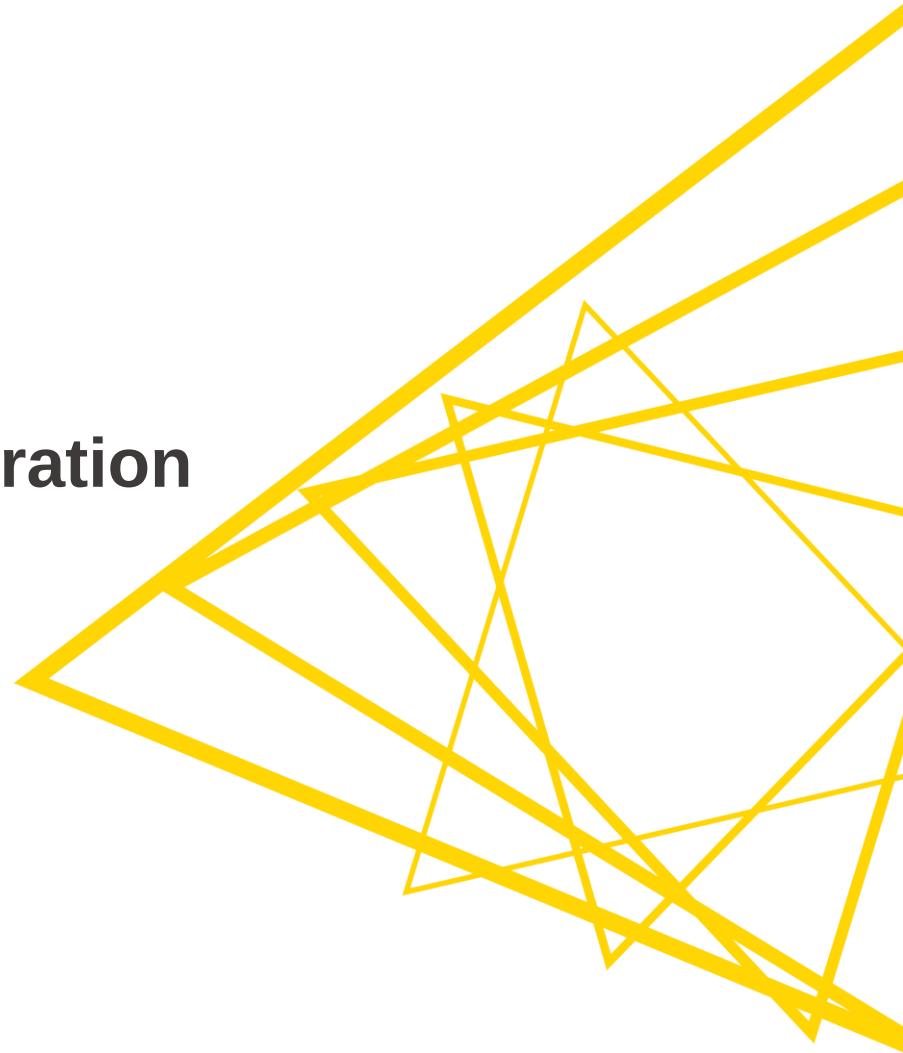
Open for Innovation
KNIME

[L3-PC] KNIME Server: Productionizing and Collaboration

KNIME GmbH



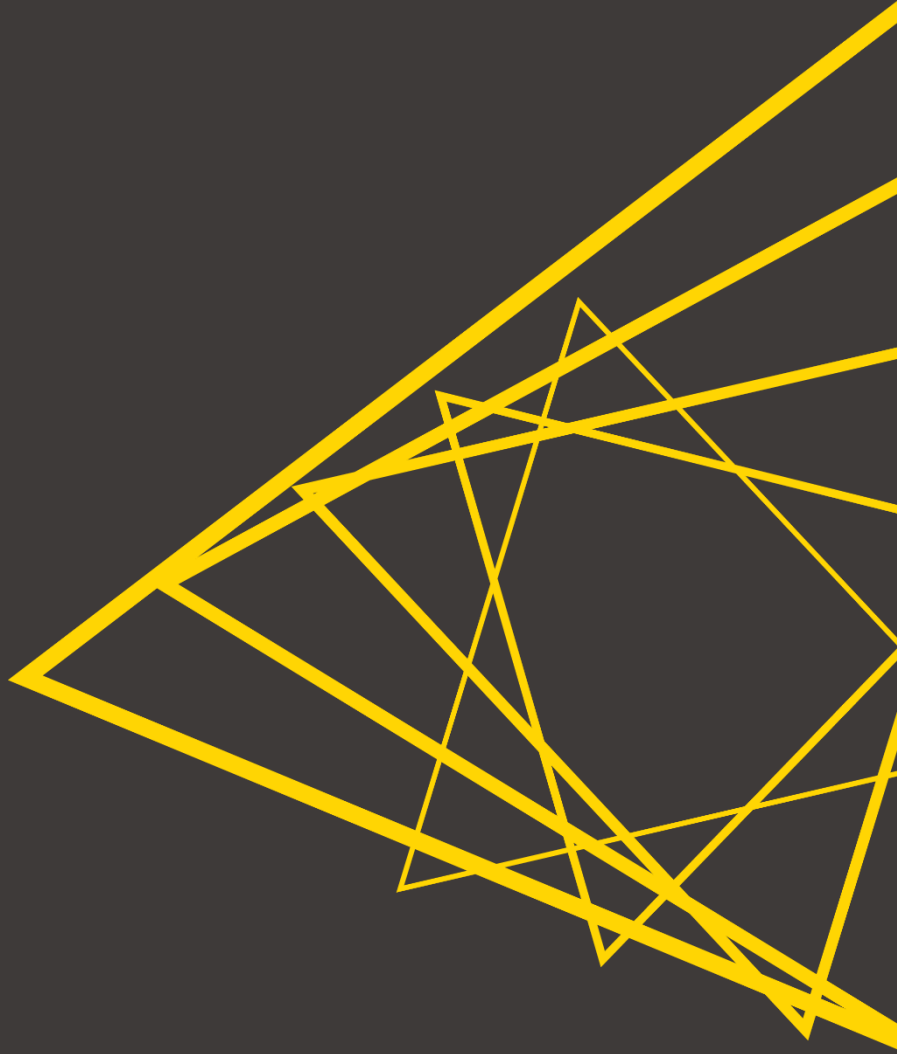
Structure of the Course

Session	Topic	Duration
Session 1	**Introduction to KNIME Software & Working with KNIME Server**	75 min
Session 2	Creating Data Apps with Components	75 min
Session 3	Deploying WebPortal Data Apps	75 min
Session 4	Integrated Deployment, REST API & Administration	75 min
Session 5	Wrap Up Session	15 min

Structure of each session

- Discussion of past exercises
- Course
- Introduction of next exercises

KNIME Software - Overview



KNIME Software – One Ecosystem

Create

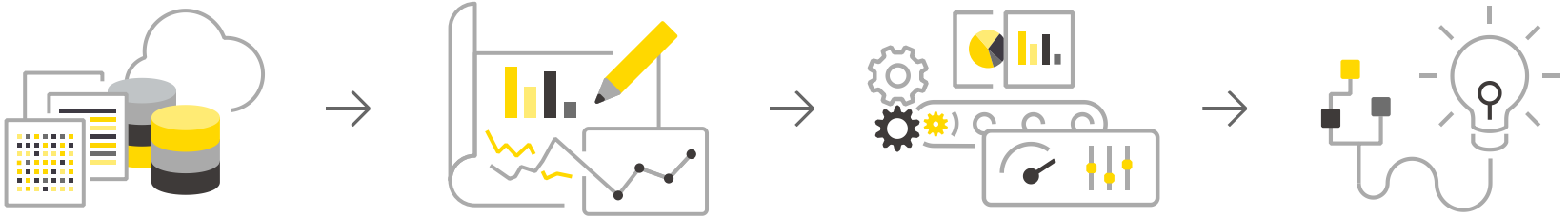
Productionize

Blend & Transform

Model & Visualize

Deploy & Manage

Consume & Interact



KNIME Analytics Platform

- KNIME Extensions
- KNIME Integrations
- Community Extensions
- Partner Extensions

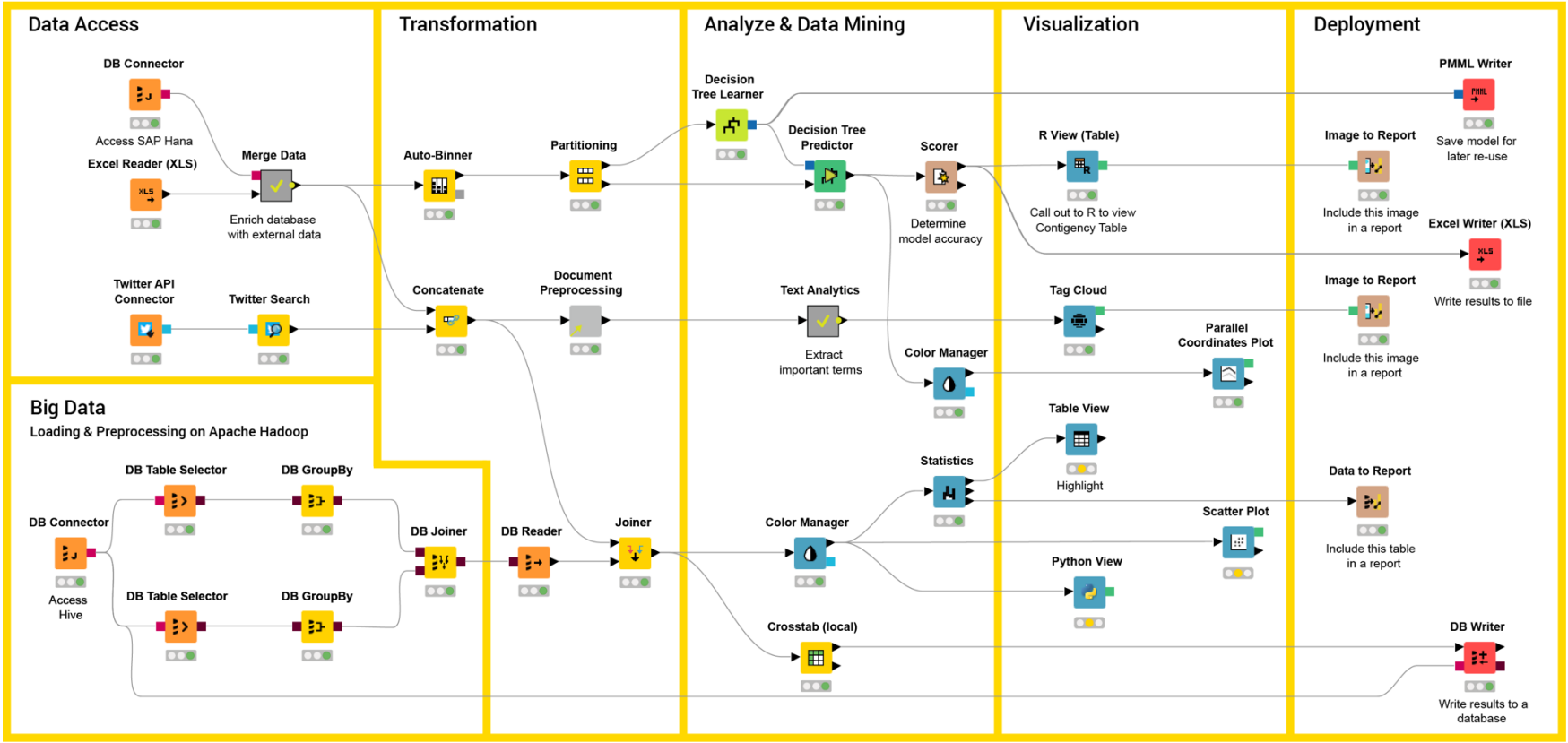
KNIME Server

- Team Collaboration
- End User Applications
- API Services
- Managed Execution

Open for Innovation KNIME

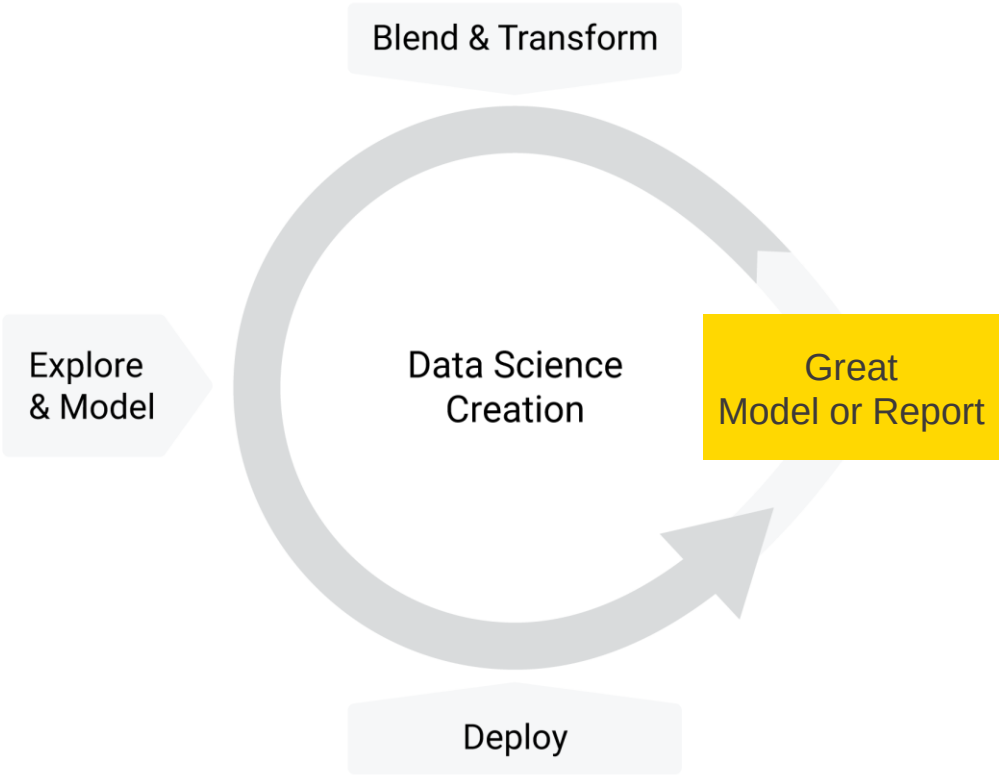
KNIME Analytics Platform: loved by individuals

3000+ Nodes for all Steps of “End-To-End” Data Science



Open for Innovation KNIME

Data Science for the Individual: Create



Data Science Practice: Multiple Stakeholders' Needs

Data Engineers
Data Science "coders" (Python, etc.)
Data Science Specialists
Data Science Visual workflow / generalists

Smart Business Users (more than Excel)
Application Users – Interaction required
Application Users – Made to spec
Report Consumers

Model / ML Operations
Feeding production systems

Operations Consumption
(Applications, Systems, Edge, etc.)

IT Operations
- Centralized resources / strategies
- Standards and preferred platforms used, infrastructure options
- Exit strategies

IT Security
- Data, applications

Financial / Risk Oversite
- Costs allocation

Compliance officer
Data/model Governance, traceability, GDPR

KNIME Software - one Ecosystem

Create

Productionize

Gather & Wrangle

Model & Visualize

Deploy & Manage

Consume & Optimize

Open Source
KNIME Analytics Platform

- KNIME Extensions
- KNIME Integrations
- Community Extensions
- Partner Extensions

Commercial
KNIME Server

- KNIME WebPortal
- Data Science as a Service

Data Science Practice: Teams !

Data Engineers
Data Science “coders” (Python, etc.)
Data Science Specialists
Data Science Visual workflow / generalists

Smart Business Users (more than Excel)
Application Users – Interaction required
Application Users – Made to spec
Report Consumers

Model / ML Operations
Feeding production systems

Operations Consumption
(Applications, Systems, Edge, etc.)

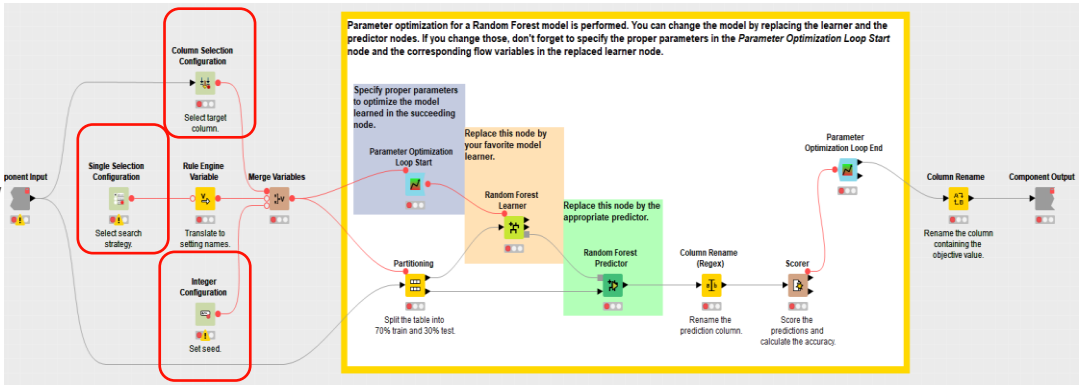
IT Operations
Centralized resources / strategies
Standards and preferred platforms used, infrastructure options
Exit strategies
IT Security
Data, applications

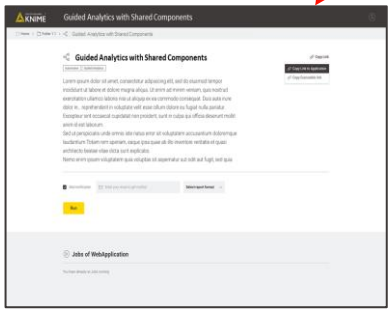
Financial / Risk Oversite
Costs allocation
Compliance officer
Data/model Governance, traceability, GDPR

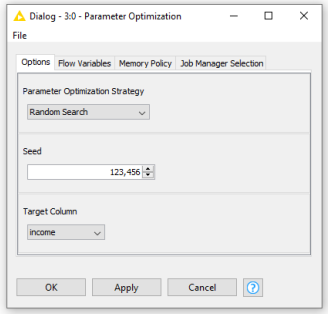
Reuse, Share and Document

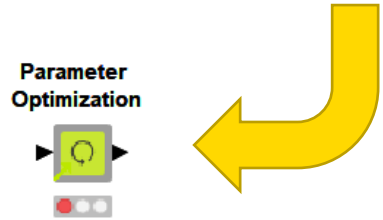
Workflows & Components build all types of applications – automated and interactive

Workflow









Component

Features:

- Self Documenting
- No limits: All nodes
- DB, Spark, DL, Python etc.
- Task packaging
- Mix and Match
- Sharable / Reusable / Instantiated

https://www.knime.com/blog/knime-analytics-platform-40-components-are-for-sharing

Capture IP, Leverage Specialists, Collaborate

Sharable / Reusable / Instantiated Workflows, Components and Collaborative Development

Database Specialists

Data Engineers

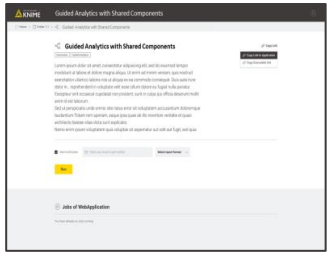
Python Specialists

Data Science Specialists

Visualization Specialists

Data Governance

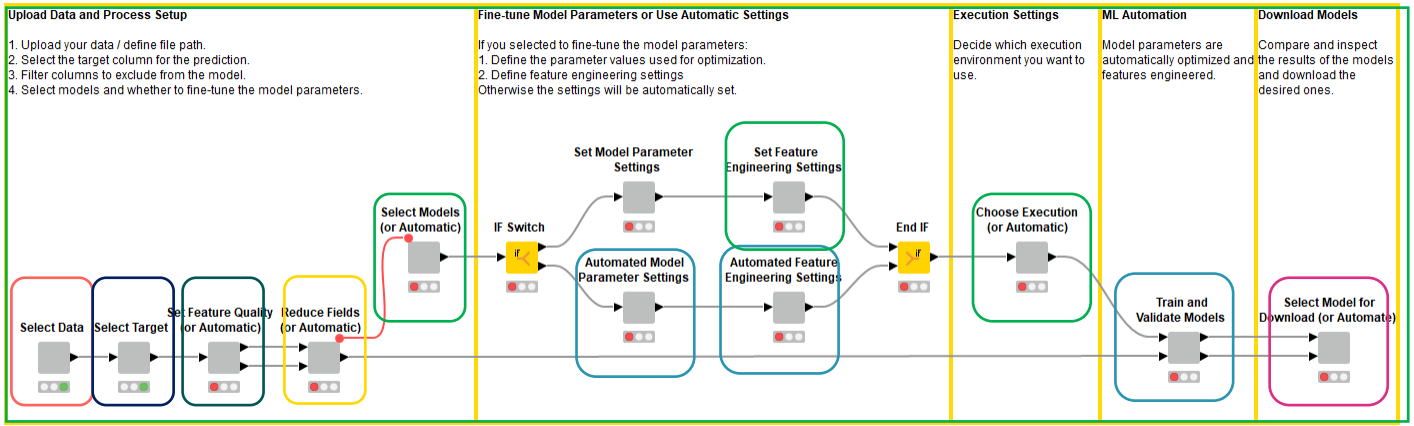
Data Science Generalists



As a web-based application

Guided Machine Learning

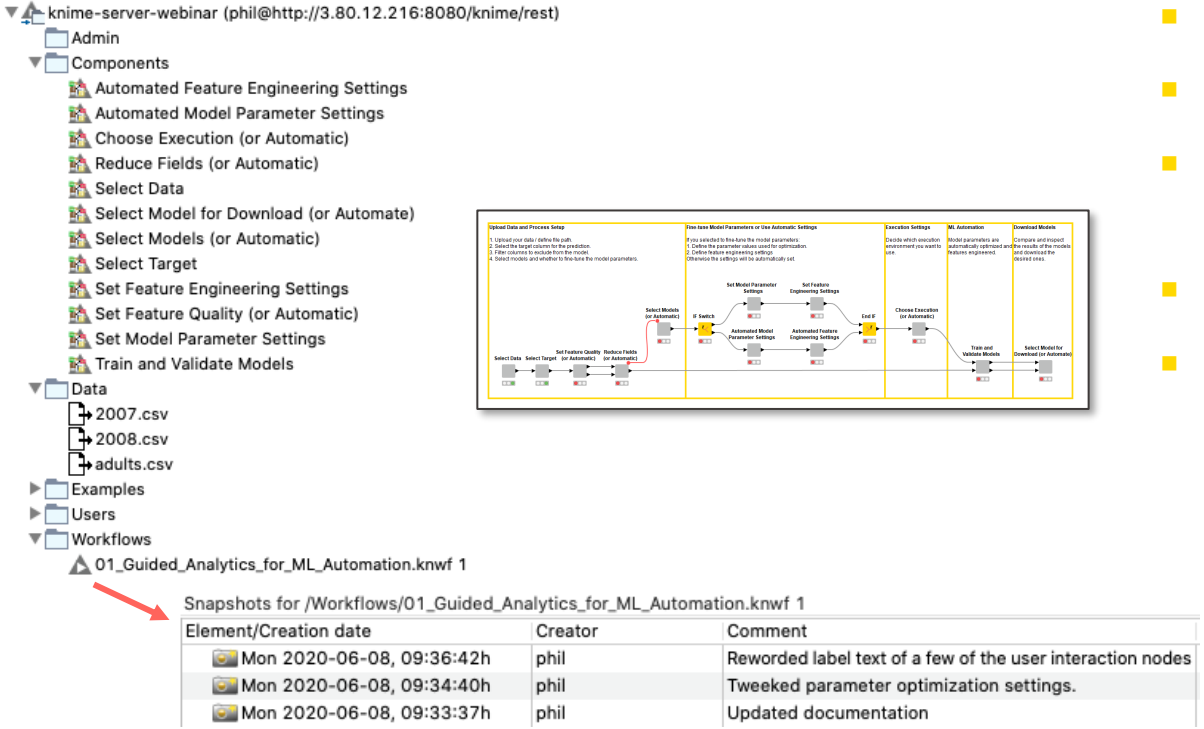
within a workflow (manual/automated)



Open for Innovation KNIME

Recoverable, Backward Compatible, No Migration

Workflows, Components and Collaboration



Features:

- Instantiated / Updated
- Secure (create, use, etc.)
- Versioned and snapshots
  - Version comparison
- Locked / Encrypted
- Reproducible
  - Guaranteed backward compatible

https://www.knime.com/blog/knime-meets-knime-will-they-blend

Data Science Practice: Multiple Stakeholders' Needs

Data Engineers
Data Science "coders" (Python, etc.)
Data Science Specialists
Data Science Visual workflow / generalists

Smart Business Users (more than Excel)
Application Users – Interaction required
Application Users – Made to spec
Report Consumers

Model / ML Operations
Feeding production systems

Operations Consumption
(Applications, Systems, Edge, etc.)

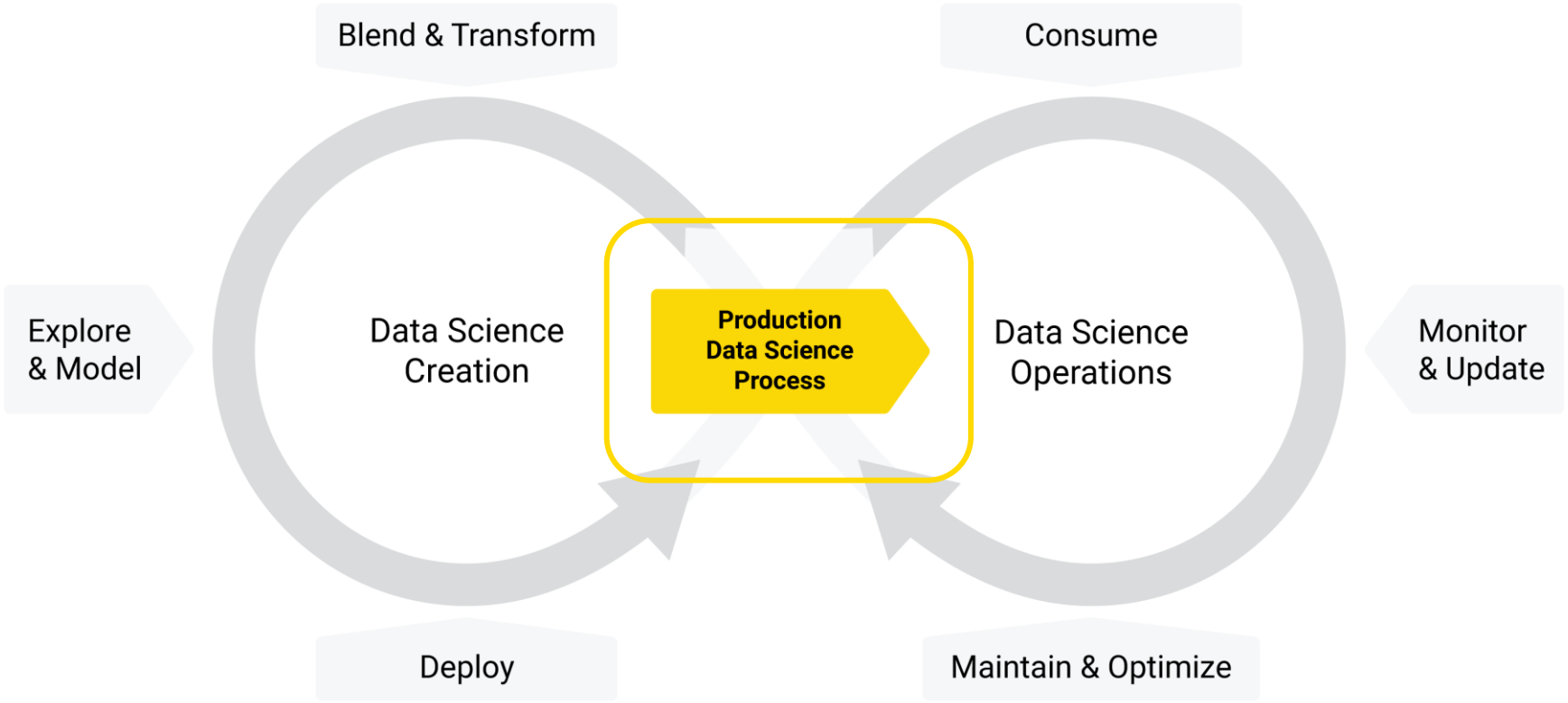
IT Operations
- Centralized resources / strategies
- Standards and preferred platforms used, infrastructure options
- Exit strategies

IT Security
- Data, applications

Financial / Risk Oversite
- Costs allocation

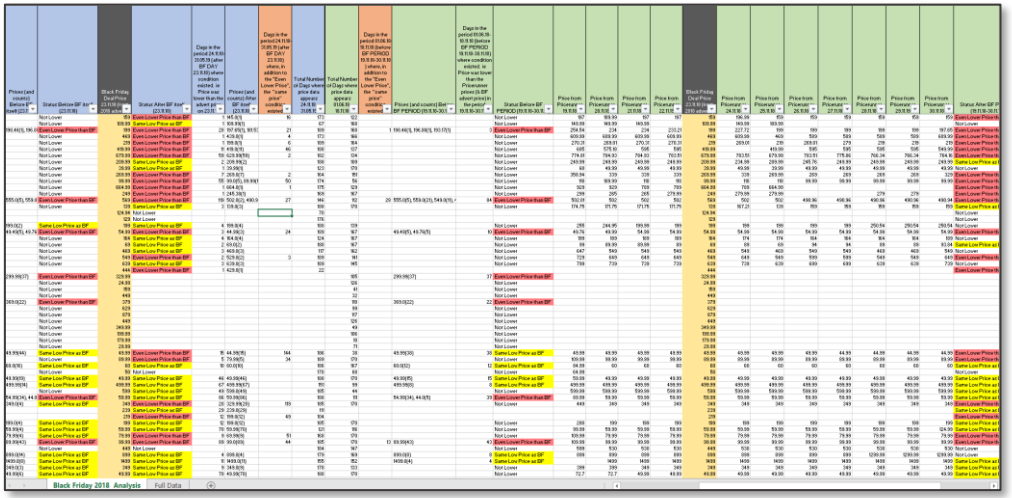
Compliance officer
Data/model Governance, traceability, GDPR

Data Science for the Business: Creation & Production



Empower Business Users Appropriately

Delivering reports and output to business users appropriately



https://www.knime.com/community/continental-nodes-for-knime-xls-formatter

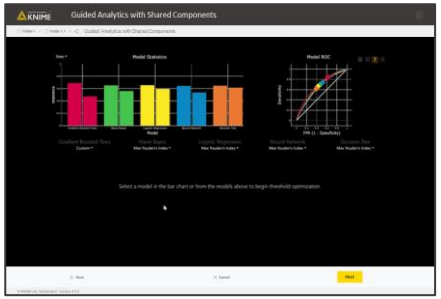
Features:

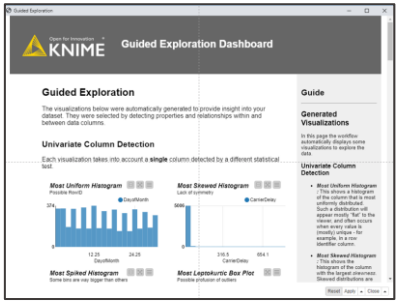
- Visualizations
  - Plotly, JavaScript, etc.
- Reports Creation BIRT
- Integration with
  - Excel
    - Functionality exploitation not just CSVs
  - PowerBI
  - Tableau
  - Qlik
  - Spotfire
  - …

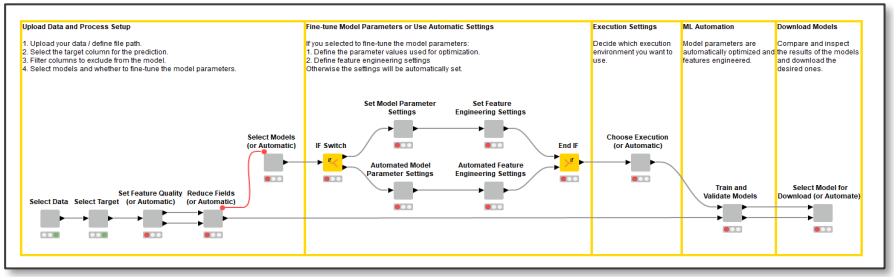
Empower Business Users Appropriately

Guided Analytics for Building Applications

*Appropriate levels of Automation & Human Interaction at any stage of the Data Science Life Cycle depending on **task** and **audience***







Features:

- Workflows and webportal nodes build interactive applications & dashboards
- KNIME WebPortal manages access

https://www.knime.com/blog/principles-of-guided-analytics

Open for Innovation KNIME

Data Science Practice: Multiple Stakeholders' Needs

Data Engineers
Data Science "coders" (Python, etc.)
Data Science Specialists
Data Science Visual workflow / generalists

Smart Business Users (more than Excel)
Application Users – Interaction required
Application Users – Made to spec
Report Consumers

Model / ML Operations
Feeding production systems

Operations Consumption
(Applications, Systems, Edge, etc.)

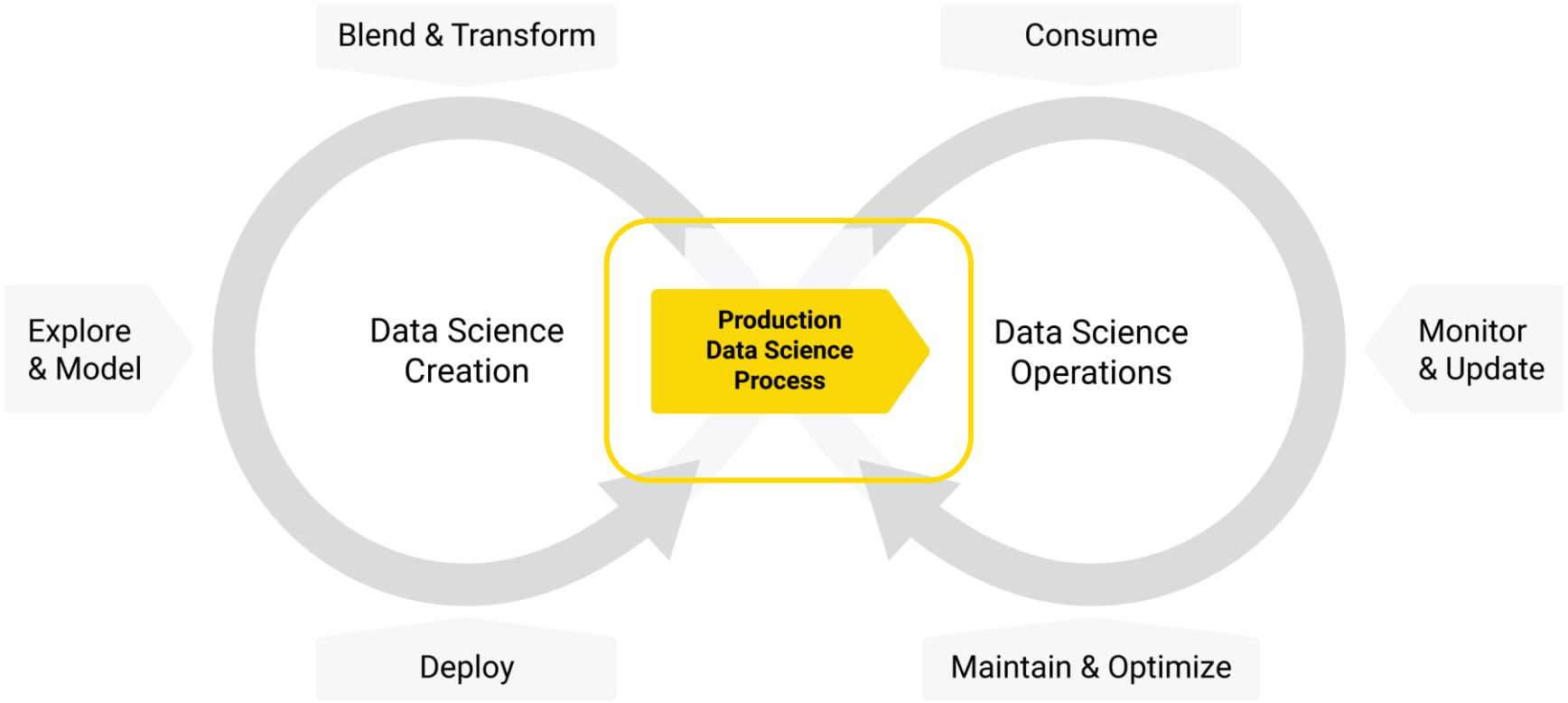
IT Operations
- Centralized resources / strategies
- Standards and preferred platforms used, infrastructure options
- Exit strategies

IT Security
- Data, applications

Financial / Risk Oversite
- Costs allocation

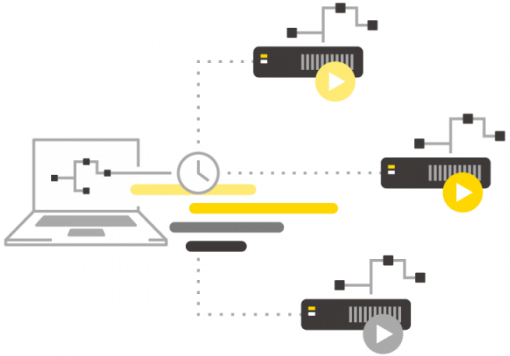
Compliance officer
Data/model Governance, traceability, GDPR

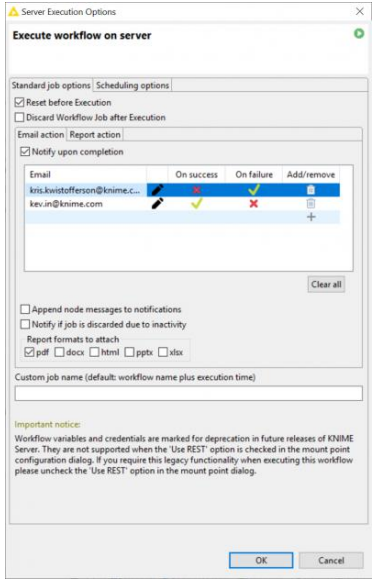
Data Science for the Business: Creation & Production



Flexible Delivery Options: Automate

Automated workflow execution





Features:

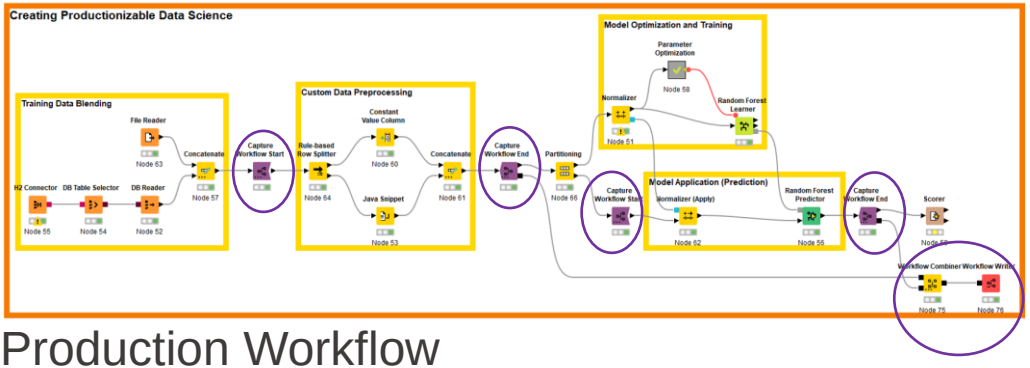
- Scheduled
- Triggered
- Called (Rest / SAAS)
- Call Actions based on status
- Scale and Pin Execution
- View, edit, execute workflows remotely

https://docs.knime.com

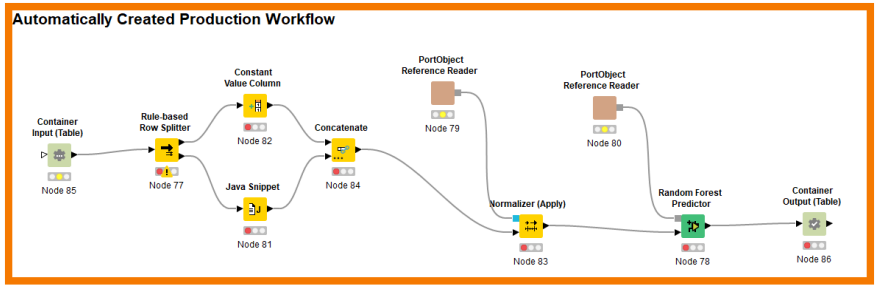
Continuous Integration / Continuous Deployment

Integrated Deployment

Creation Workflow



Production Workflow



https://www.knime.com/integrated-deployment

Features:

- Eliminates the gap between Creation and Production
  - mark what's necessary for production
  - not just the model
- no manual intervention
  - No rewrite, copy paste
- Capture all nodes & settings
- Automatically created production workflow
- Always in sync

Data Science Practice: Multiple Stakeholders' Needs

Data Engineers
Data Science "coders" (Python, etc.)
Data Science Specialists
Data Science Visual workflow / generalists

Smart Business Users (more than Excel)
Application Users – Interaction required
Application Users – Made to spec
Report Consumers

Model / ML Operations
Feeding production systems

Operations Consumption
(Applications, Systems, Edge, etc.)

IT Operations
- Centralized resources / strategies
- Standards and preferred platforms used, infrastructure options
- Exit strategies

IT Security
- Data, applications

Financial / Risk Oversite
- Costs allocation

Compliance officer
Data/model Governance, traceability, GDPR

Governance & Compliance

Explainability / Interpretability of models

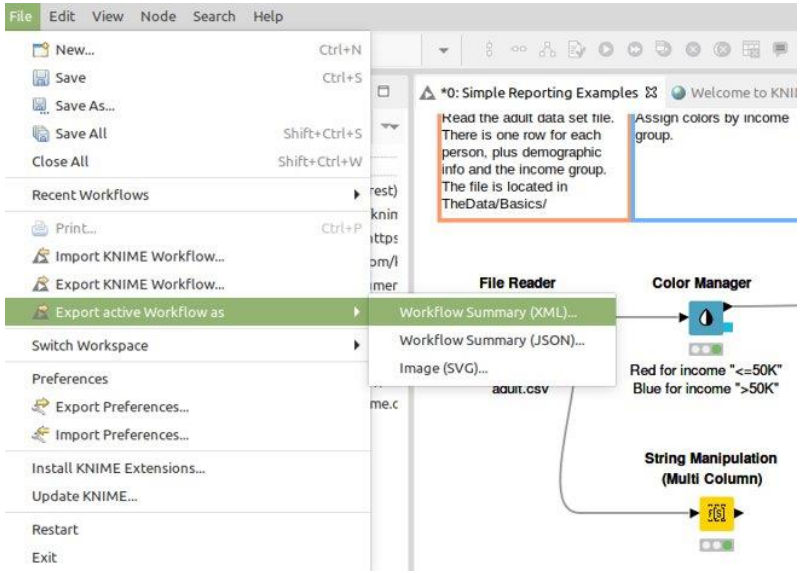
Features:

- Many Techniques available
  - LIME
  - SHAP
  - Shapley
  - Partial Dependence / ICE
  - Binary Classification Inspector

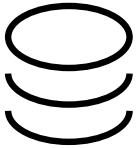
https://hub.knime.com/knime/extensions/org.knime.features.mli/latest

Governance & Compliance

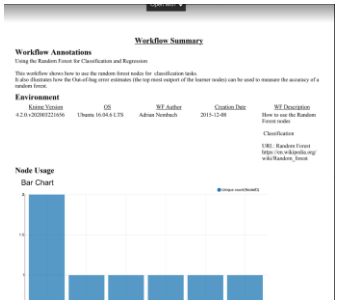
Data / Model Lineage



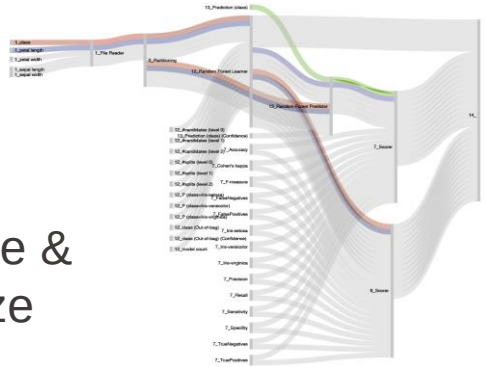
Archive



Document



Explore & Analyze



Open for Innovation KNIME

Data Science Practice: Multiple Stakeholders' Needs

Data Engineers
Data Science "coders" (Python, etc.)
Data Science Specialists
Data Science Visual workflow / generalists

Smart Business Users (more than Excel)
Application Users – Interaction required
Application Users – Made to spec
Report Consumers

Model / ML Operations
Feeding production systems

Operations Consumption
(Applications, Systems, Edge, etc.)

IT Operations
- Centralized resources / strategies
- Standards and preferred platforms used, infrastructure options
- Exit strategies

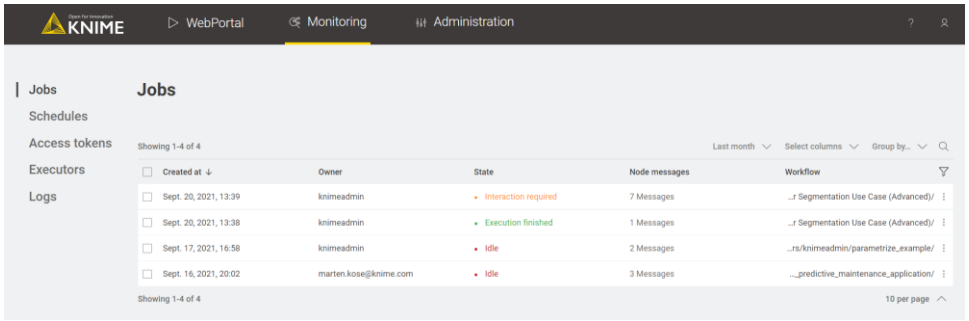
IT Security
- Data, applications

Financial / Risk Oversite
- Costs allocation

Compliance officer
Data/model Governance, traceability, GDPR

Manage Infrastructure & Users

Central Management and Monitoring capabilities

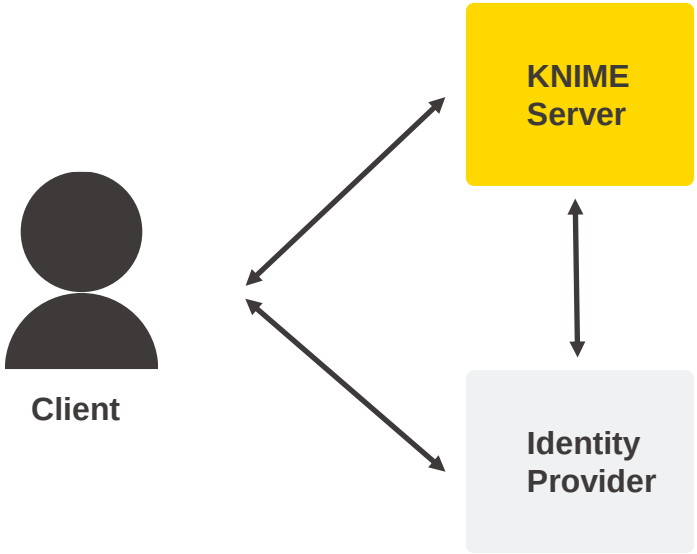


Features:

- Client Customizations
  - Custom update sites
  - Manage preferences via profiles
  - Node repository & libraries
- Monitor server activity
  - Running and scheduled jobs
  - Adjust permissions
  - Manage ongoing services

Single Sign-on, Integrate with multiple Security protocols

OAUTH, LDAP, AD Integration

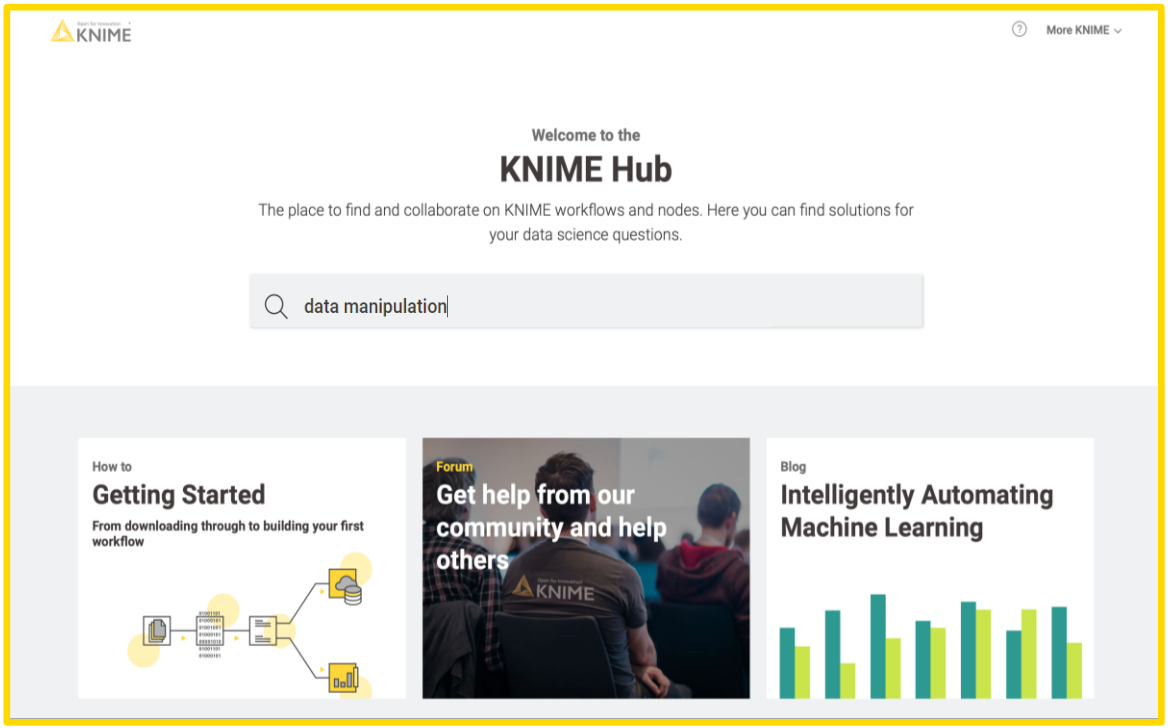


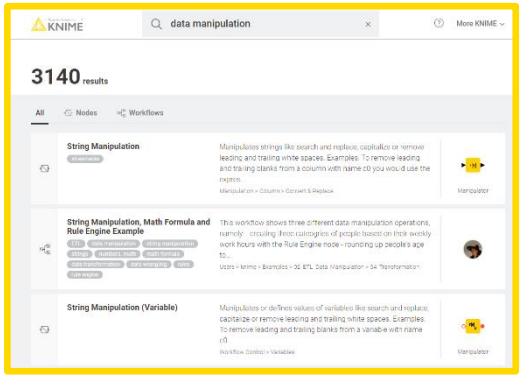
Features:

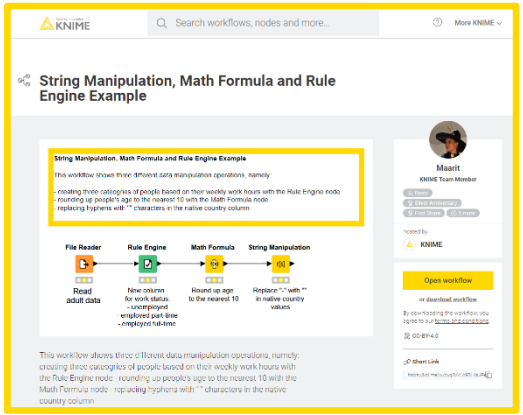
- Single sign-on (SSO) to KNIME Server
- Integrate with multiple identity providers
- Flexible configuration capabilities to map users and groups
- Manage all aspects of KNIME usage

https://docs.knime.com

KNIME Hub: Sharing Resources



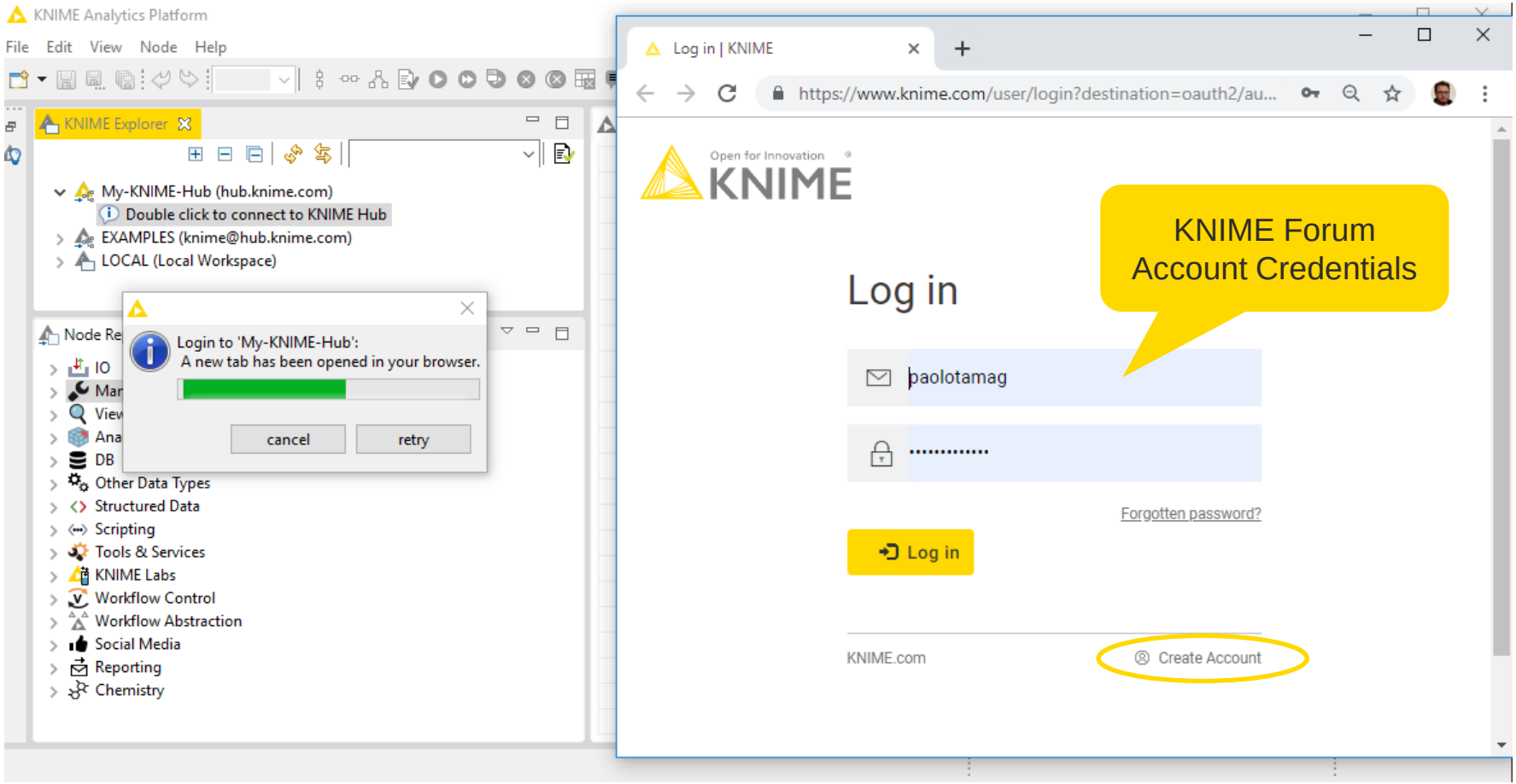




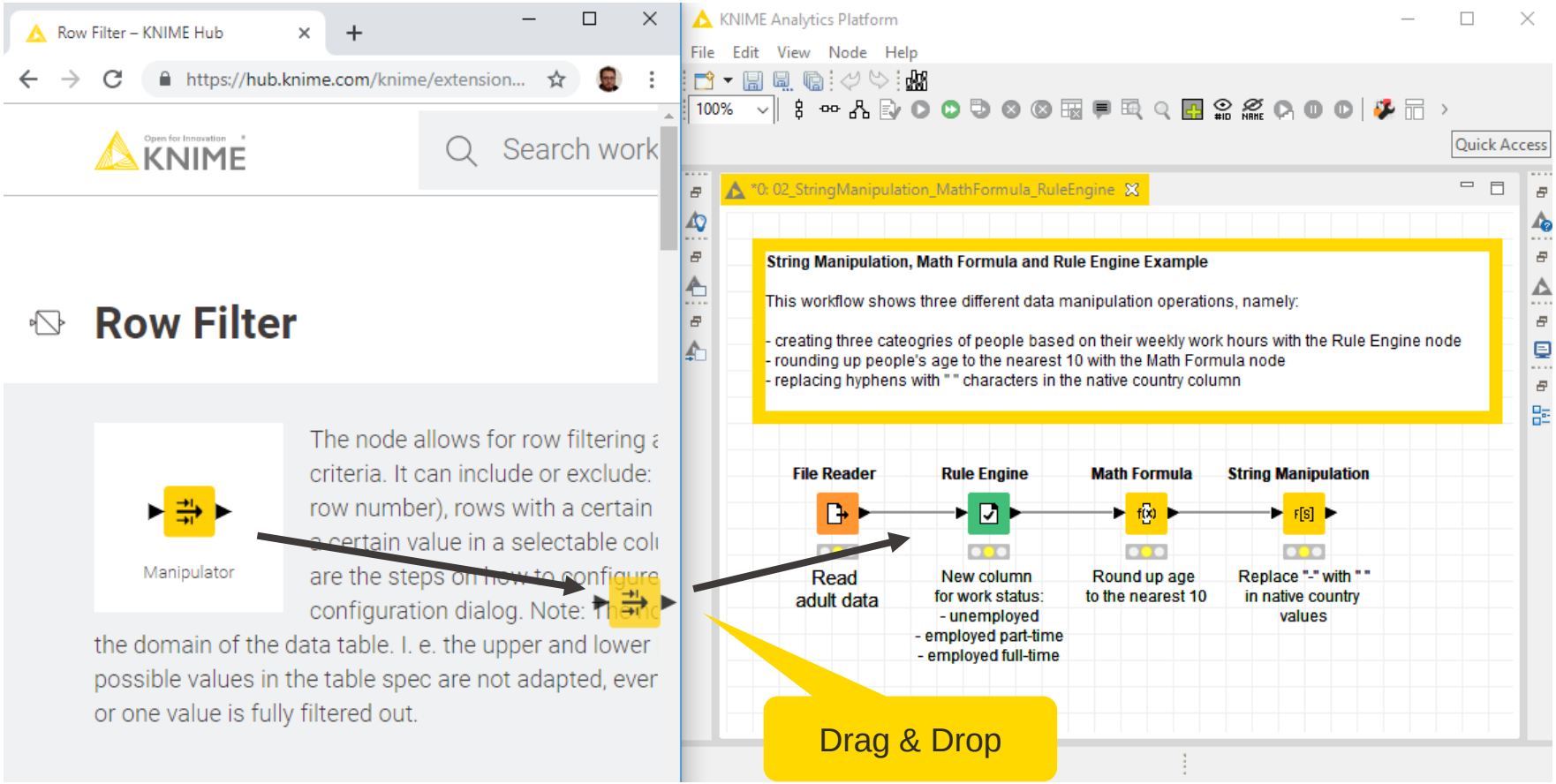
https://hub.knime.com/

Open for Innovation KNIME

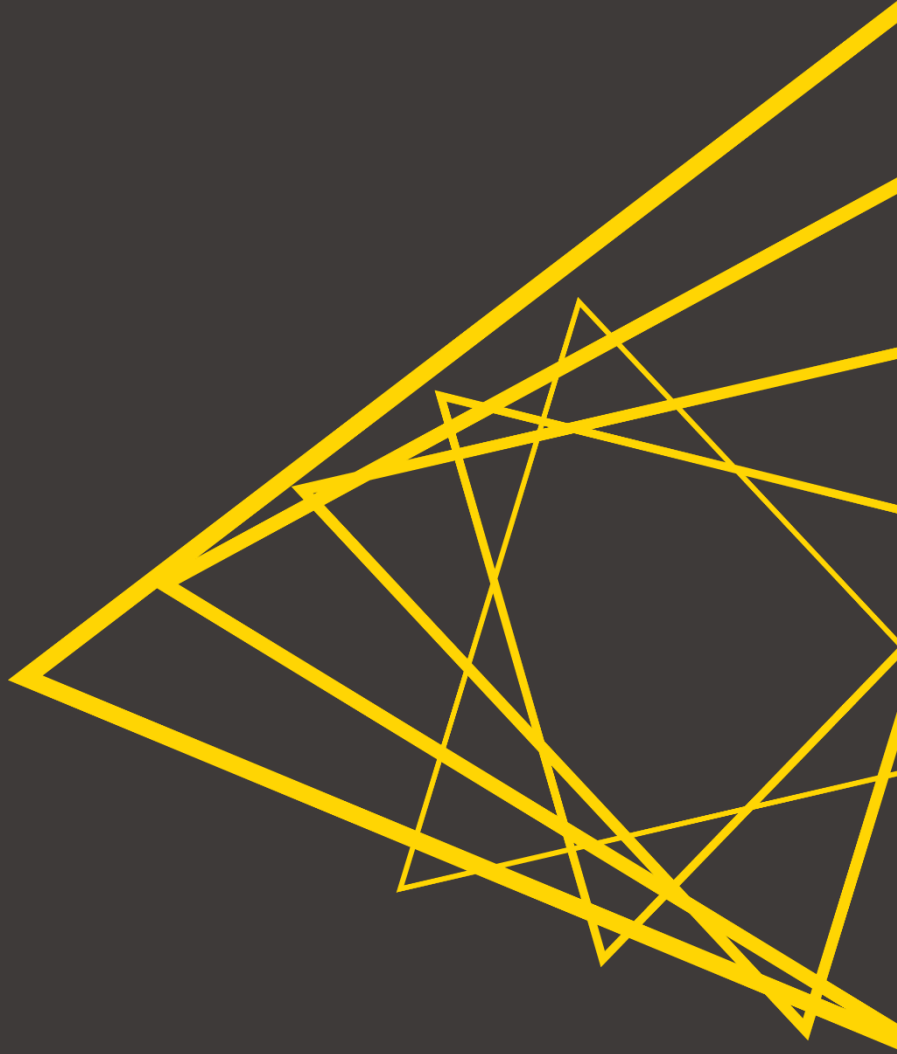
Log in to KNIME Hub and publish your Workflows



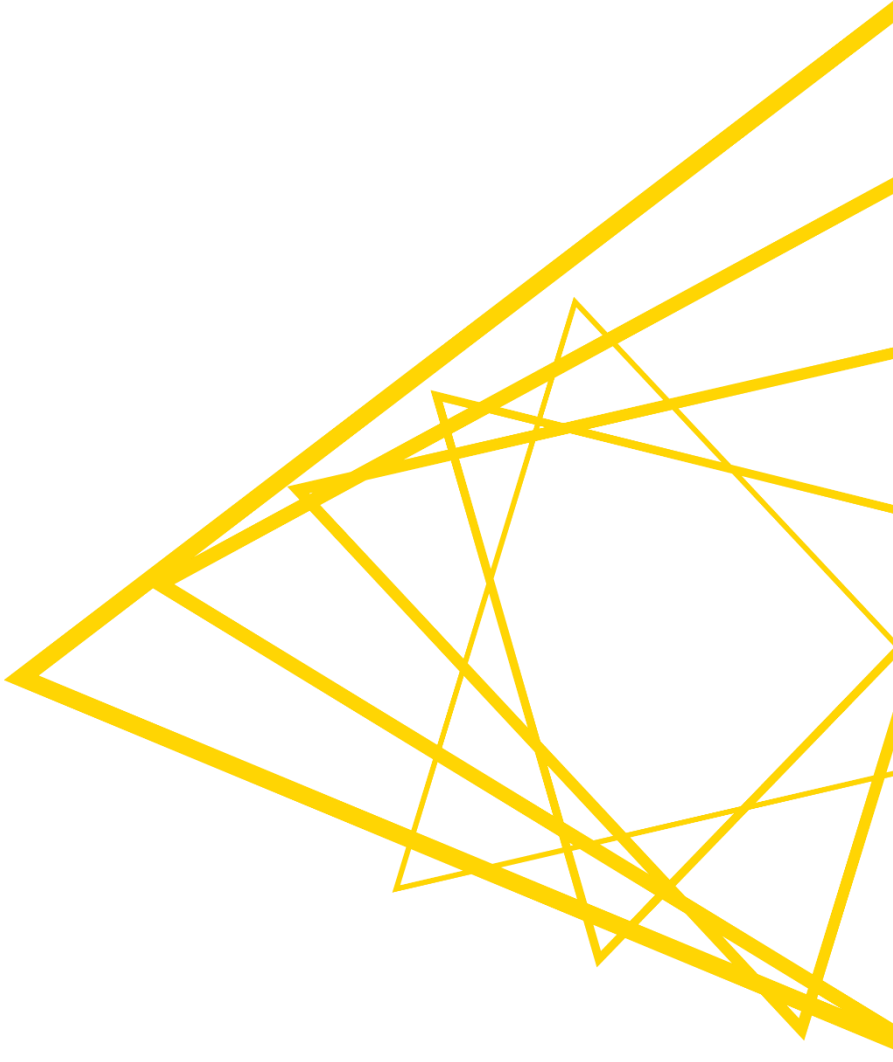
Edit the Workflow



Working with KNIME Server



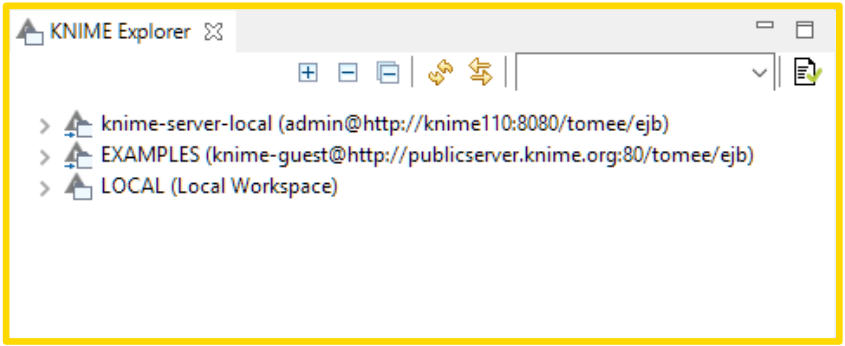
Connecting to KNIME Server

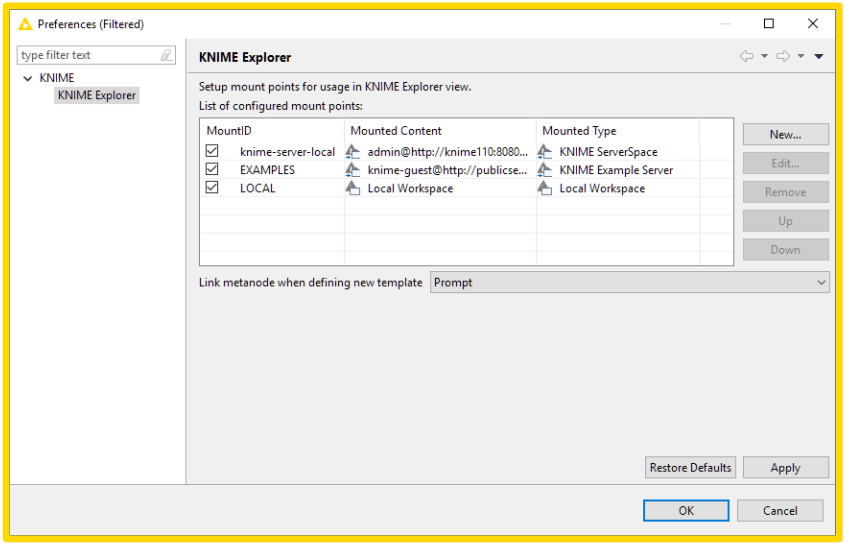


Set Up a New Mount Point

Server connections are shown as “mount points” in the KNIME Explorer. To add a new mount point simply:

1. Click the Configure button in the KNIME Explorer.
2. Click New…
3. Configure a mount point with your details

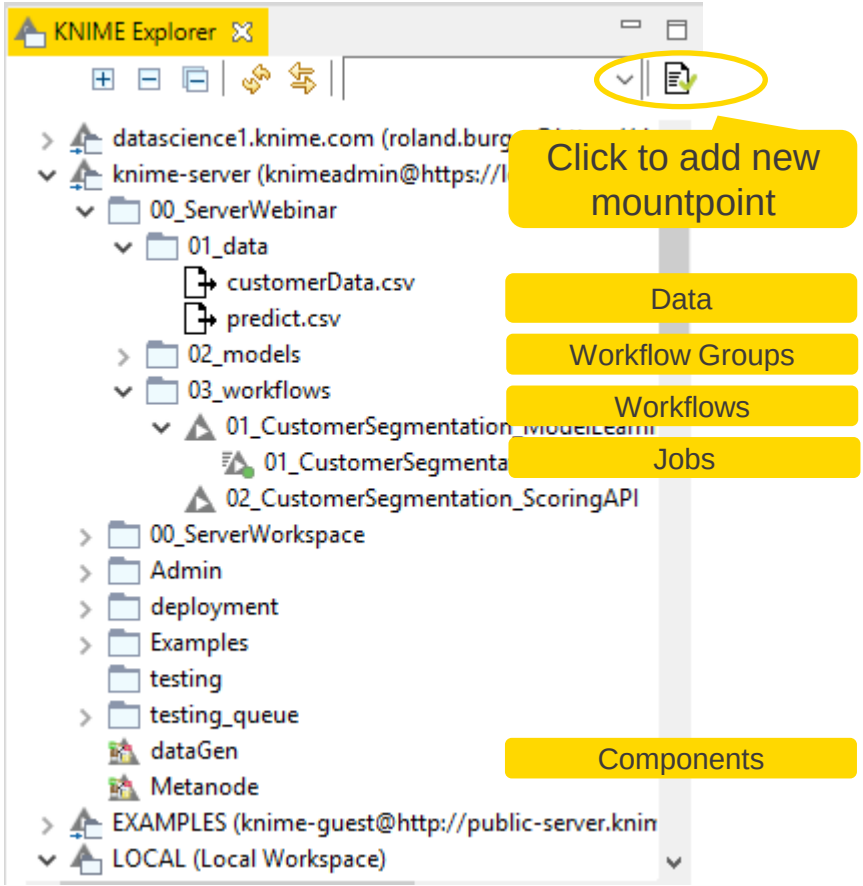




Server Mount Point as a Shared Repository (1/2)

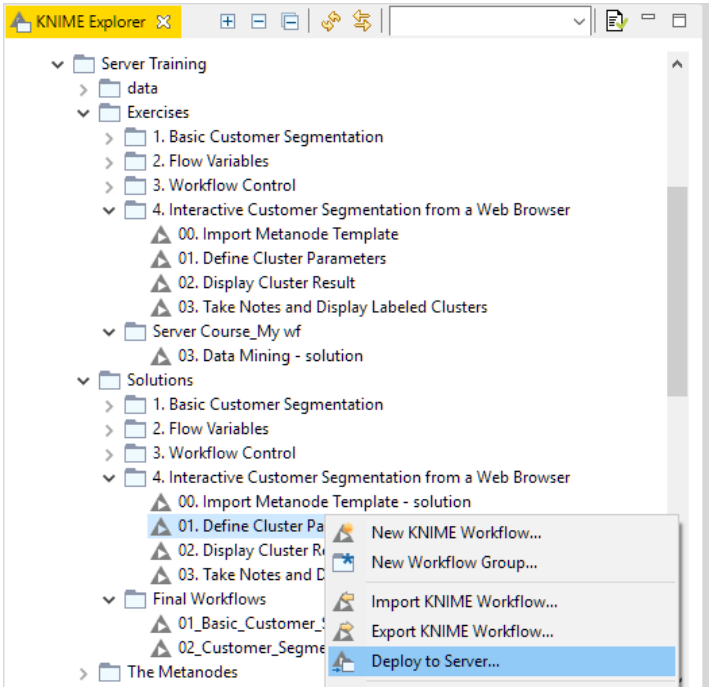
The Server provides an area in the Explorer for sharing work with your colleagues. Use workflow groups to organize your workflows, components, and data files.

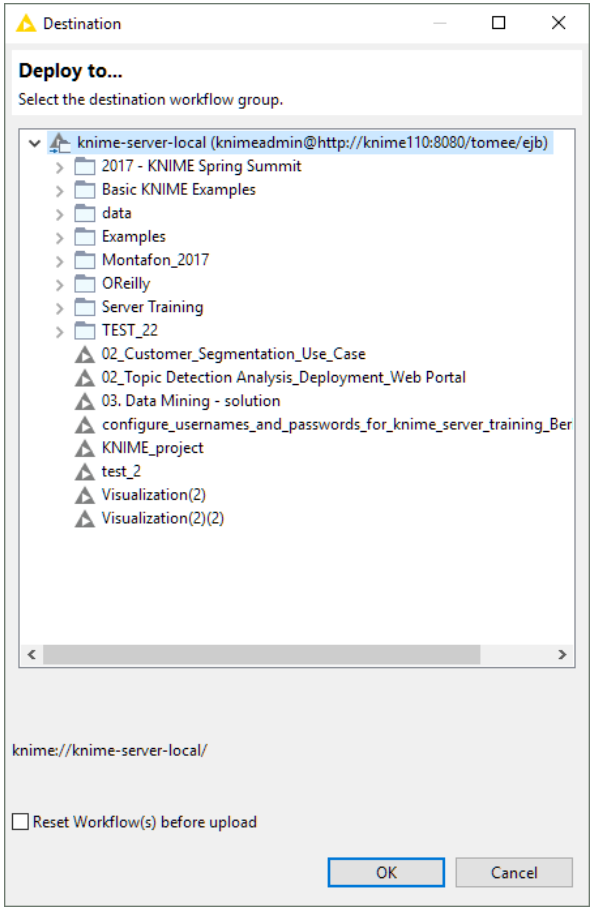
To move resources simply drag and drop or copy and paste



Server Mount Point as a Shared Repository (2/2)

Another way to deploy resources on KNIME Server..

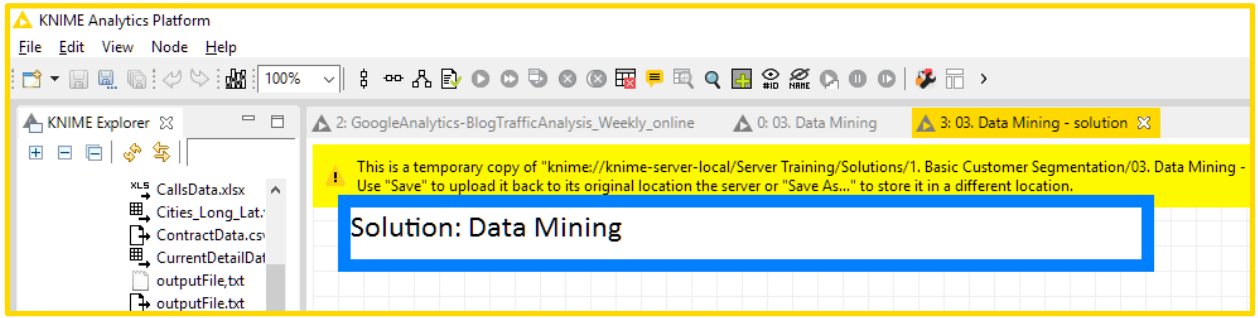




Open for Innovation KNIME

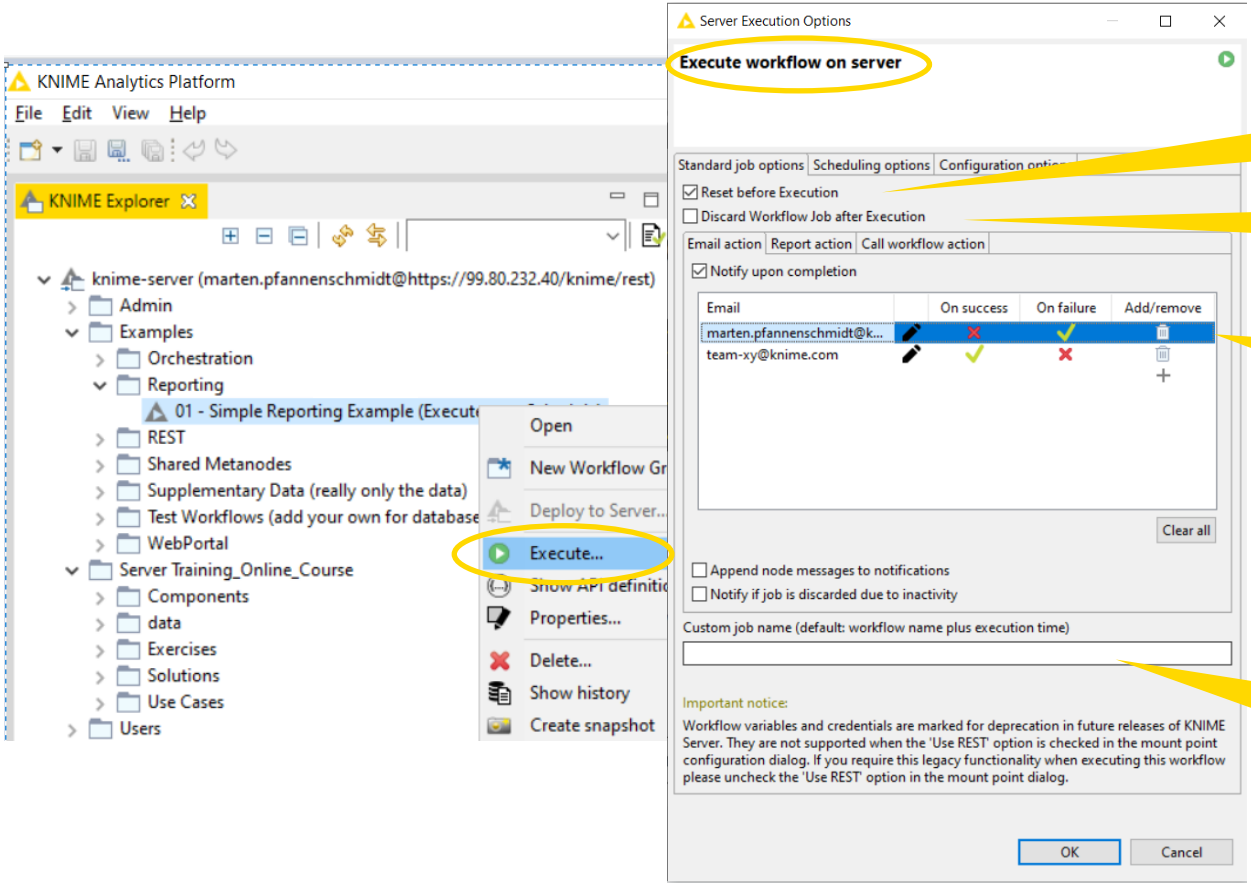
Inspecting a Workflow from KNIME Server

- By double-clicking a workflow on KNIME Server, the client downloads it (to a temporary location) and subsequently opens it automatically
- The yellow bar at the top of the editor indicates that this is a temporarily downloaded server workflow



Automation - Remote Execution

Executing a Workflow on the Server – Remote Execution



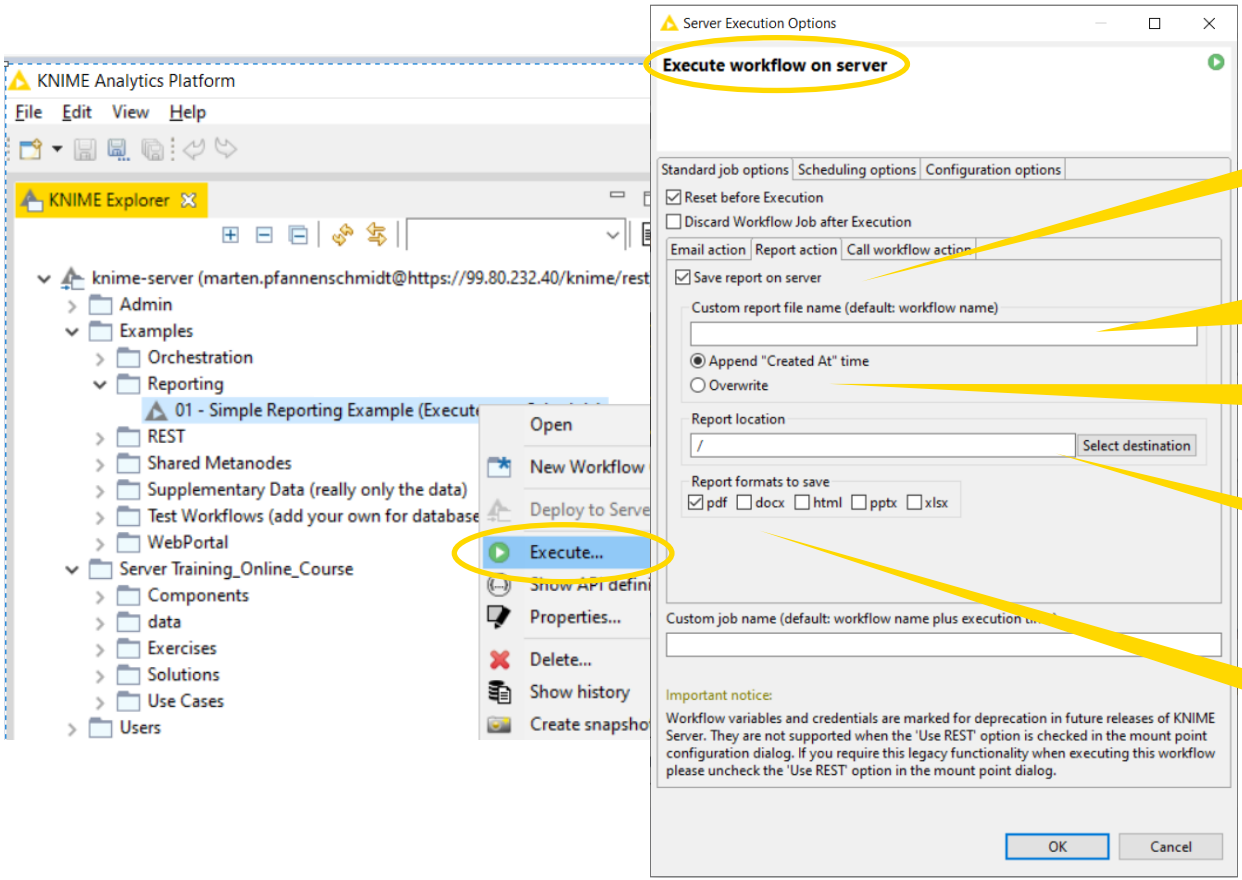
Check to reset workflow before execution. All nodes are reset (including File and Database Reader nodes, etc.).

If selected, the executed job is deleted immediately after execution, and is not saved.

Enter one or multiple email addresses (separated by commas) to which a notification is sent after the workflow execution has been finished.

The name of the workflow job as it is displayed in the server view. By default this is the name of the workflow. The execution date is always appended to the name.

Executing a Workflow on the Server – Remote Execution



If the workflow contains a report, you can select to save it on KNIME Server.

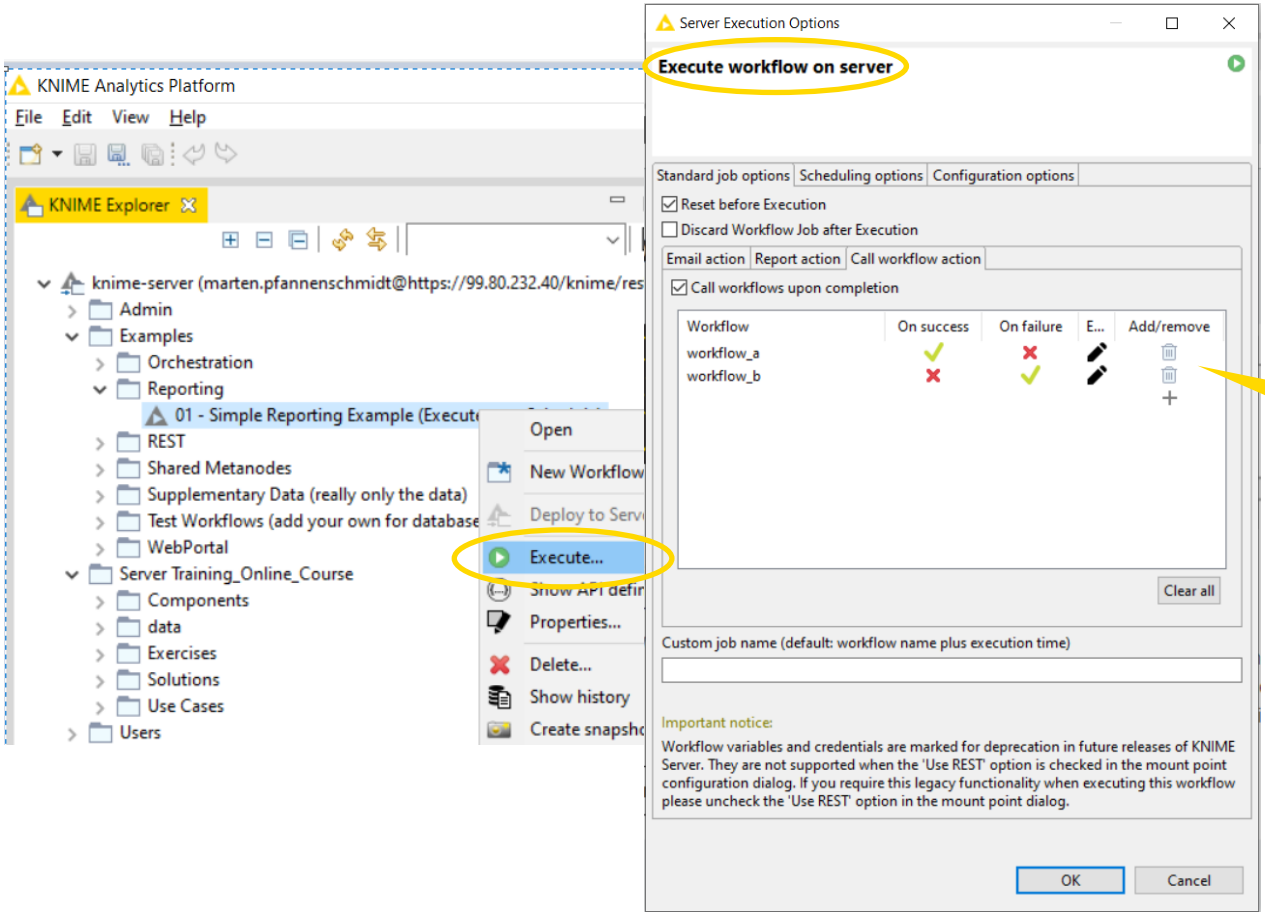
By default the name of the report is the name of the workflow. You can define a custom report file name, if you wish so.

You can overwrite the report with every execution or append a timestamp.

Here you can define the location, where you want to store the report on KNIME Server.

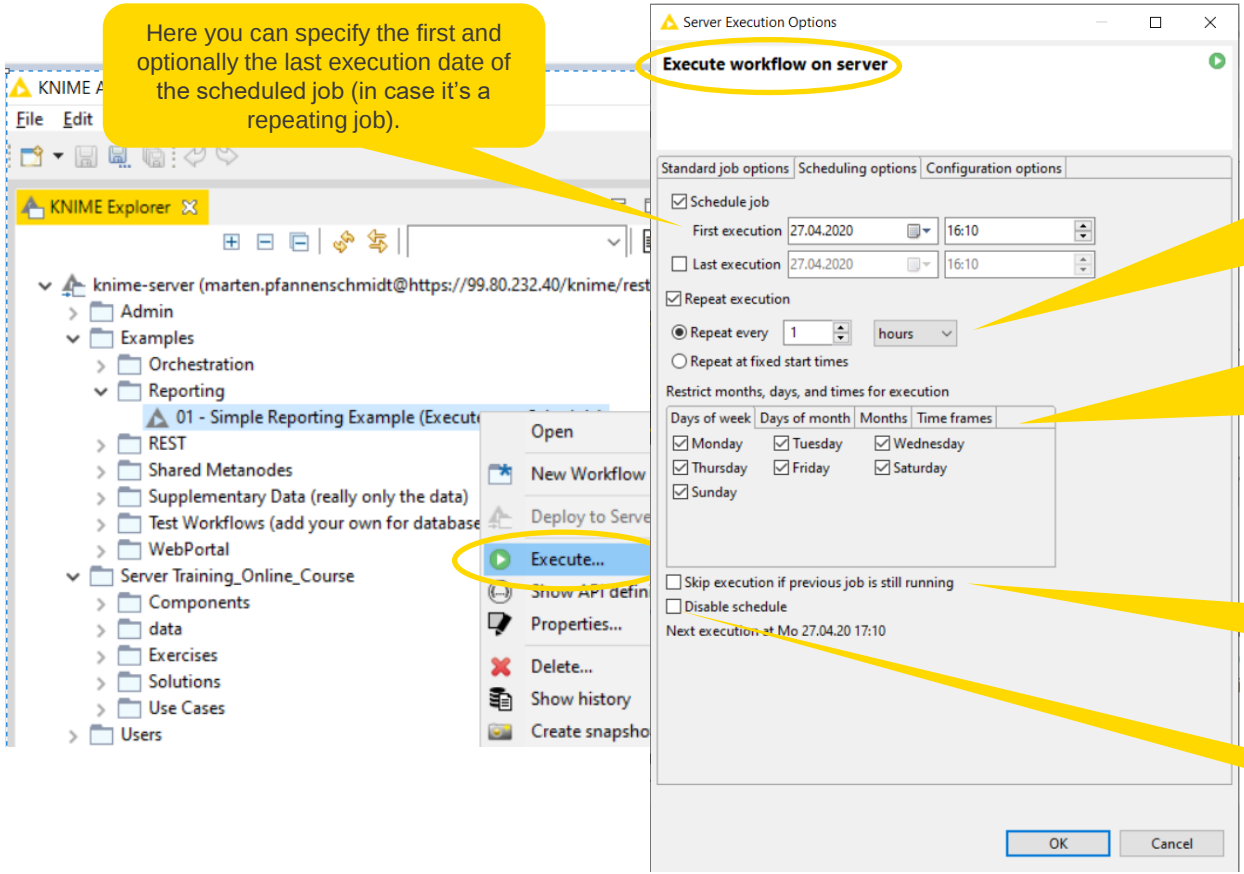
You can select in which formats the report should be saved.

Executing a Workflow on the Server – Remote Execution



Depending on whether your job executed successful or not, you can configure to run subsequent workflows

Executing a Workflow on the Server – Remote Execution



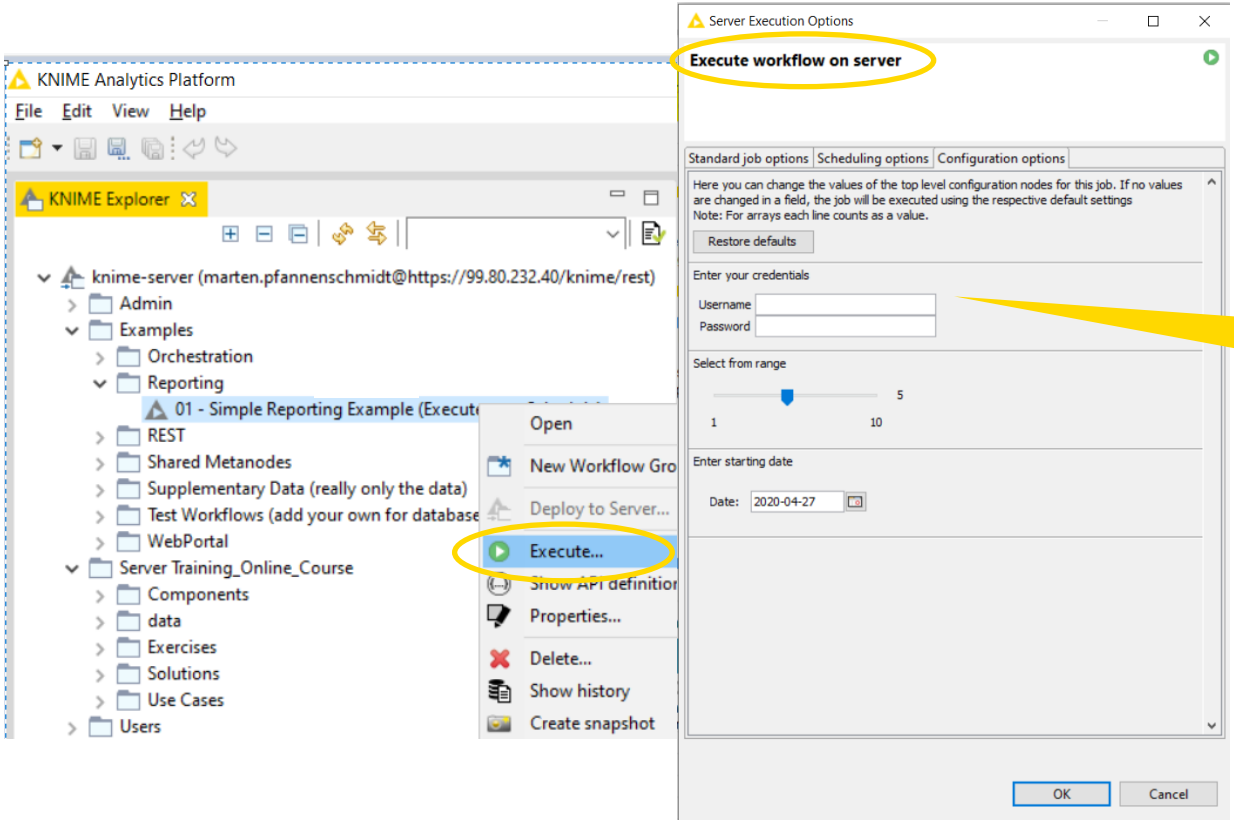
Repeating jobs can be repeated after a certain number of minutes, hours, or days. The latter takes into account daylight saving, i.e. the start hour will be the same in winter and summer (e.g. 12:00).

By default, repeating jobs are executed every day. Here you can filter whether they should run only on certain days of the week, days of the month, or only in certain months. "Last" means the last day of the month.

If the previous job is still running when the next execution is supposed to start, you can opt to skip this execution.

Scheduled jobs can be disabled temporarily.

Executing a Workflow on the Server – Remote Execution



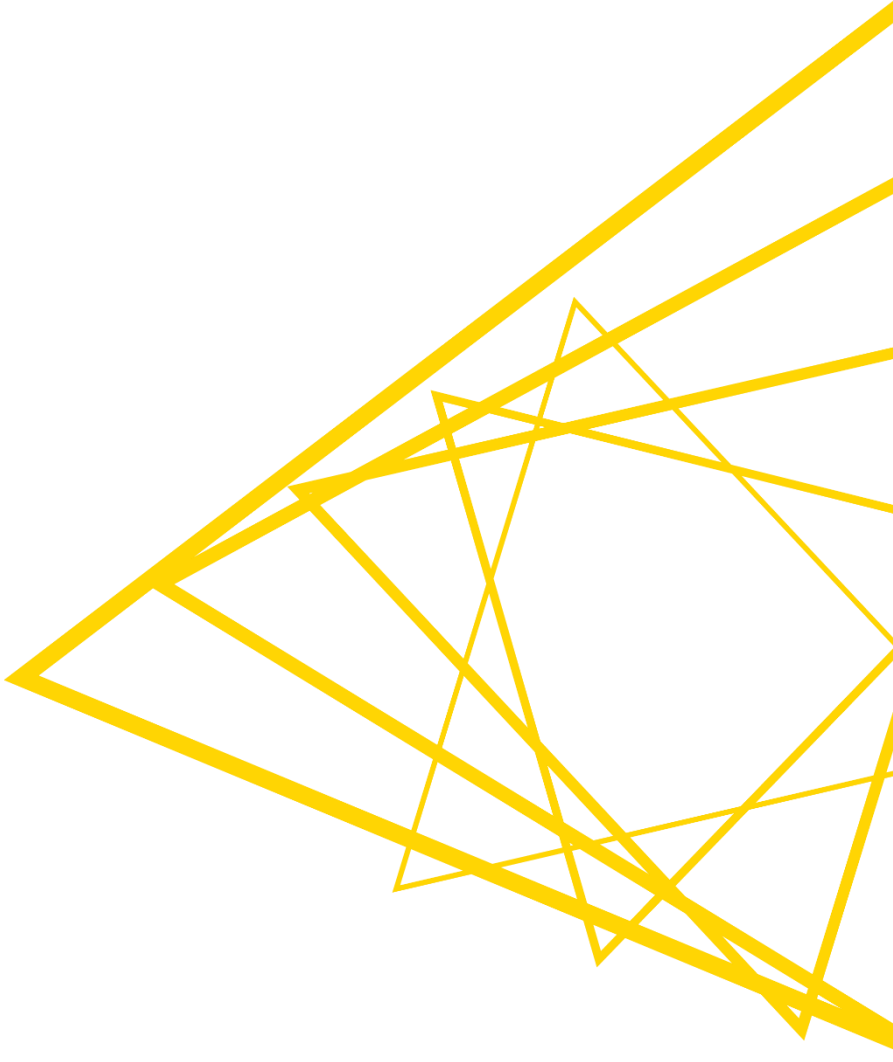
Configuration nodes from the workflows can be set via execution dialog to parametrize workflow execution

Workflow Jobs

Remotely executed workflows are run as ***Jobs***:

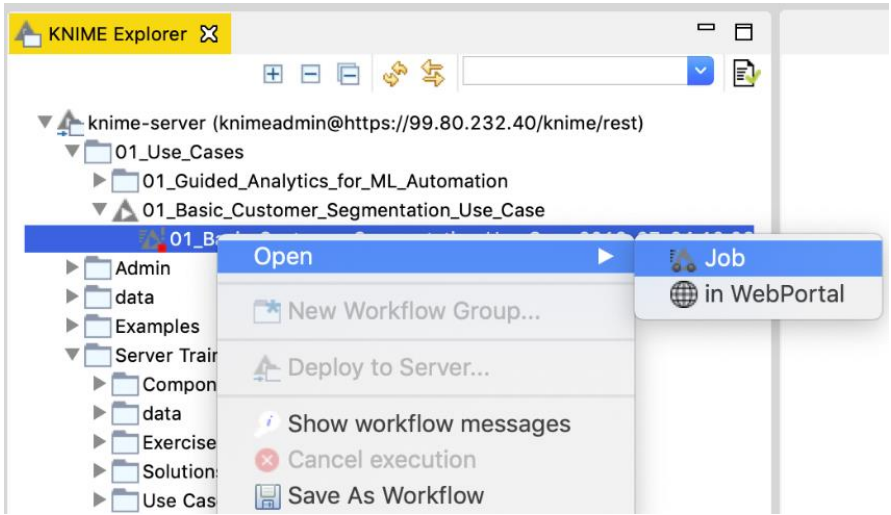
- A workflow job is a copy of the workflow with specific settings and data
- Jobs are tied to the version from when the job was created
- Orphaned jobs are colored red
- Jobs have messages (e.g. successful or failure)
- Jobs can be saved as a workflow for data provenance and debugging (right-click → Save as)

Remote Workflow Editor

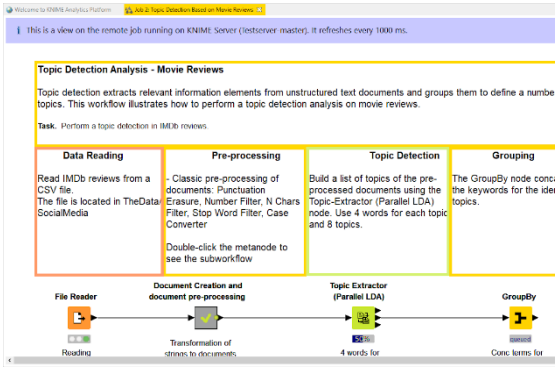


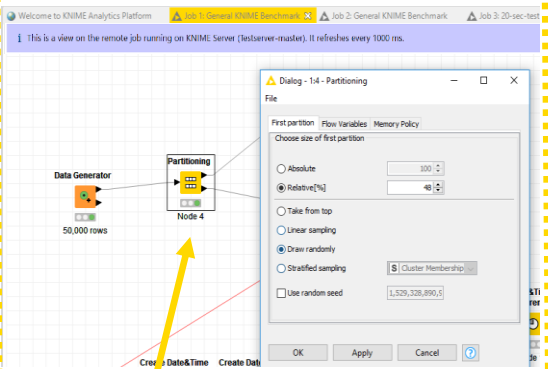
Remote Workflow Editor – 1/3

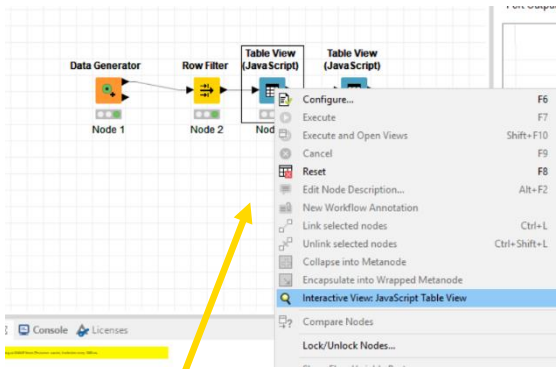
- Remote Control of Job running on KNIME Server
- Capabilities:
  - Live update of workflow job execution (executing node and progress)
  - Execute and cancel execution supported
  - Add/delete nodes
  - Change node settings
  - Inspect data tables / flow variables
  - JavaScript nodes can show data/views



Remote Workflow Editor – 2/3







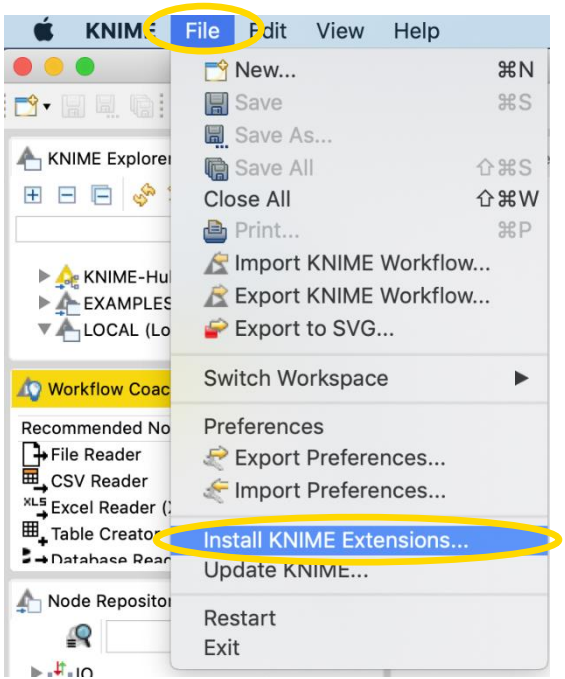
What's my workflow doing now?

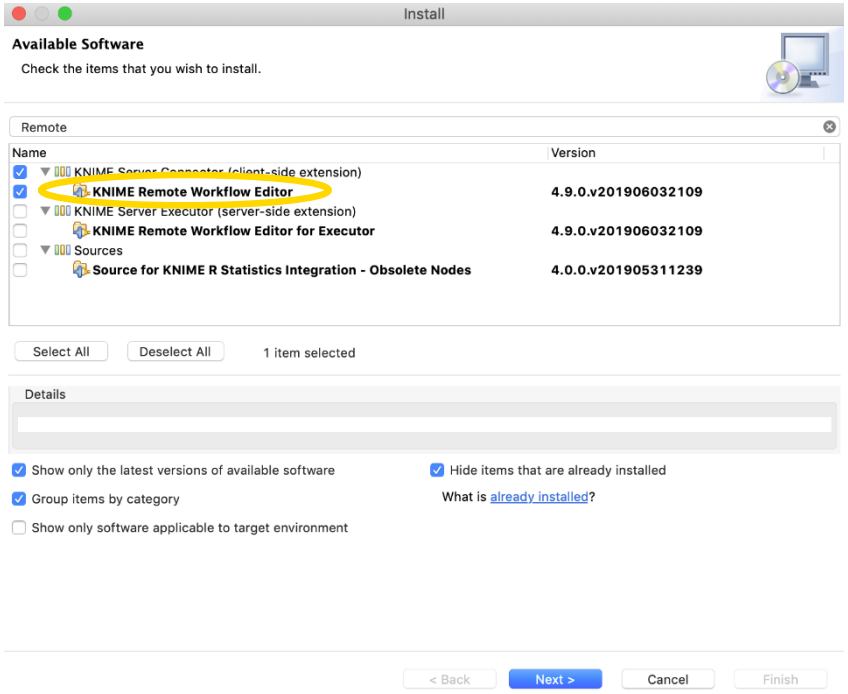
Change node configurations in the Job

JavaScript View support

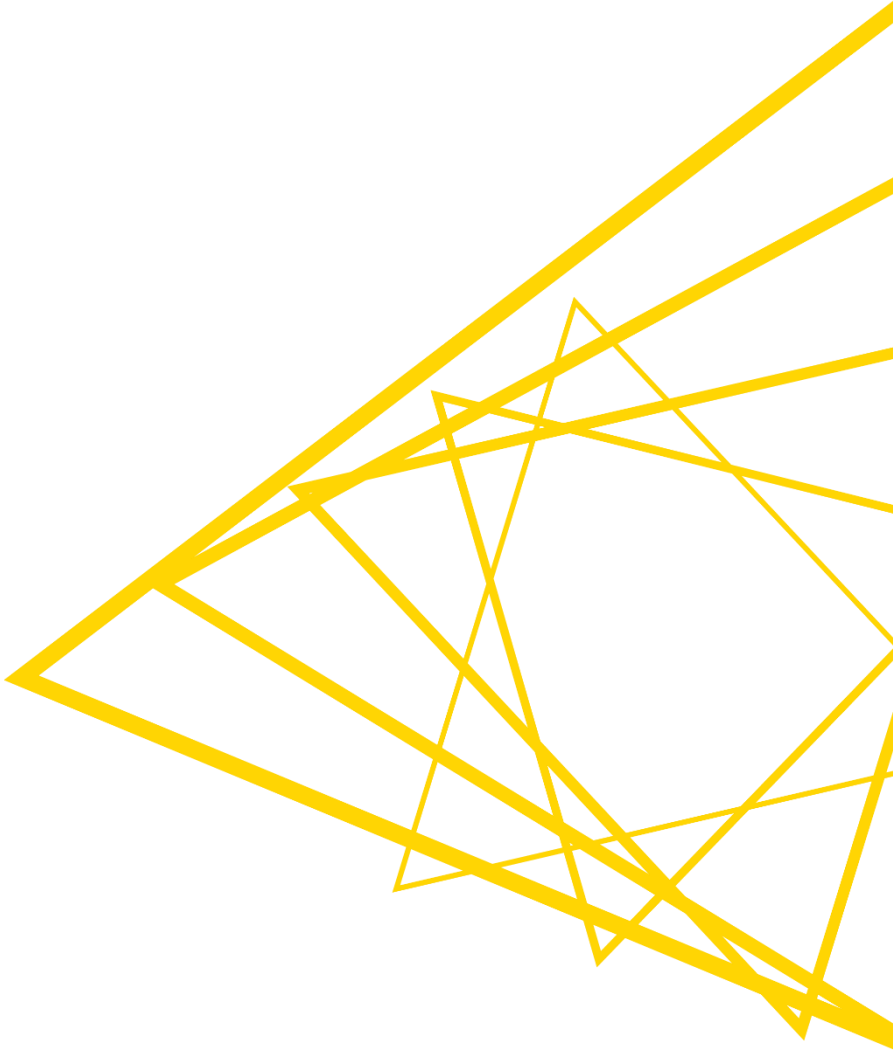
Remote Workflow Editor– 3/3

- To use the Remote Workflow Editor please install the KNIME extension Remote Workflow Editor





Permissions



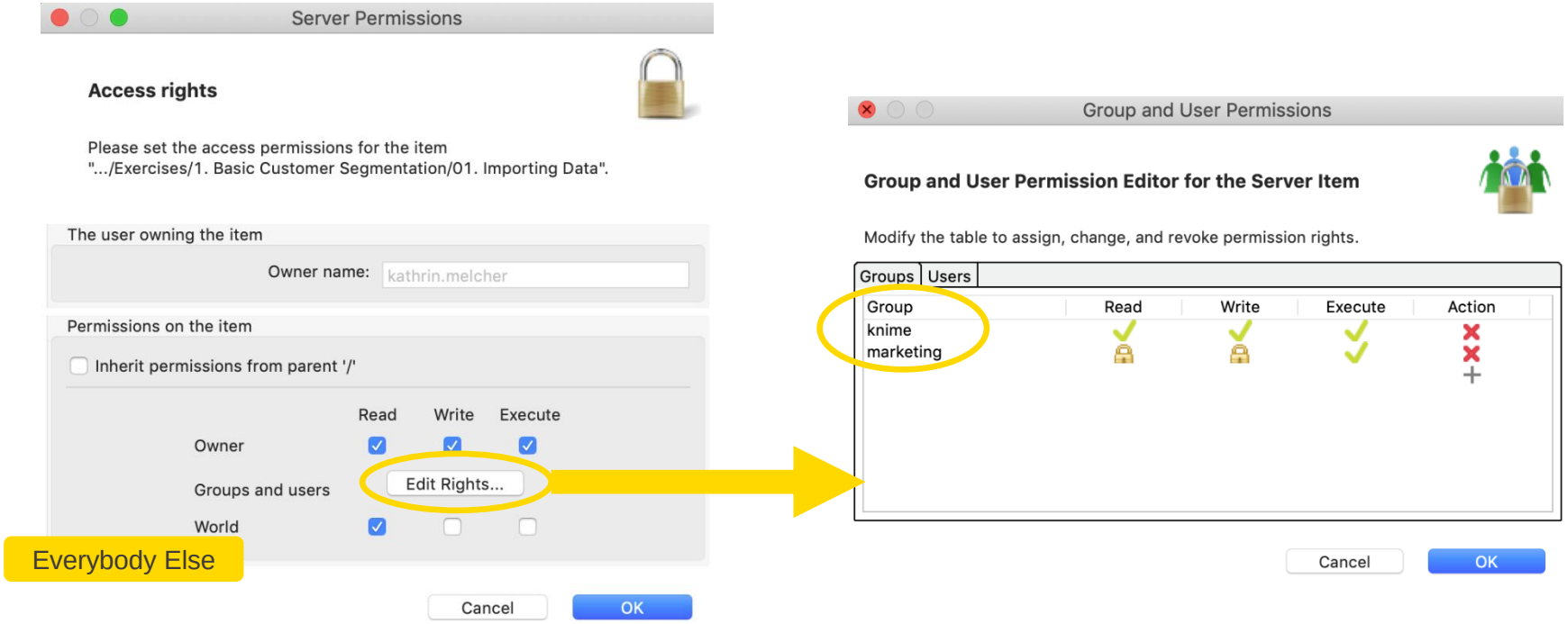
Permissions

- Permissions can be set for all types of items: workflows, workflow groups, components, and data files
- Permissions are assigned to either individual people or user groups
- The user who uploads an item, automatically becomes its owner
- Users with admin rights have no restrictions on permissions
- The owner, plus everyone with admin rights, can assign and change permissions for an item
- It is also possible to set permissions on schedules, such that a schedule can be maintained/changed by a team member while the owner is e.g. on vacation

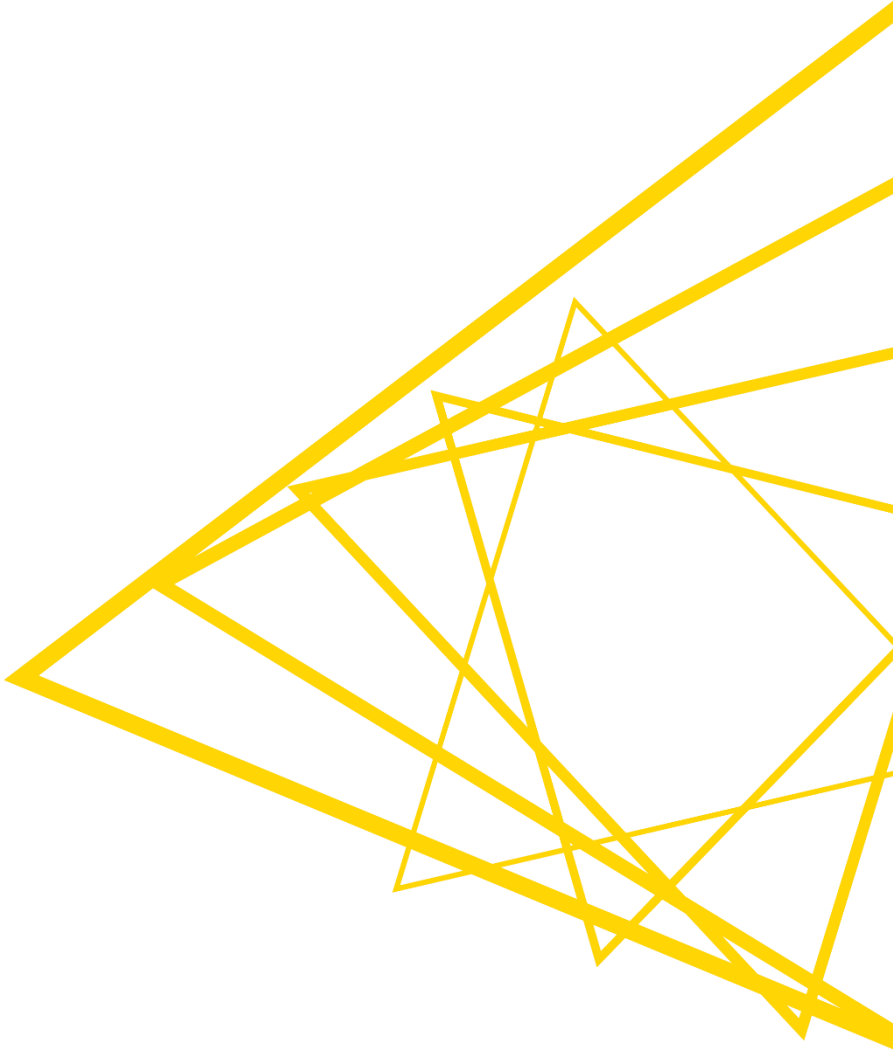
Permission Types

Type	Workflow	Workflow Groups	Files/Components
Read	Download a workflow job - including data	See the content of a workflow group	▪ File: download data and execute workflows that use the data ▪ Component: use and download
Write	Overwrite, create snapshots, and delete workflows	Create and upload new items in a workflow group	Overwrite a file or component
Execute	Execute a workflow by creating a workflow job		

Setting Permissions

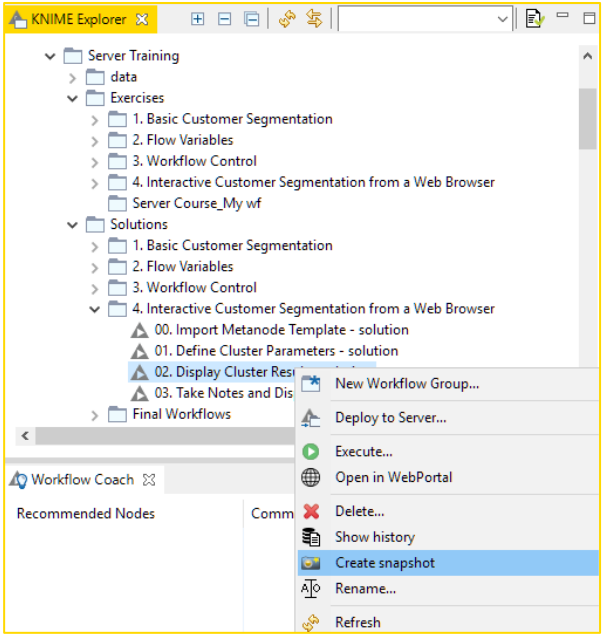


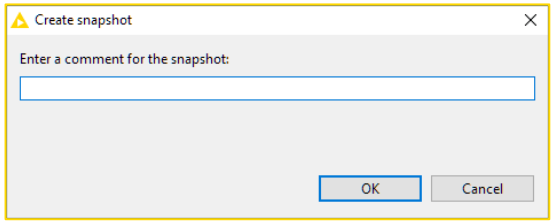
Versioning

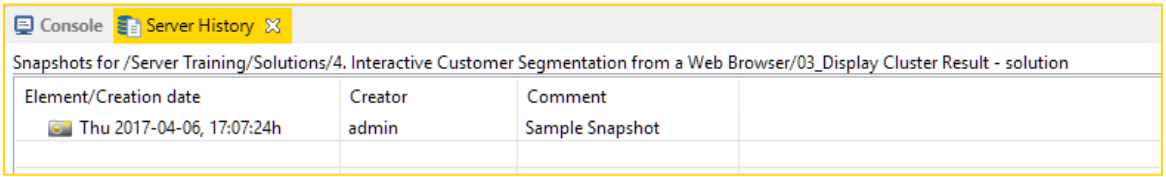


Versioning

- Possibility of creating a history of items on Server
- Create snapshots of workflows, data files, and components. These are stored with a timestamp and a comment

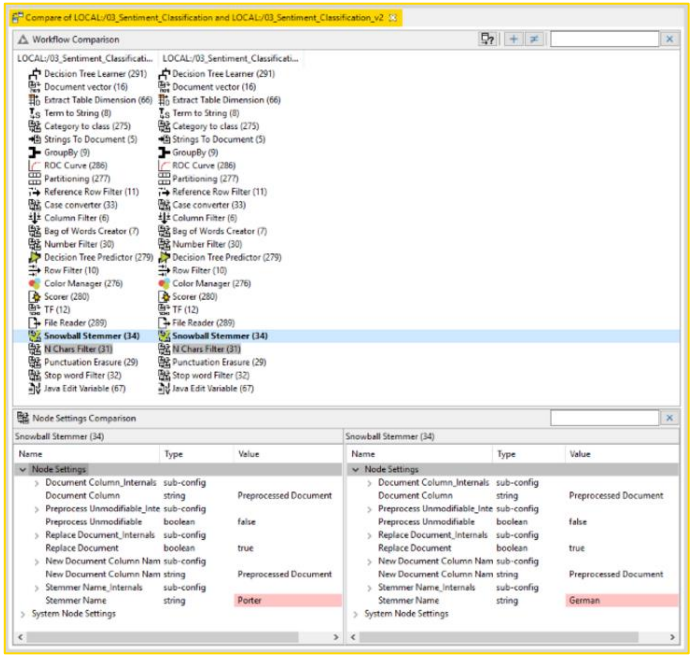


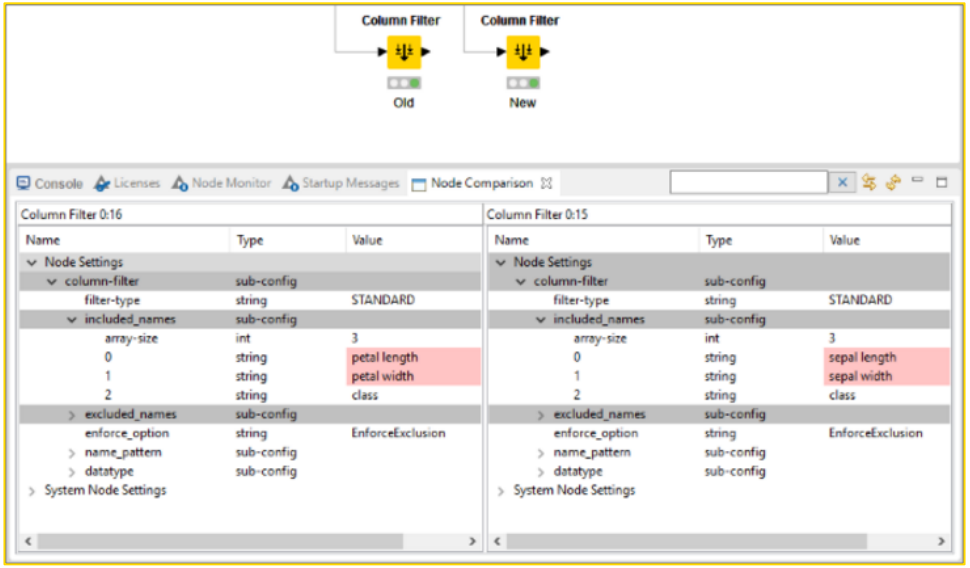




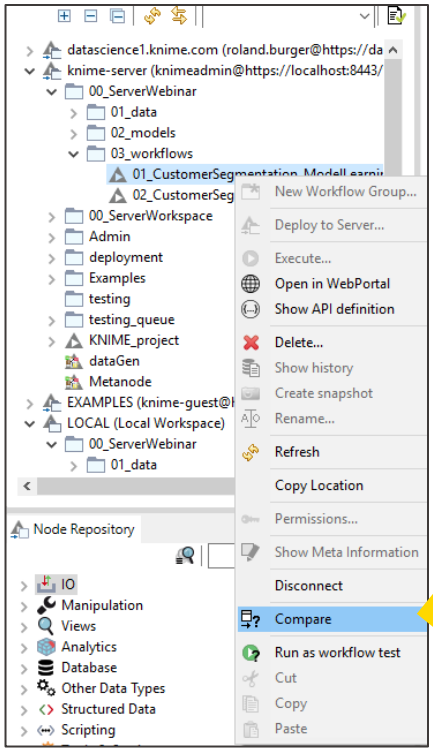
KNIME Workflow Difference (1/2)

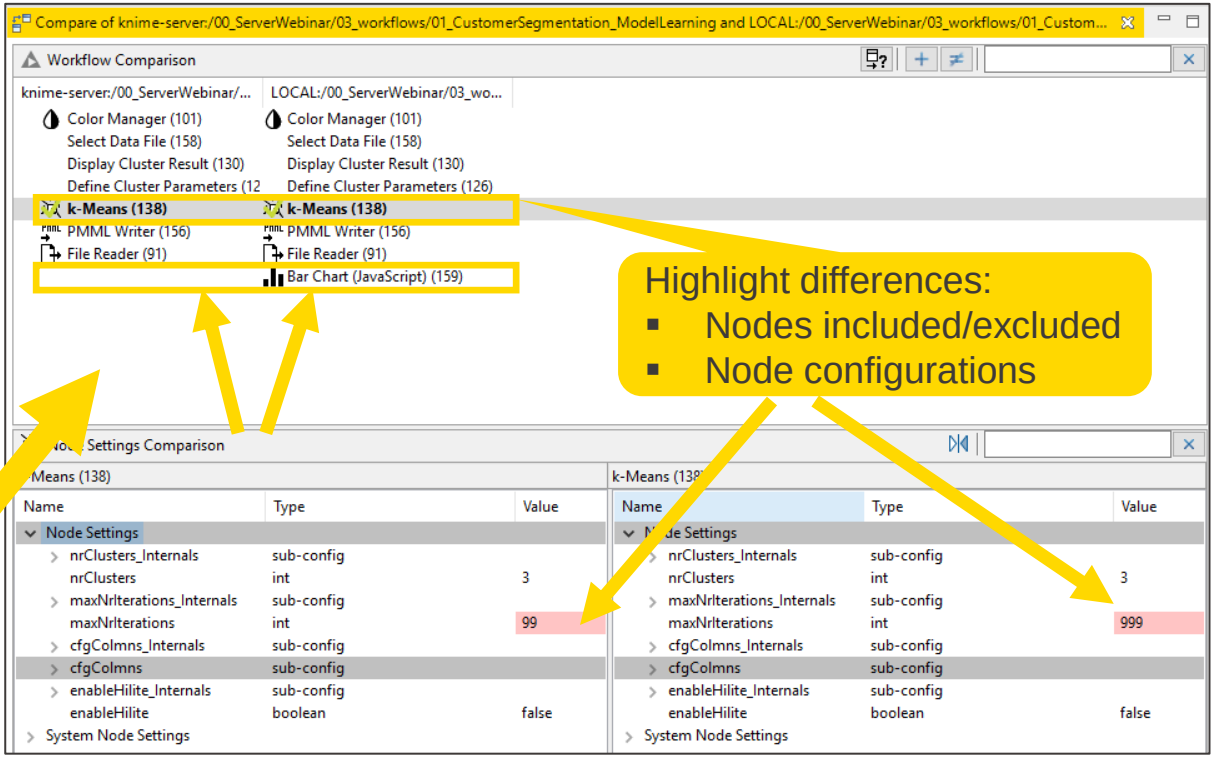
- Automates identification and comparison of nodes in a workflow, metanodes, and two different workflows
- Identifies insertions, deletions, substitutions, and parameter change



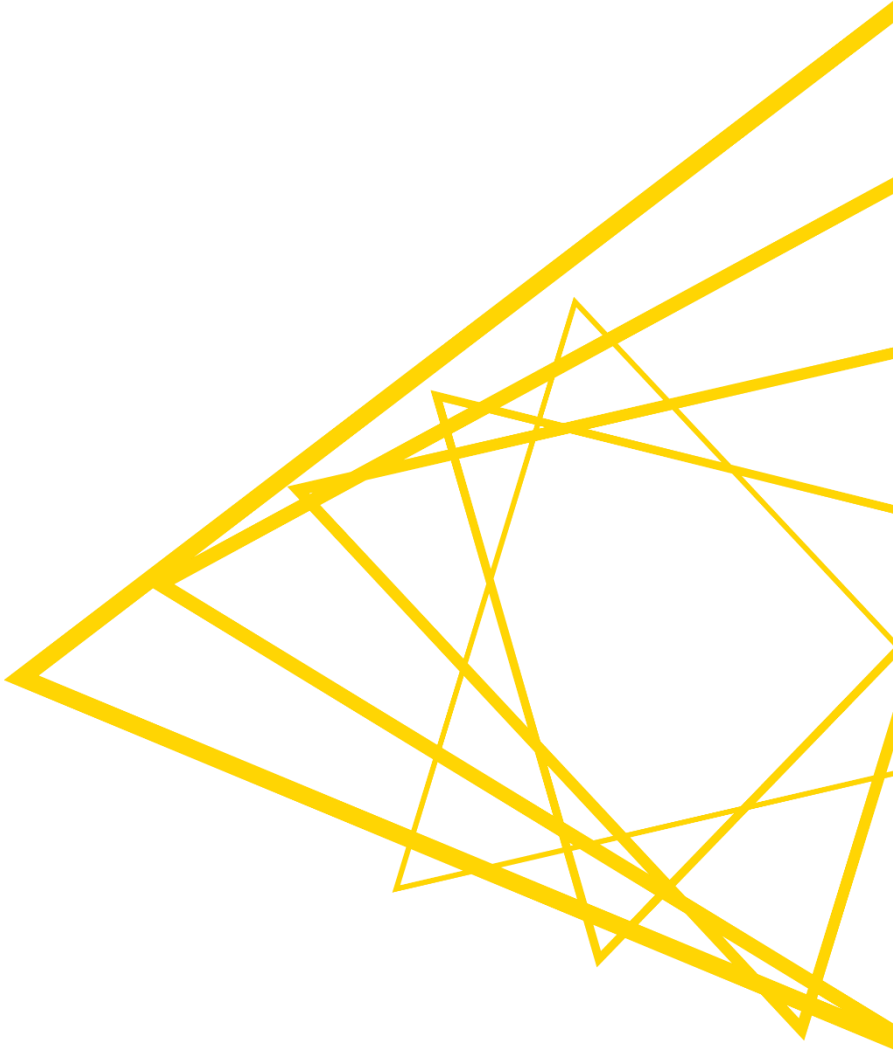


KNIME Workflow Difference (2/2)





Review and Exercises



Config Details – Access KNIME Server

- KNIME Server Address: http://3.123.122.13
- KNIME WebPortal Address: http://3.123.122.13/knime

Login credential: *firstname.lastname* (*)

Password: *knime*

*for double names and double surnames the whitespace has been removed

Working with KNIME Server, Activity 1

- Configure a **mount point** for **KNIME Server** with the details provided in the *Config Details – Access KNIME Server* slide at the end of the slide deck
- Download Server Training Material in your LOCAL workspace (hint: drag and drop or copy and paste the entire folder)

Working with KNIME Server, Activity 2

Path to the workflow:

Exercises → 01. Remote Workflow Execution

- Create a local copy of the workflow and rename it as *“01. Remote Workflow Execution_YourInitials”*
- Deploy the workflow on KNIME Server under the directory *Uploaded Workflows*
- Schedule a workflow execution (5 mins later)

Working with KNIME Server, Activity 3 (1/2)

- Make some changes to your local copy of the workflow (e.g. generate 5 clusters instead of 4) and afterwards deploy it to KNIME Server under the directory *Uploaded Workflows*
- Make sure to create a snapshot when updating your workflow on KNIME Server
  - Overwrite the existing item
  - Check the option Create Snapshot before Overwriting and provide a name to the snapshot
  - The snapshot is listed in the Server History view (View → Other → Server History)

Working with KNIME Server, Activity 3 (2/2)

- Download the snapshot to LOCAL Workspace
- Use the WorkflowDiff feature to compare the downloaded workflow with the latest workflow deployed on KNIME Server:
  - use *Ctrl* (or *Cmd*) button to select both workflows from KNIME Explorer;
  - right click on one of the two workflows
  - click *Compare*
  - select the two nodes that you wish to compare
  - *click Show configuration differences of highlighted nodes*

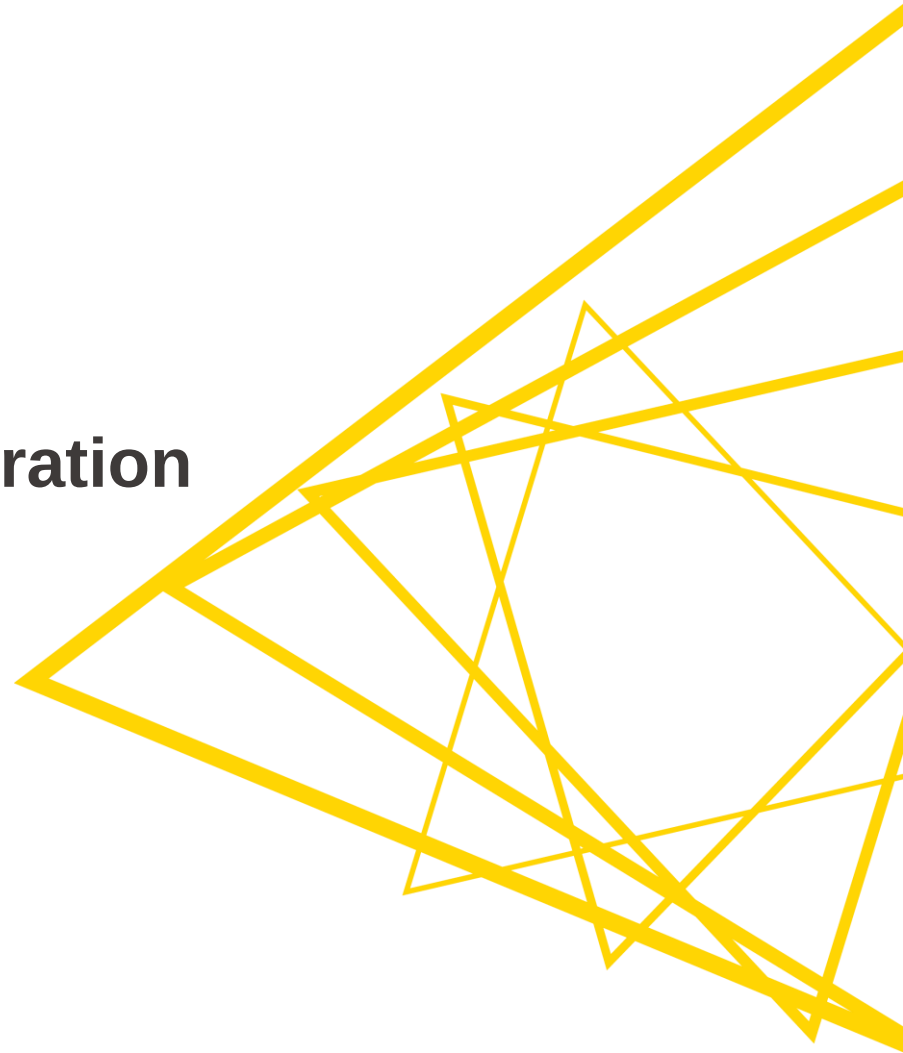
Open for Innovation
KNIME

Thank You!

Open for Innovation
KNIME

[L3-PC] KNIME Server: Productionizing and Collaboration

KNIME GmbH



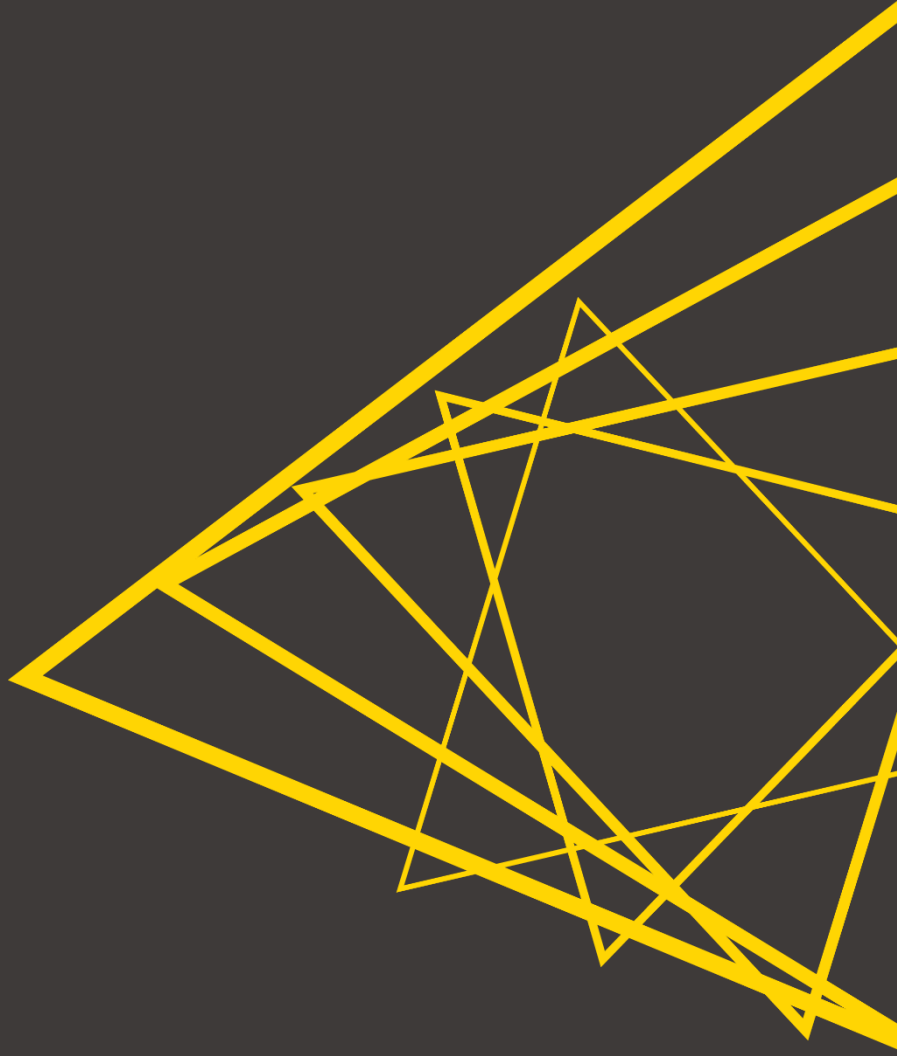
Structure of the Course

Session	Topic	Duration
Session 1	Introduction to KNIME Software & Working with KNIME Server	75 min
Session 2	**Creating Data Apps with Components**	75 min
Session 3	Deploying WebPortal Data Apps	75 min
Session 4	Integrated Deployment, REST API & Administration	75 min
Session 5	Wrap Up Session	15 min

Structure of each session

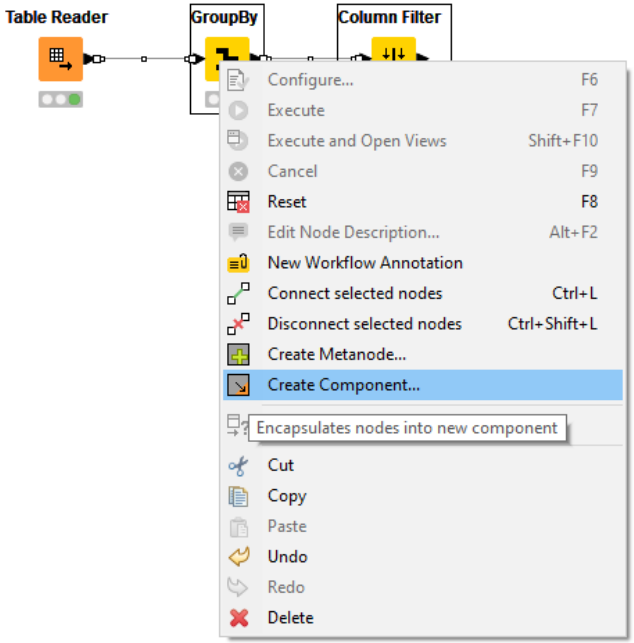
- Discussion of past exercises
- Course
- Introduction of next exercises

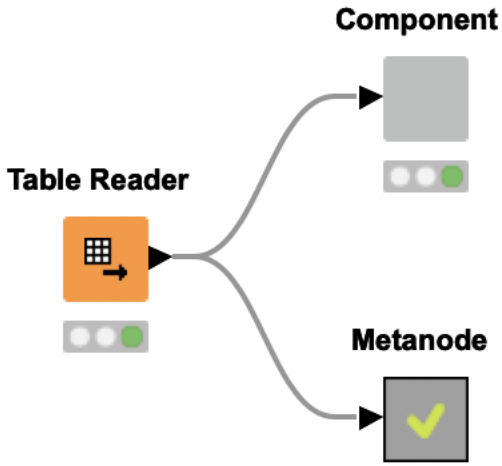
Components



What are they good for?

- Components encapsulate functionalities that can be reused as your personal customized KNIME nodes, to perform tasks that you often repeat.
- They can also be shared with others via KNIME Hub and KNIME Server.

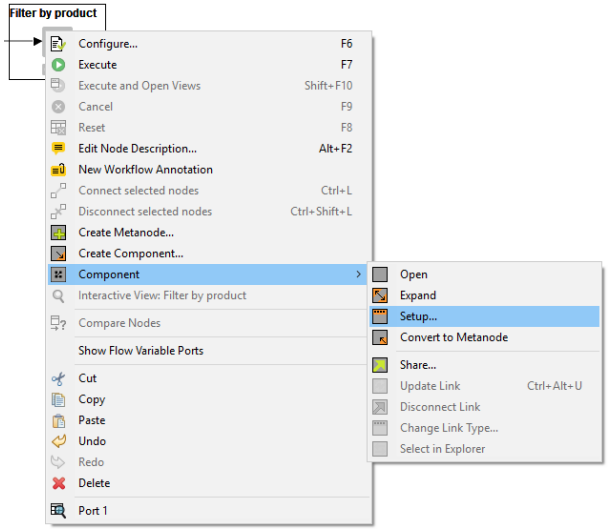




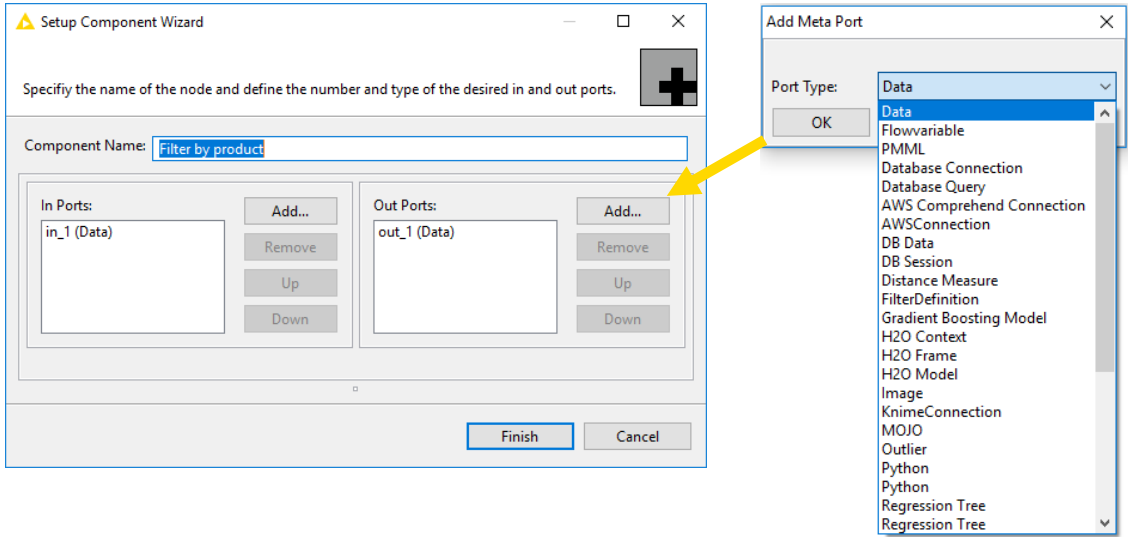
Metanodes vs. Components

	Metanodes	Components
Configuration	Not configurable	Via Configuration nodes (local workflow) and Widget nodes (KNIME WebPortal)
Variable scope	Global	Configurable: Local or global
WebPortal Usage	Executed in the background	JavaScript Views and Widgets inside the component are shown on a WebPortal page
Execution mode	Normal execution	Allows Simple Streaming execution
Recommended uses	Workflow cleaning	Enabling custom interactions, producing interactive views, sharing functionalities

Component Setup

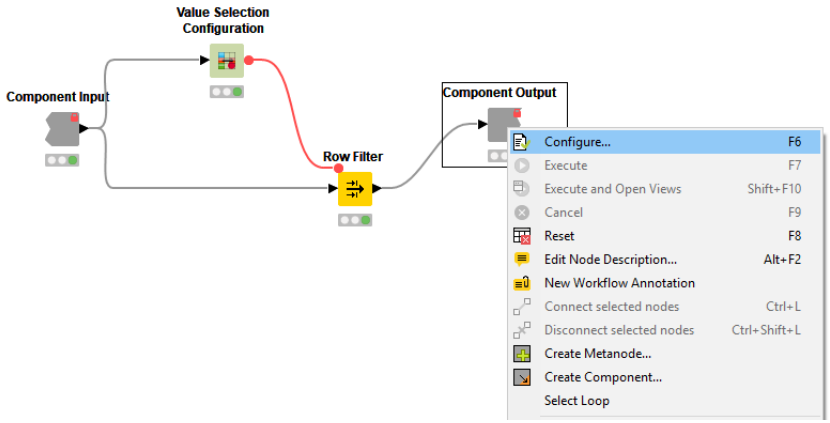


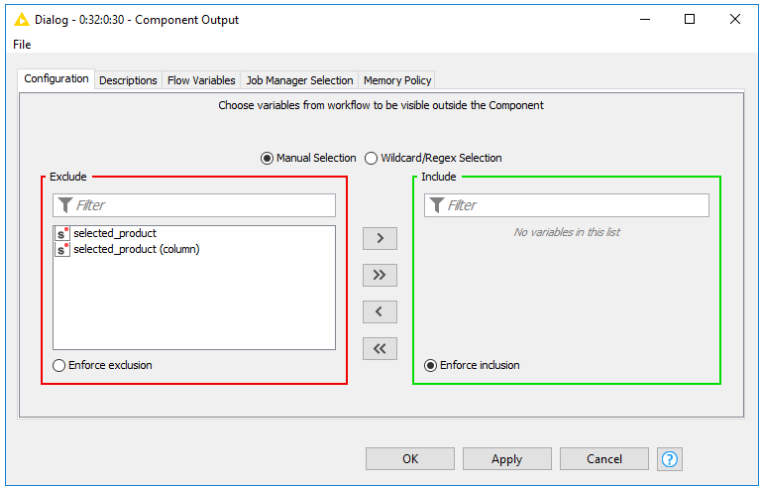
- Add input and output ports to components
- Remove ports to adapt to changes after creation of the component



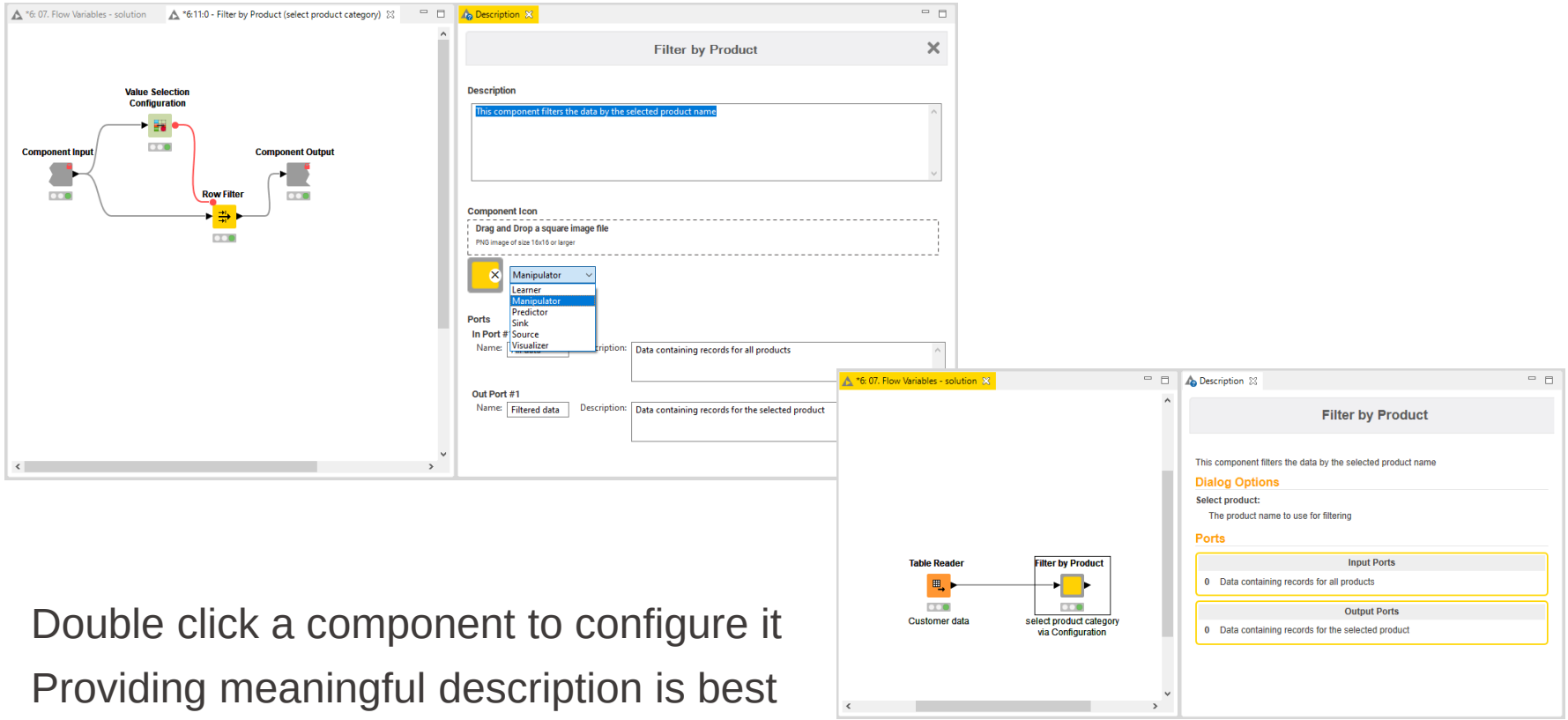
Flow Variable Scope of Components

- Flow Variables are -by default - only available locally inside the component
- Configure the component input/output to pass Flow Variables from/to outside the component



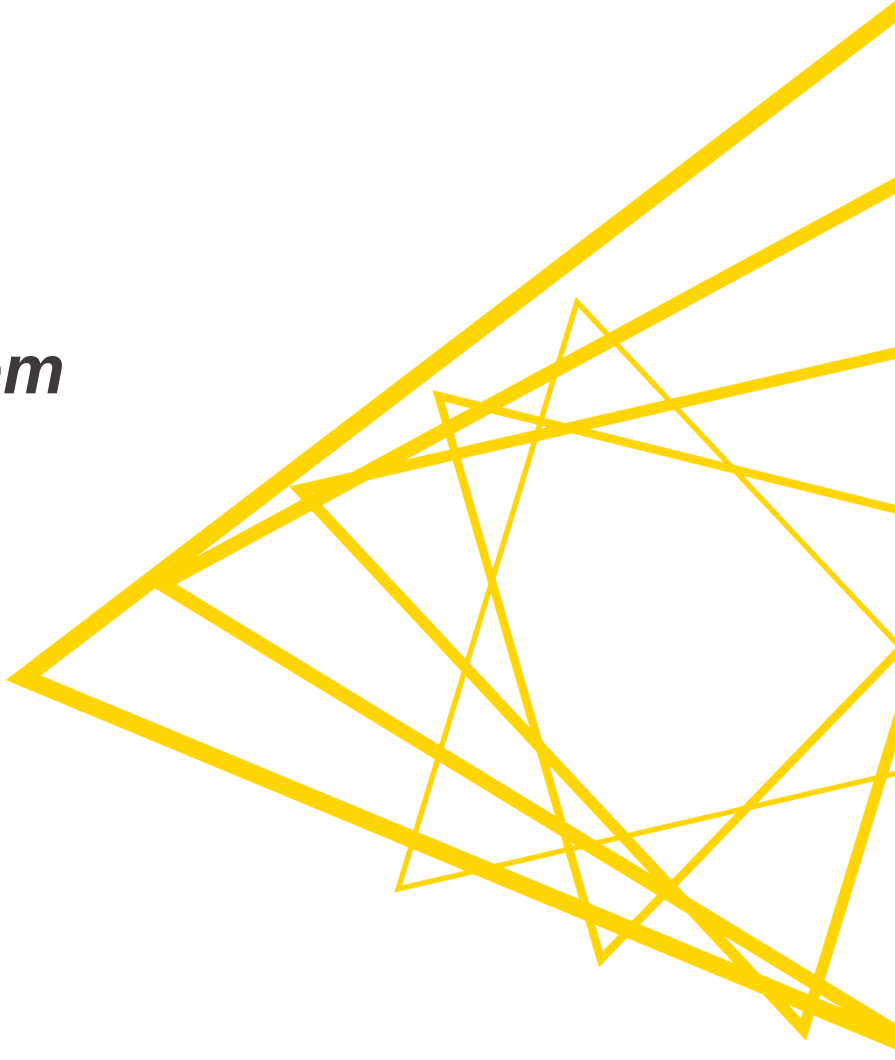


Component Description



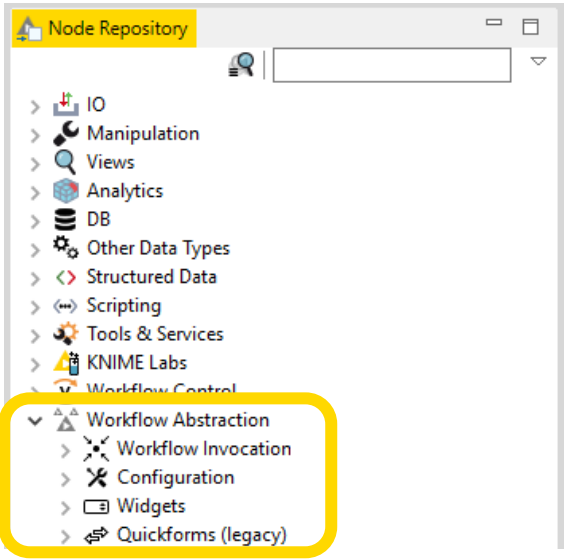
- Double click a component to configure it
- Providing meaningful description is best practice

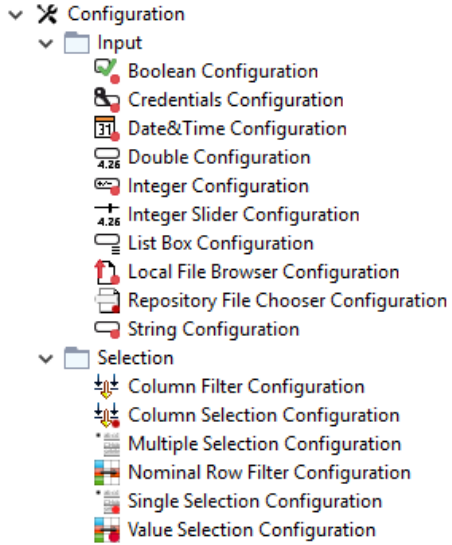
Components: *Creating custom nodes from KNIME nodes*



Creating a Configuration Dialog

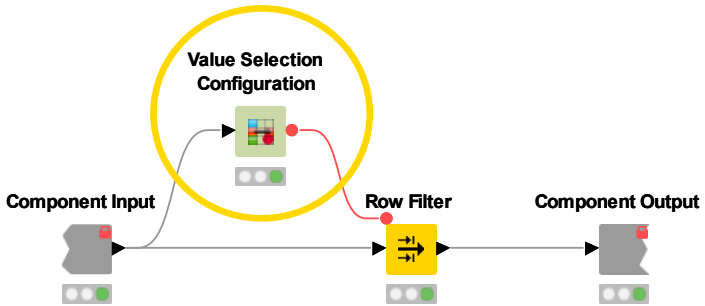
- Configuration nodes allow creation of a Component's configuration dialog
- Configuration nodes enable different types of user inputs such as string input, integer input, selecting one value from a list and [many more]

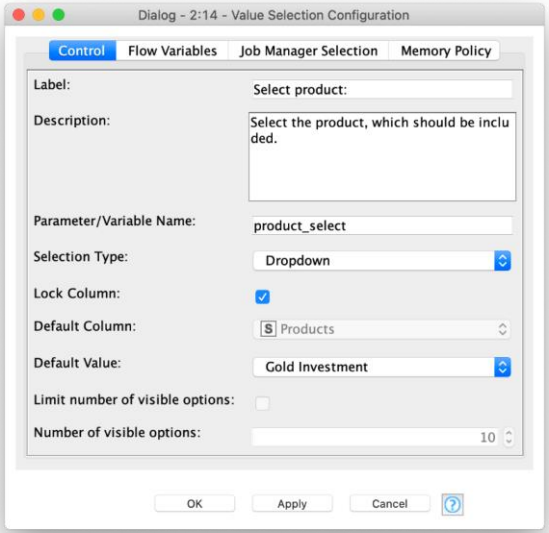




Creating a Configuration Dialog

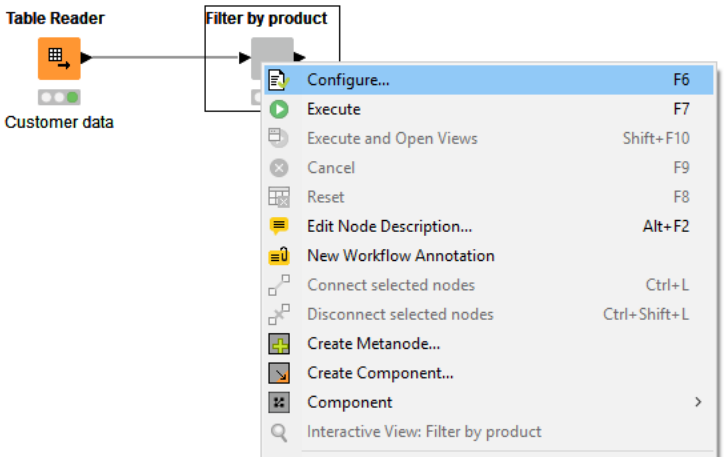
- Use Configuration nodes to create Flow Variables
- Every Configuration node has a label, description and parameter/variable name
- Depending on the node, additional options and visual properties can be set
- Flow variables created in the Configuration nodes can then be used to overwrite the settings of subsequent nodes

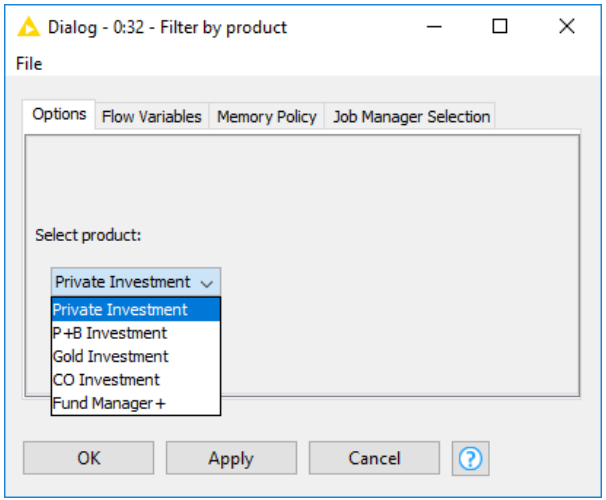




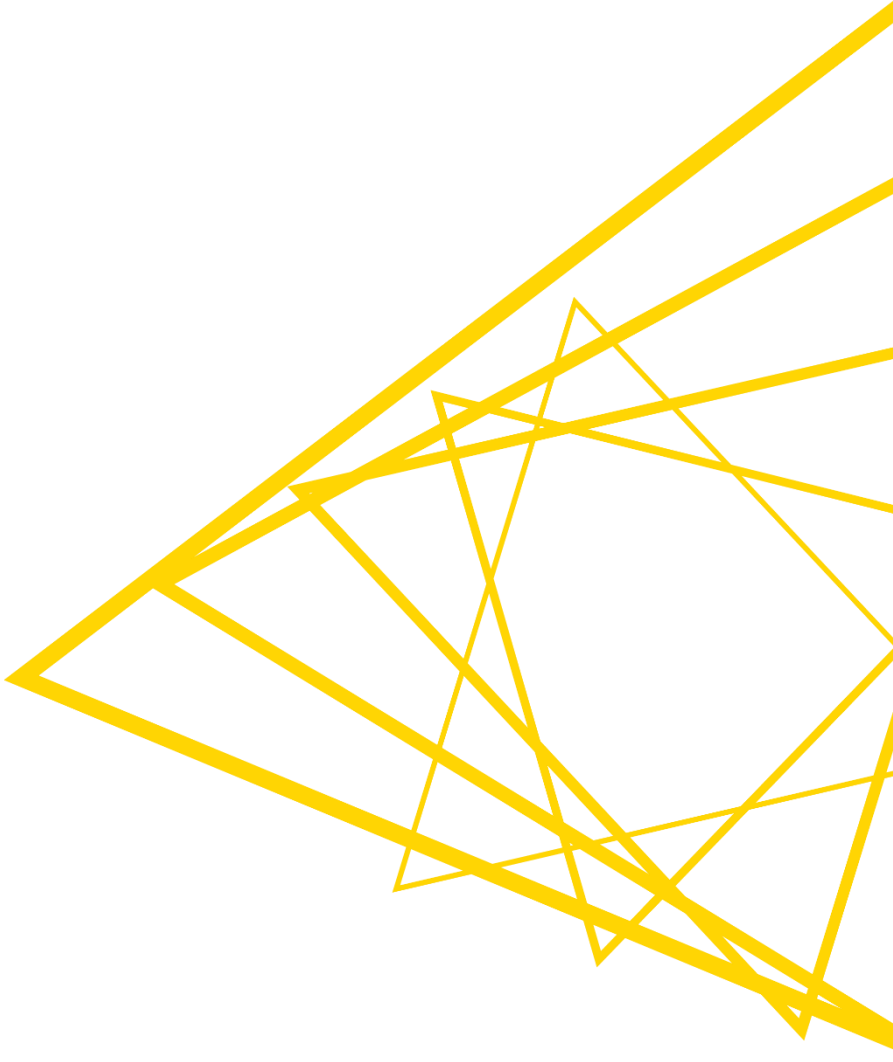
Configuration of a Component

- Double click a component to configure it



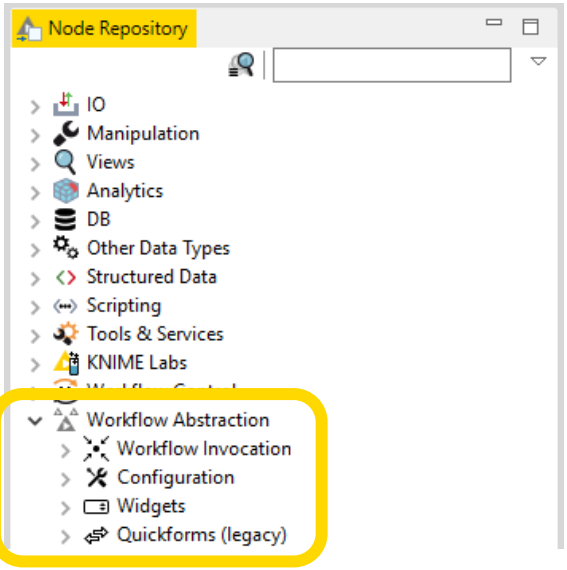


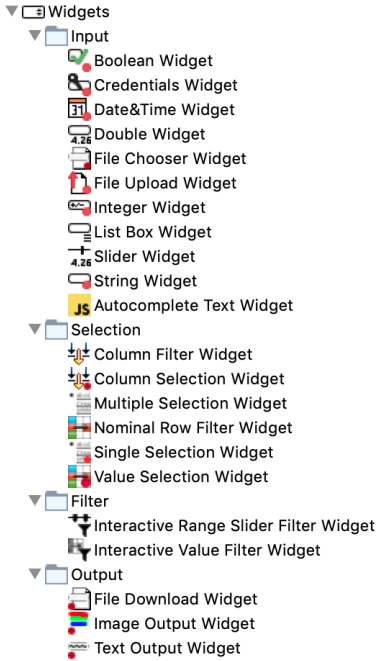
Component Composite View



Creating a Composite View

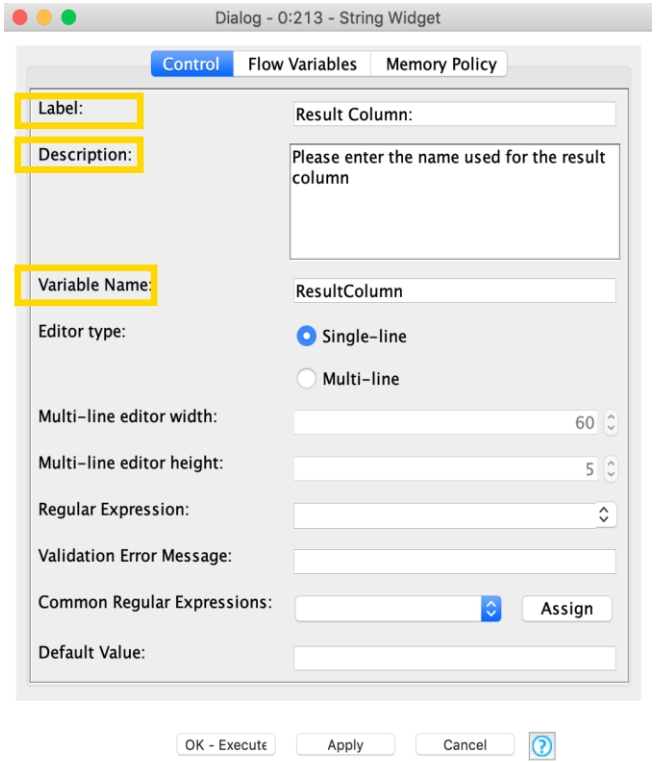
- JavaScript visualization and Widget nodes allow creation of a composite view
- Widget nodes allow user interaction in a composite view, e.g.
  - Adjustment of parameters via various input nodes
  - Selection of predefined values
  - Consumption of outputs

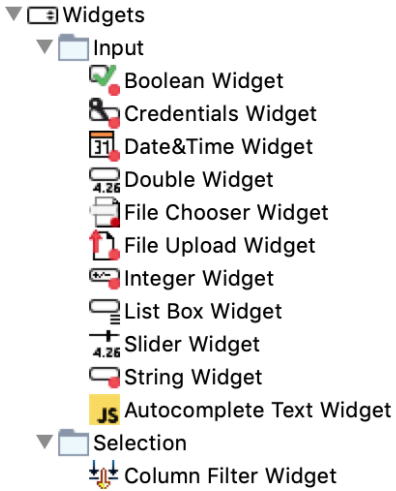




Configuration of Widget Nodes – Input Nodes

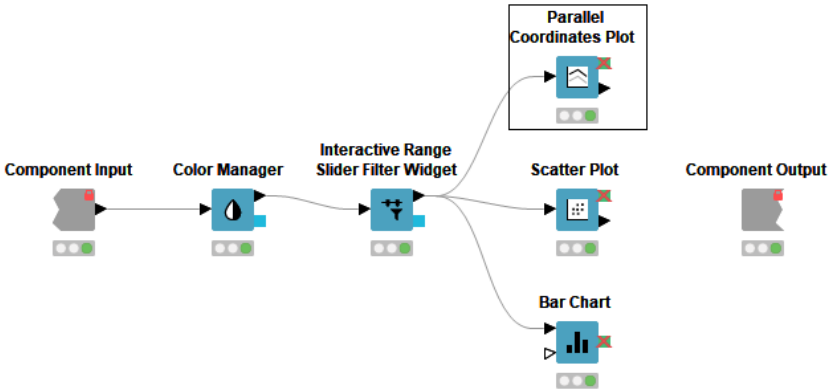
- Widget input nodes have very similar configuration dialogs

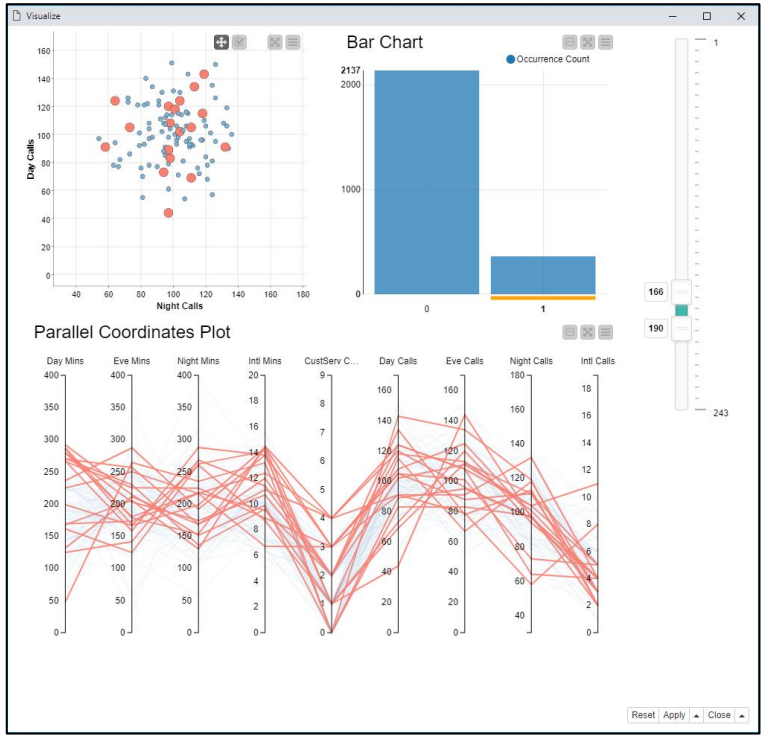




Component Composite View

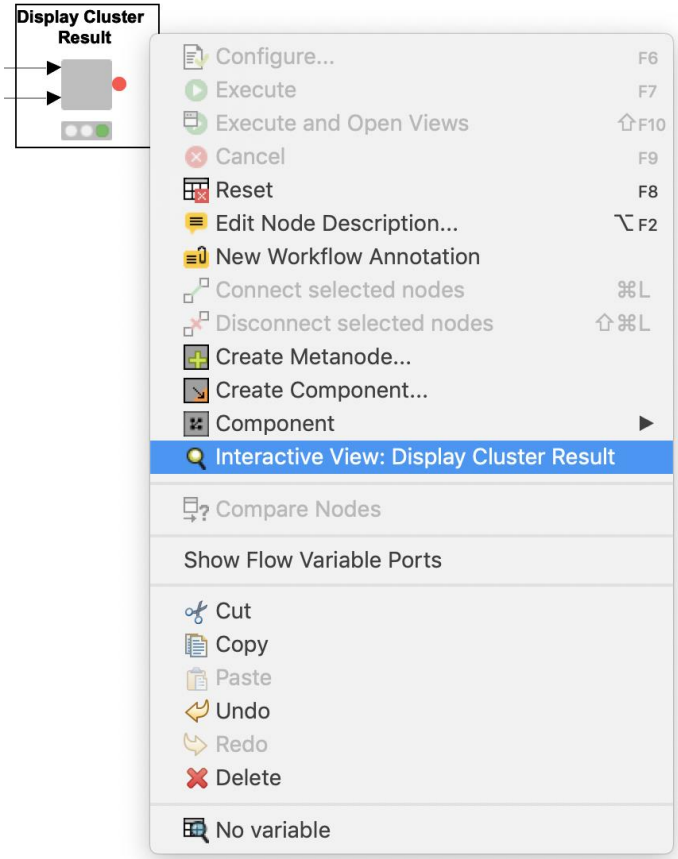
- Multiple JavaScript View nodes can be combined in Components
- Selections are transmitted to all other views
- User can give input via Widget nodes

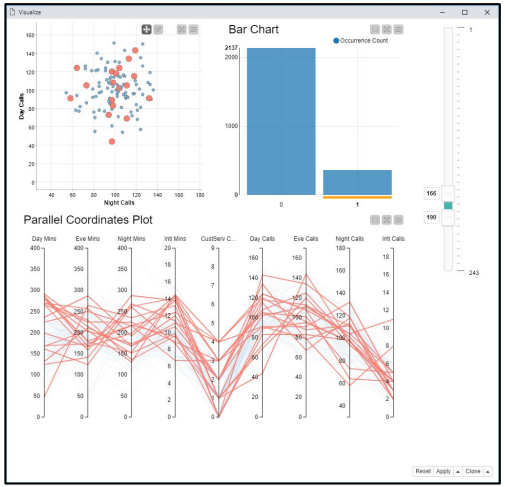




Component Composite View

- You can open a composite view directly from the node's context menu.
- It will open a browser window that displays all containing JavaScript and Widget nodes.
- The same view will also be shown via KNIME WebPortal





Best Practice

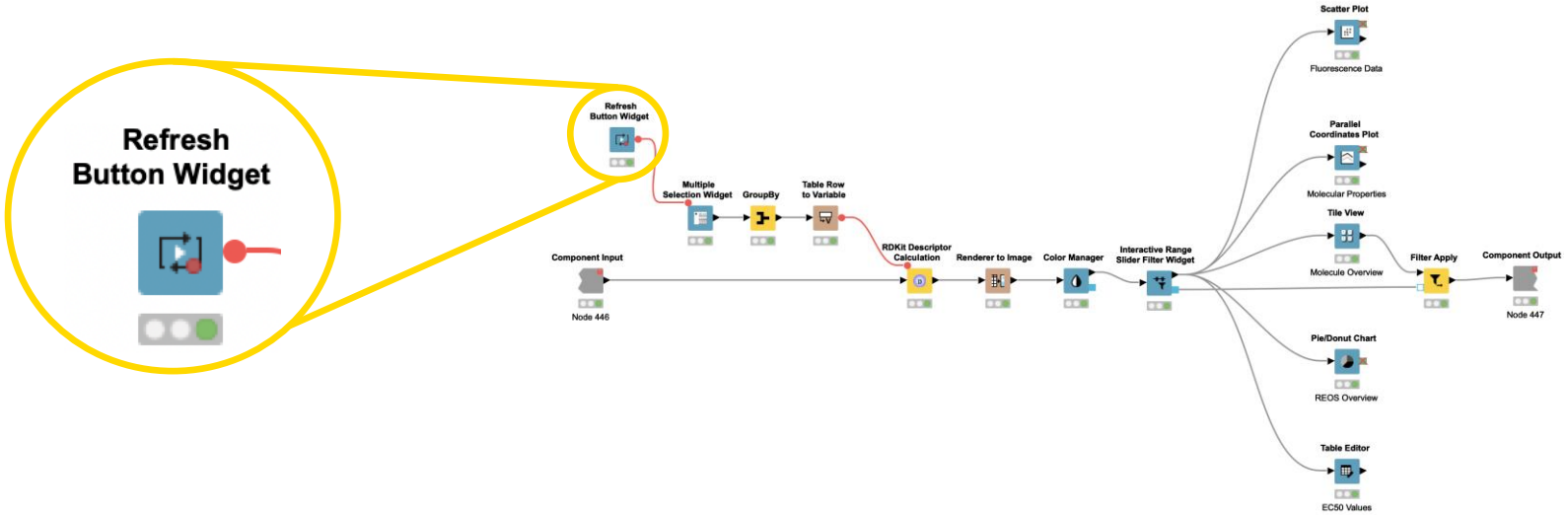
- Widget nodes for users input/interaction need to be separated from visualization nodes (JS nodes)

- IMPORTANT:
  - First component with widget nodes for user interaction/input
  - Second component with visualization of data based on previous user inputs

NEW: Use Refresh Button to combine Widget nodes and JS nodes

Refresh Button

- Adds a button widget with configurable text which can be used to refresh the composite view of the component.
- To use the node, connect the variable output port to the nodes which should be re-executed. The downstream nodes of those connected nodes will also be re-executed when the widget is clicked.

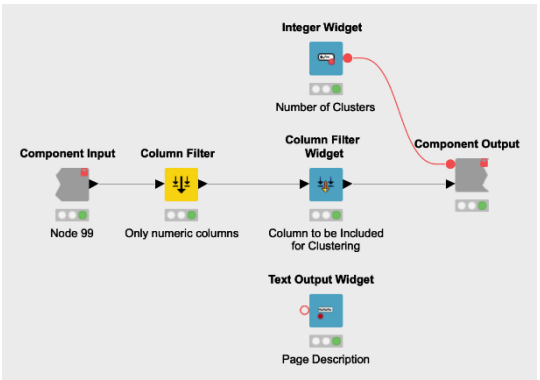


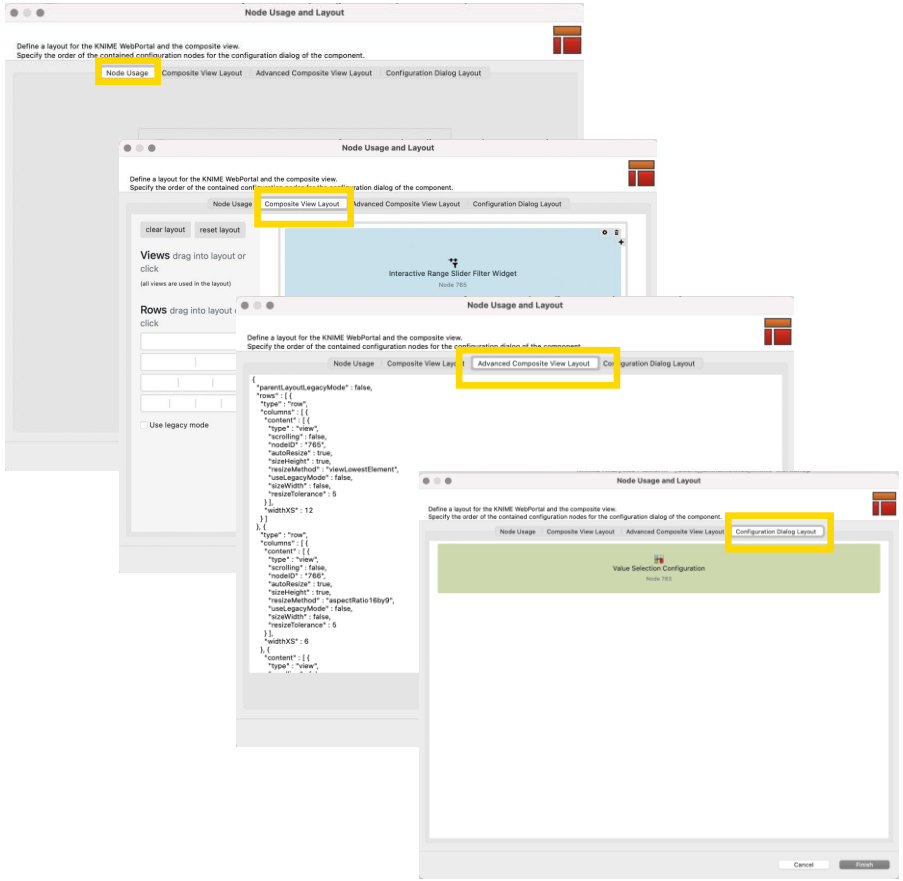
Layouting

- A layout can be defined for any Component that contains at least one widget node or JavaScript-enabled view
- The layout editor can be accessed from the top toolbar, when inside the component

Defining the Layout

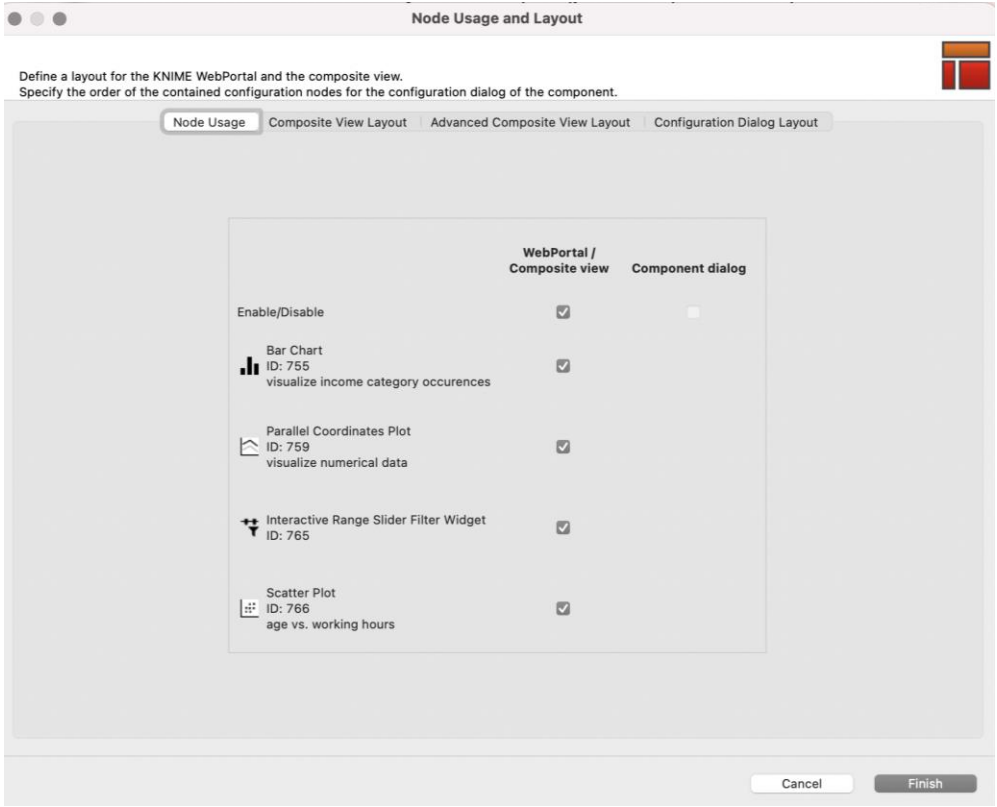
- Four tabs available:
  - Node Usage
  - Composite View Layout
  - Advanced Composite View Layout
  - Configuration Dialog Layout





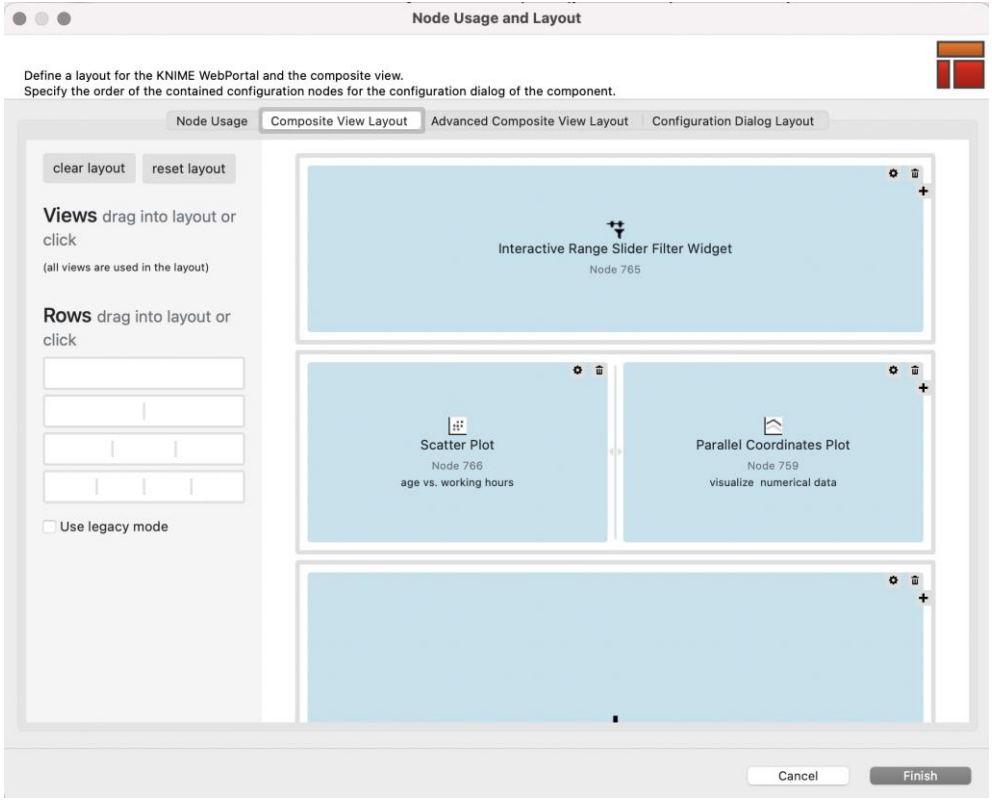
Layouting I – Node Usage

- Enabling the view of Widget nodes on the WebPortal or on the Component view
- Enabling the view of the input given by the Configuration node on the Component Dialog



Layouting II – Visual Layout

- Makes it easier to specify how individual JavaScript views are laid out in a WebPortal page or Component composite view



Layouting – Order Items to Show on the WebPortal

- The “Append the IDs to node names” button on the top bar shows the ID of each node

- This is useful to reorder the items in the layout structure for the WebPortal

Layouting – Nested Components

- It is possible to include components inside other components
- It allows to build up a library of components that contain useful linked views and then easily assemble them to create complex views for visualizing and interacting with datasets

Use Case: Creating a Data App for Molecule Exploration

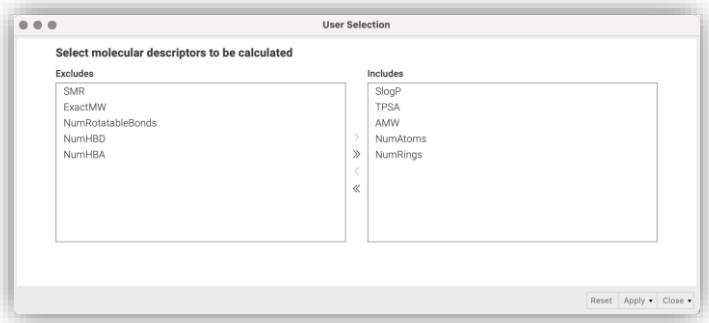
Introduction to Use Case

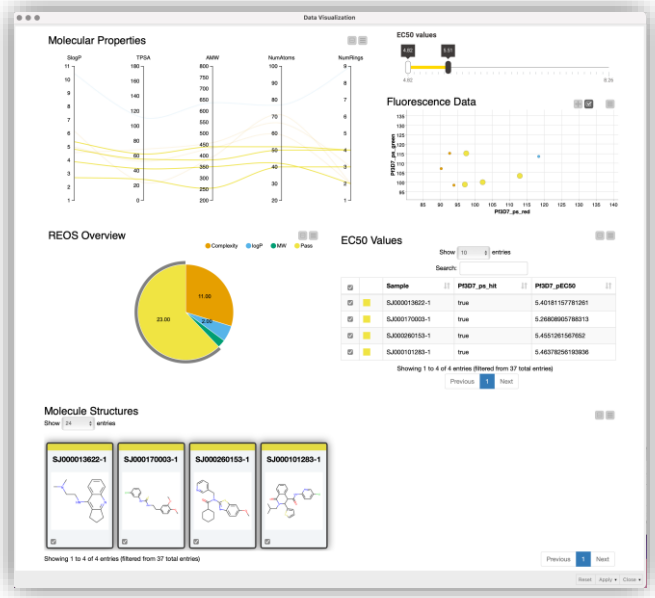
- Drug discovery: compare molecule efficacy and molecular properties (e.g. solubility, permeability, metabolic stability,…) to find best molecule for further drug development

- Aim to create an app, where
  - Users can upload their file
  - Define, which molecular properties should be calculated
  - Explore their data using interactive visualizations and select the best molecules
  - Download an Excel file with the selected molecules

Exercise 02. Data app - 2 Views

- Open Exercise 02. Data app - 2 Views:
  - Create a data app with two view
    - first view: selection of properties to calculate
    - Second view: data visualization to explore the molecules and select the best 5 molecules





Exercise 03. Data app – Single View

- Open Exercise 03. Data app - Single Views:
  - Create a data app with a single view and a Refresh Button
    - selection of properties to calculate and data visualization to explore the molecules and select the best 5 molecules

Config Details – Access KNIME Server

- KNIME Server Address: http://3.123.122.13
- KNIME WebPortal Address: http://3.123.122.13/knime

Login credential: *firstname.lastname* (*)

Password: *knime*

*for double names and double surnames the whitespace has been removed

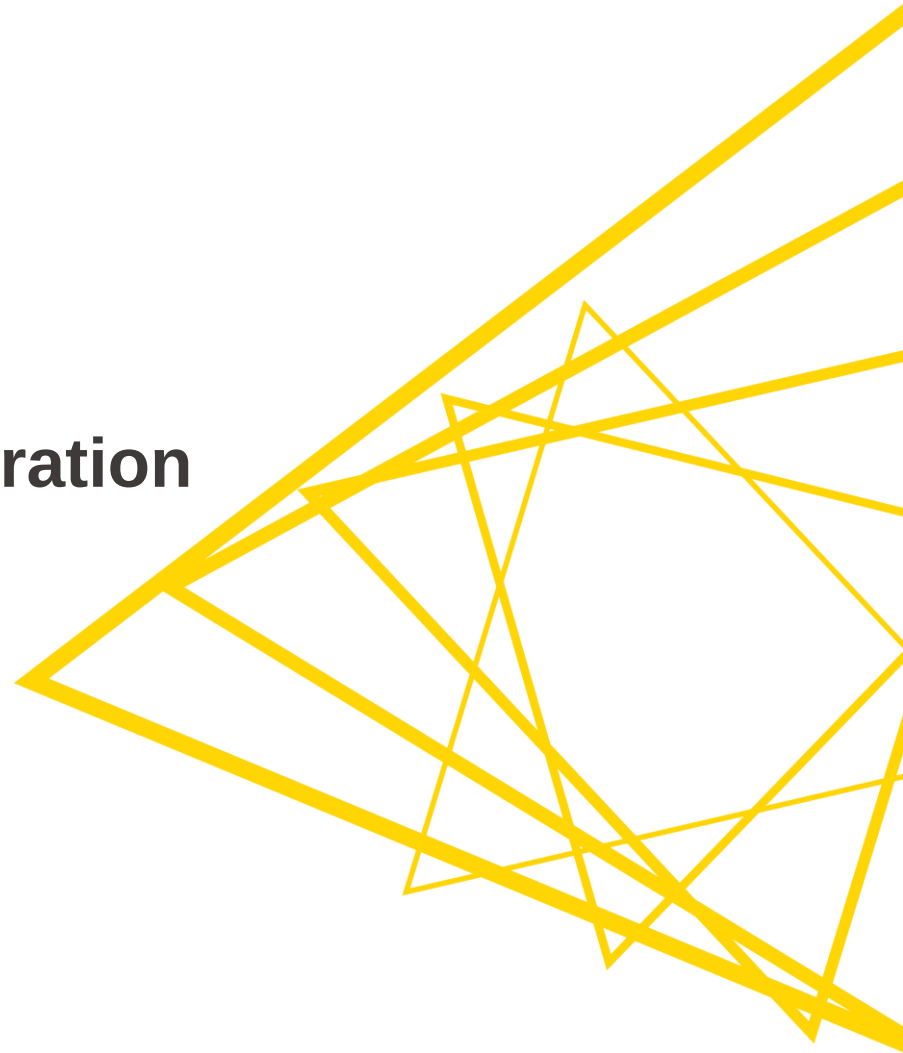
Open for Innovation
KNIME

Thank You!

Open for Innovation
KNIME

[L3-PC] KNIME Server: Productionizing and Collaboration

KNIME GmbH



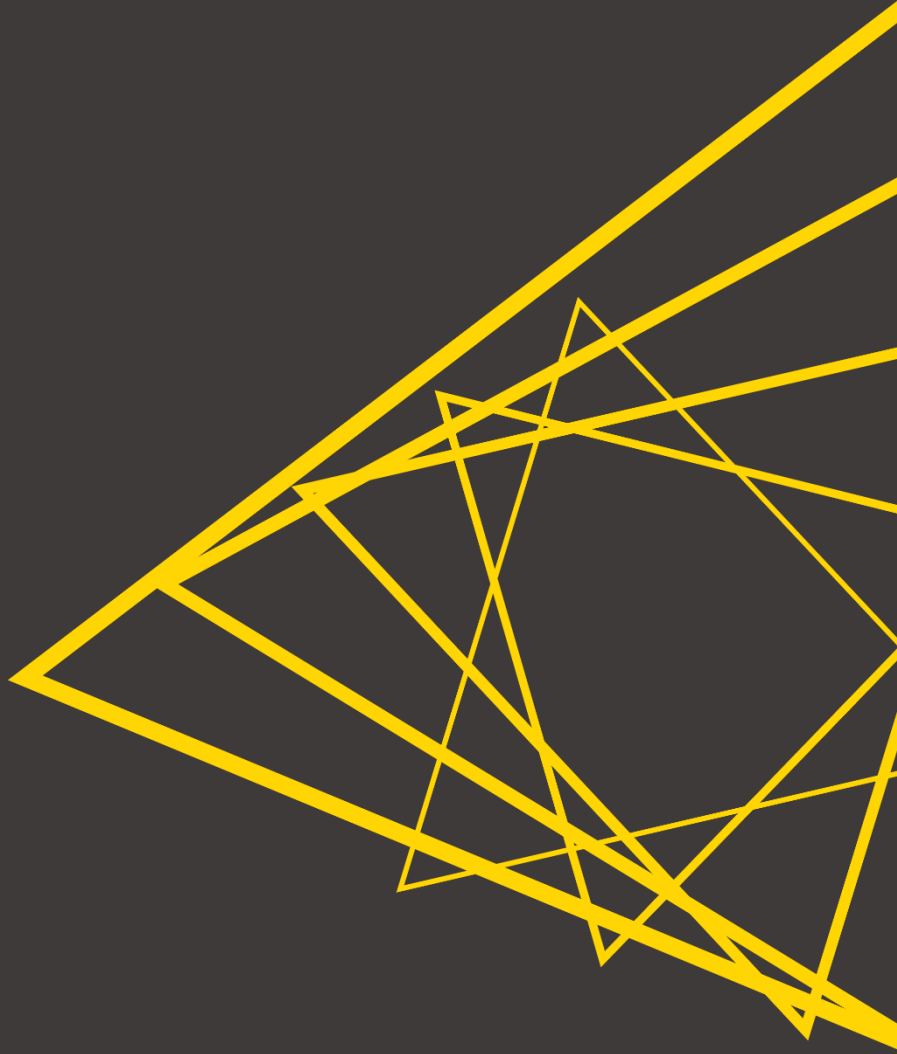
Structure of the Course

Session	Topic	Duration
Session 1	Introduction to KNIME Software & Working with KNIME Server	75 min
Session 2	Creating Data Apps with Components	75 min
Session 3	**Deploying WebPortal Data Apps**	75 min
Session 4	Integrated Deployment, REST API & Administration	75 min
Session 5	Wrap Up Session	15 min

Structure of each session

- Discussion of past exercises
- Course
- Introduction of next exercises

WebPortal Data Apps



Data Science Practice: Multiple Stakeholders' Needs

Data Engineers
Data Science "coders" (Python, etc.)
Data Science Specialists
Data Science Visual workflow / generalists

Smart Business Users (more than Excel)
Application Users – Interaction required
Application Users – Made to spec
Report Consumers

Model / ML Operations
Feeding production systems

Operations Consumption
(Applications, Systems, Edge, etc.)

IT Operations
- Centralized resources / strategies
- Standards and preferred platforms used, infrastructure options
- Exit strategies

IT Security
- Data, applications

Financial / Risk Oversite
- Costs allocation

Compliance officer
Data/model Governance, traceability, GDPR

Composite Views: KNIME AP vs KNIME WebPortal

KNIME Analytics Platform (AP)

Executed locally
Visualized in KNIME AP

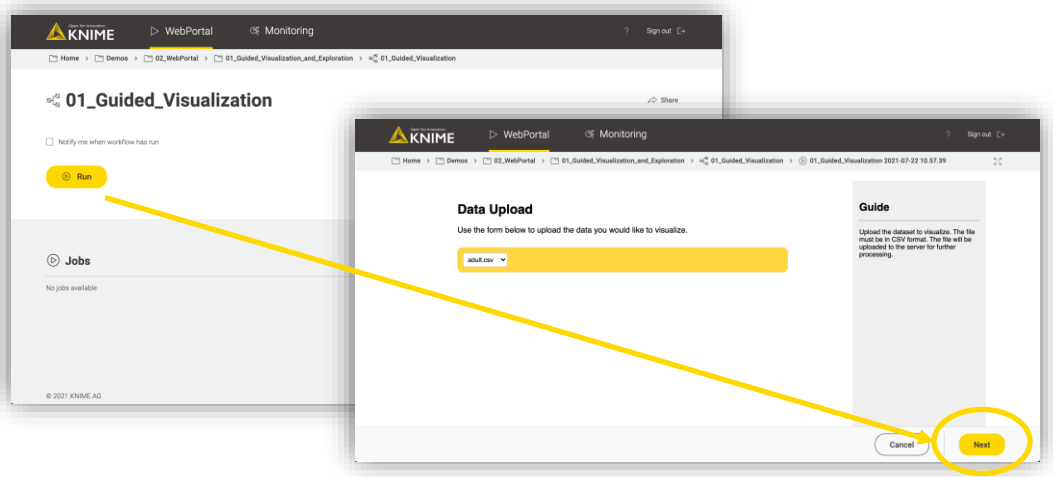
KNIME WebPortal

Executed on KNIME Server
Visualized in Browser as web app

Open for Innovation
KNIME

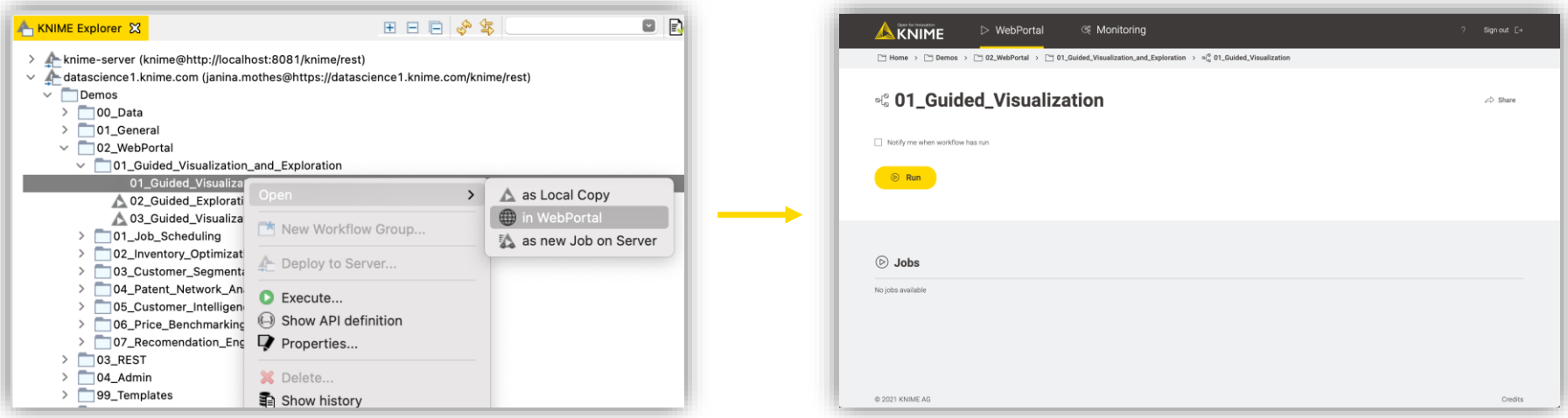
KNIME WebPortal

- All workflows on KNIME Server available as web apps
- Step-by-step execution of workflows from any browser
- Simple, clean interface for end users: Guided Analytics
- Customize layout to match corporate design



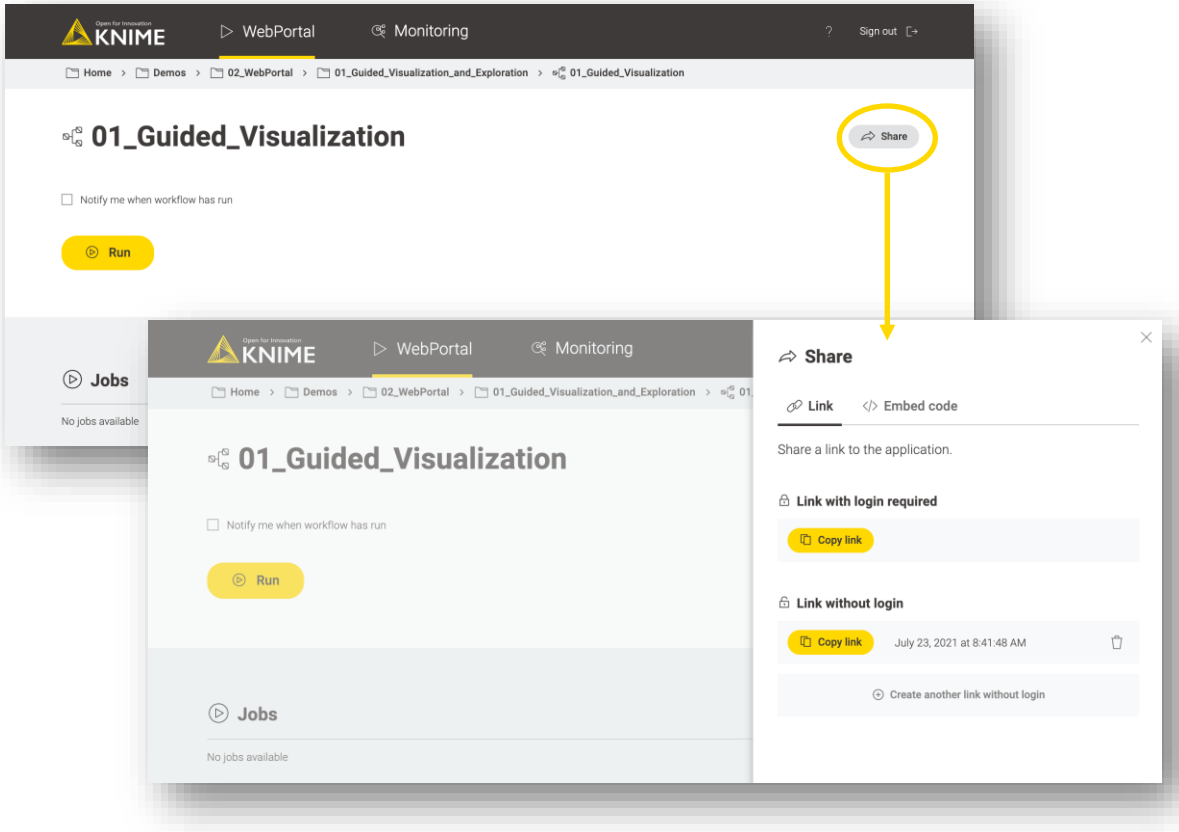
Access to KNIME WebPortal from the Client

- Possibility to start the execution of a workflow on KNIME WebPortal directly from KNIME Analytics Platform: right-click on the workflow available in the KNIME Server Mountpoint and click *Open in WebPortal*



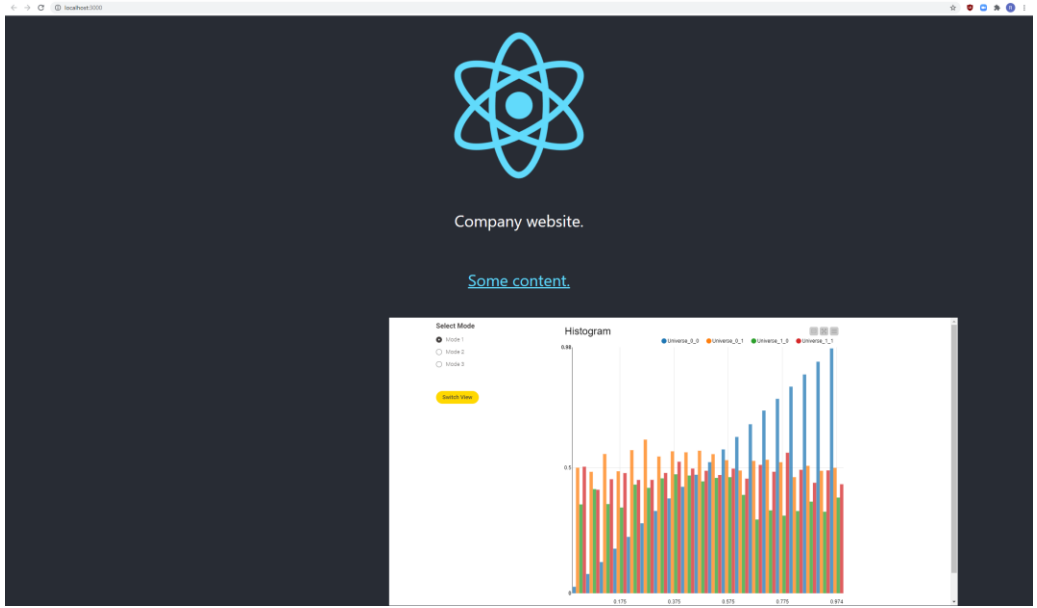
Access to KNIME WebPortal from the Browser

- KNIME WebPortal can also be accessed directly in the Browser with a specific URL
- Simply copy the link via the Share option
- There are two options:
  - Sharing a link that requires login to access the application
  - Sharing the link that does not require login to access the application



Embedded Data App

- Create shareable and embeddable links that allow end users to run Data Apps without the need to log in explicitly.
- These links can be embedded in an iframe on any website, making Data Apps accessible from anywhere to everyone.



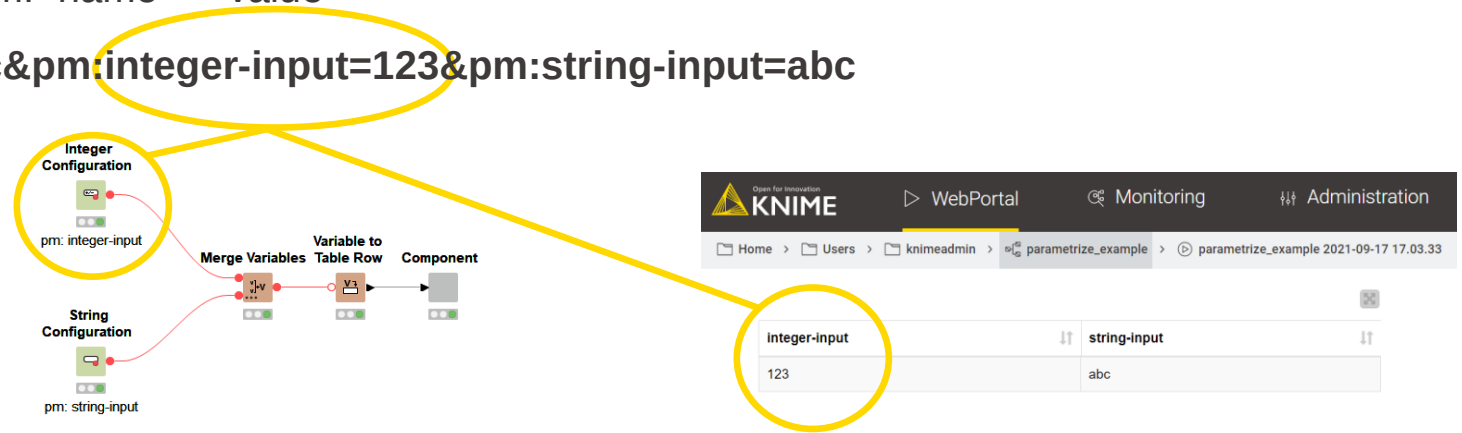
Parametrize shareable URLs

URLs are generally set up like this:

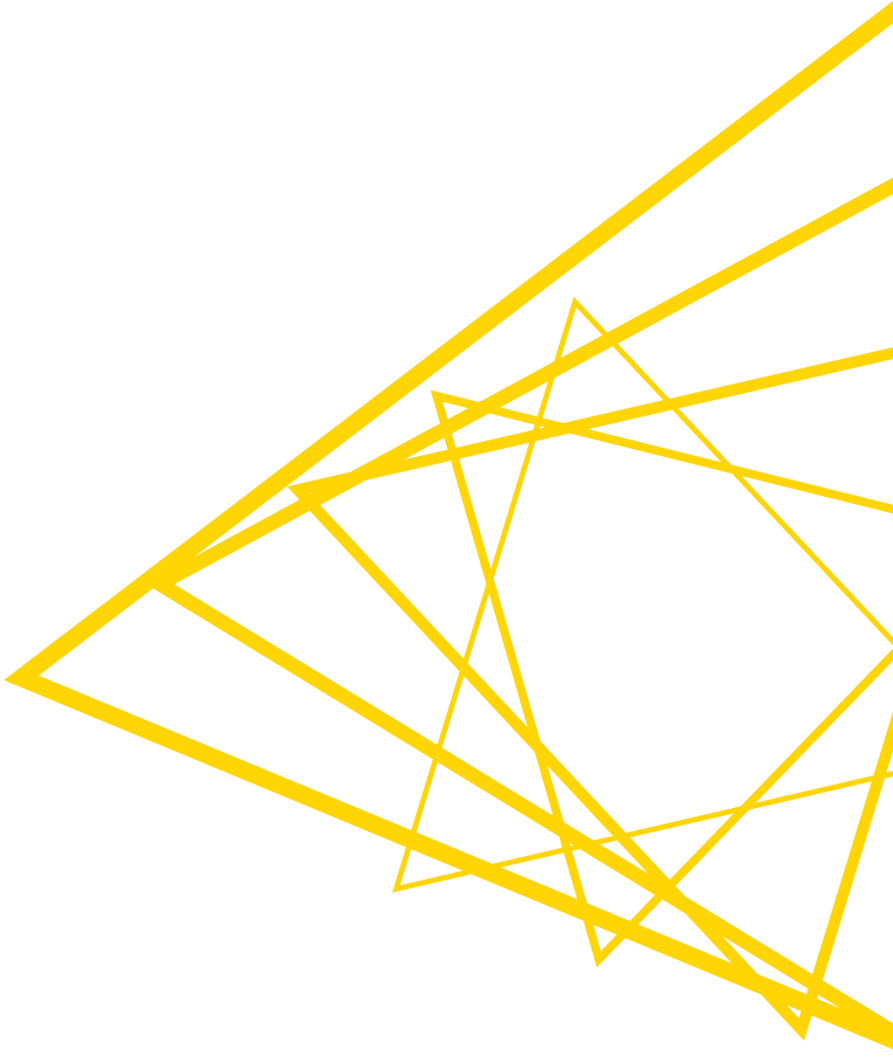
http://<*server-address*>/knime/webportal/<*ItemPath*>?exec&<*WorkflowParameters*>

- <ItemPath>: The path to a workflow/workflow group/workflow job. A workflow job is referenced with its ID like: <ItemPath>?exec=job_id
- ?exec: Redirects to the automatic execution of the workflow.
- <WorkflowParameters>: Can appear in any order appended with a leading ‘&’, e.g. like this &pm:<name>=<value>

?exec&pm:integer-input=123&pm:string-input=abc

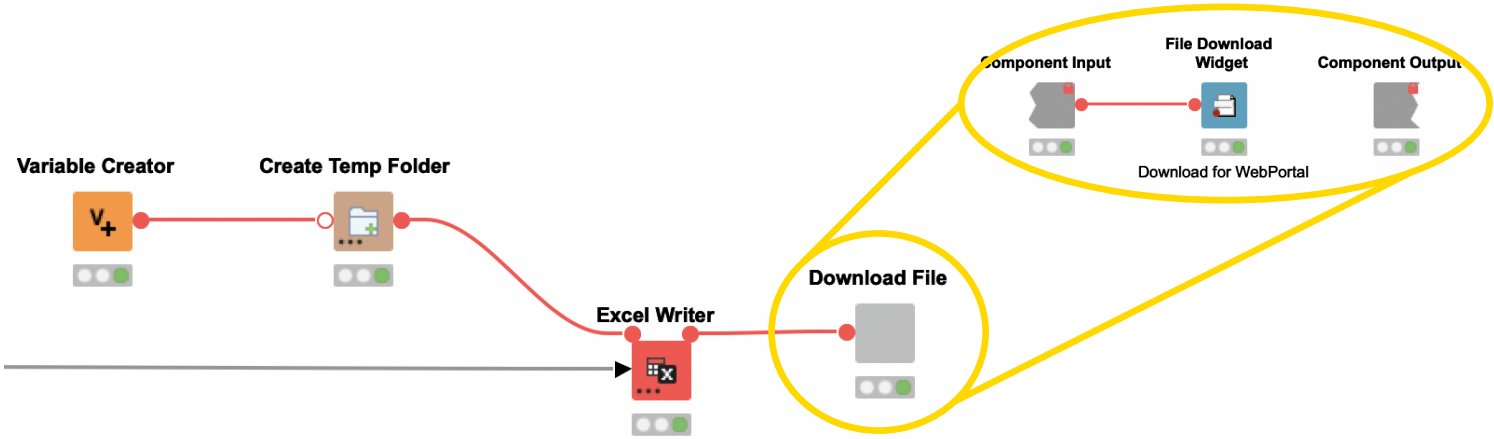


Data Apps with File Download/Upload



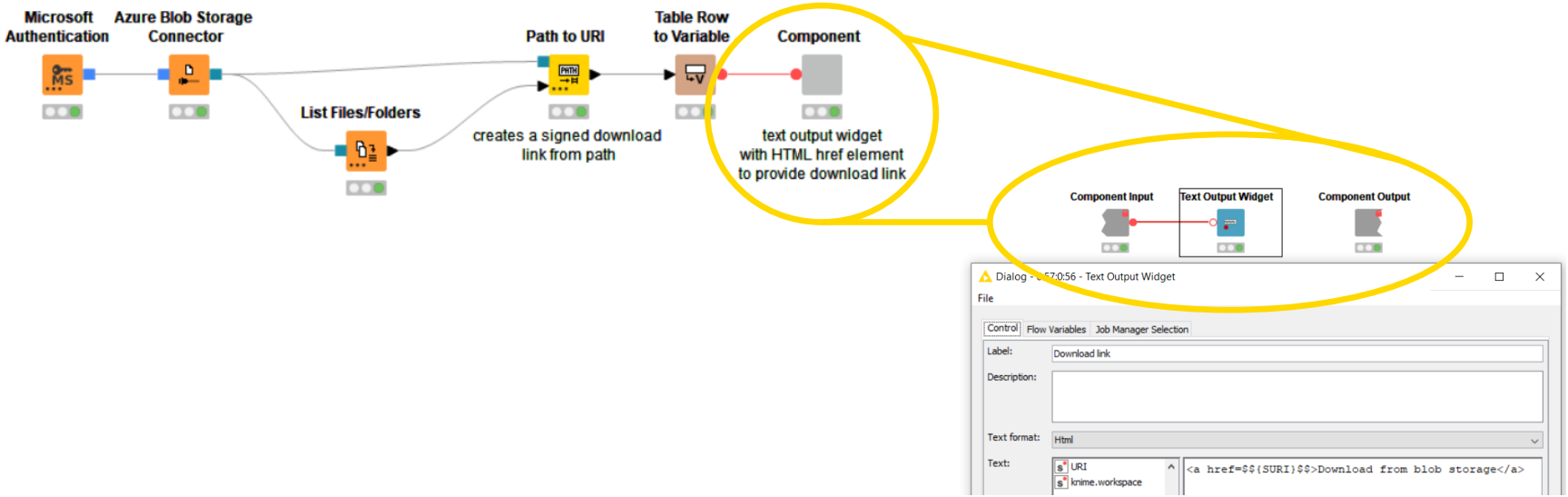
File Download

- Variable Creator: create name of file
- Create Temp Folder: create temporary folder, which is deleted with reset of workflow (Pro-Tip: use temporary folder on Server to avoid pile up of files)
- Excel Writer: writes data into file in temporary folder
- File Download Widget: view with link to download the file



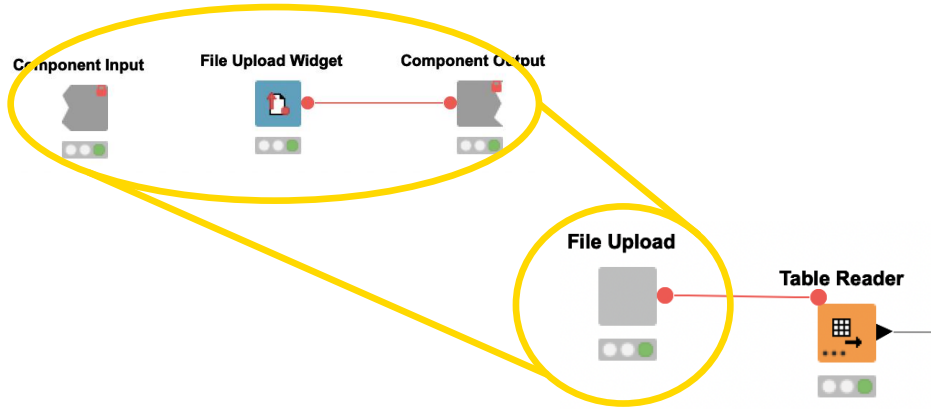
File Download via Text Output Widget

- Some file systems allow creation of signed download links
- Benefit: No need to load or save the file to server or workflow
- Allows to specify expiry date



File Upload

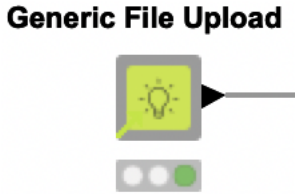
- File Upload Widget: enables the user to select a file, which is uploaded to the Server and the Path Flow Variable is created
- Reader Nodes: Path Flow Variable is passed to the Reader node to read the data into KNIME



IMPORTANT: User can only upload file types that match the selected Reader node

Verified Component: Generic File Upload

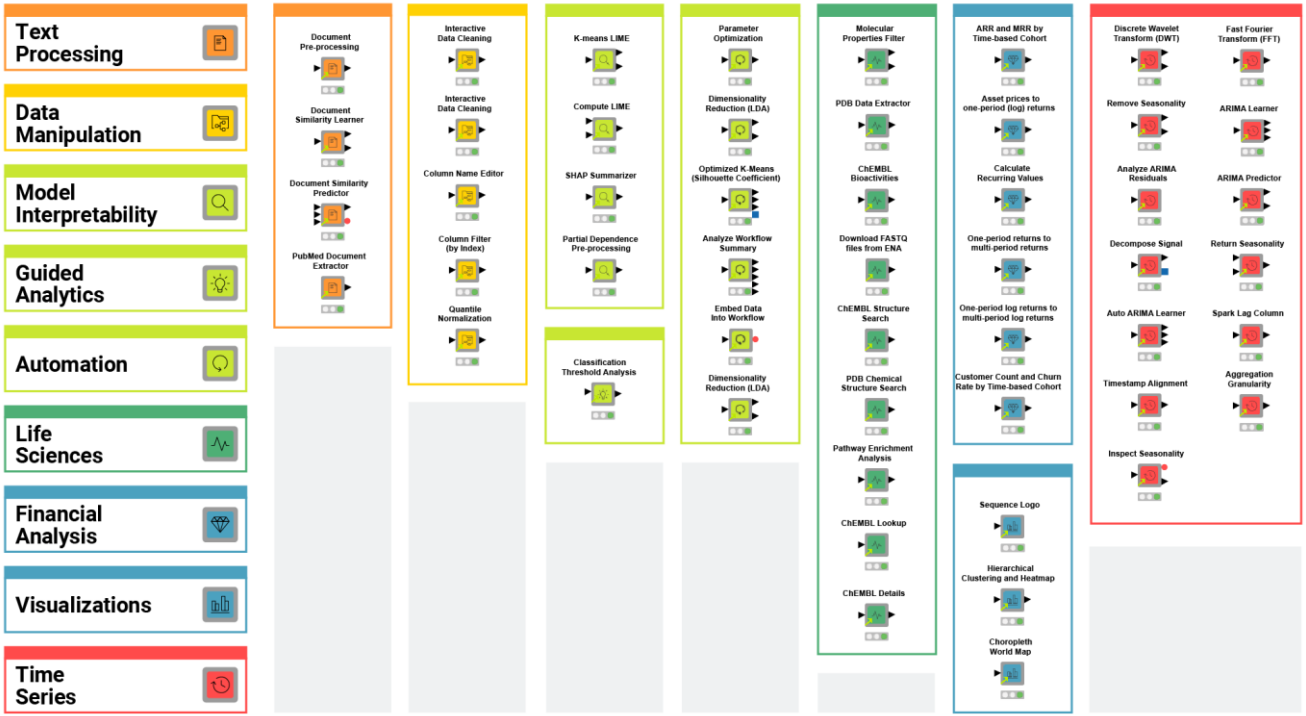
- This component can be used as a generic file upload for the KNIME WebPortal or locally in the KNIME Analytics Platform.
- The user can upload a file of different formats (.csv, .tsv, .xls, .xlsx and KNIME-native .table) and the component selects the correct node to read the data in.



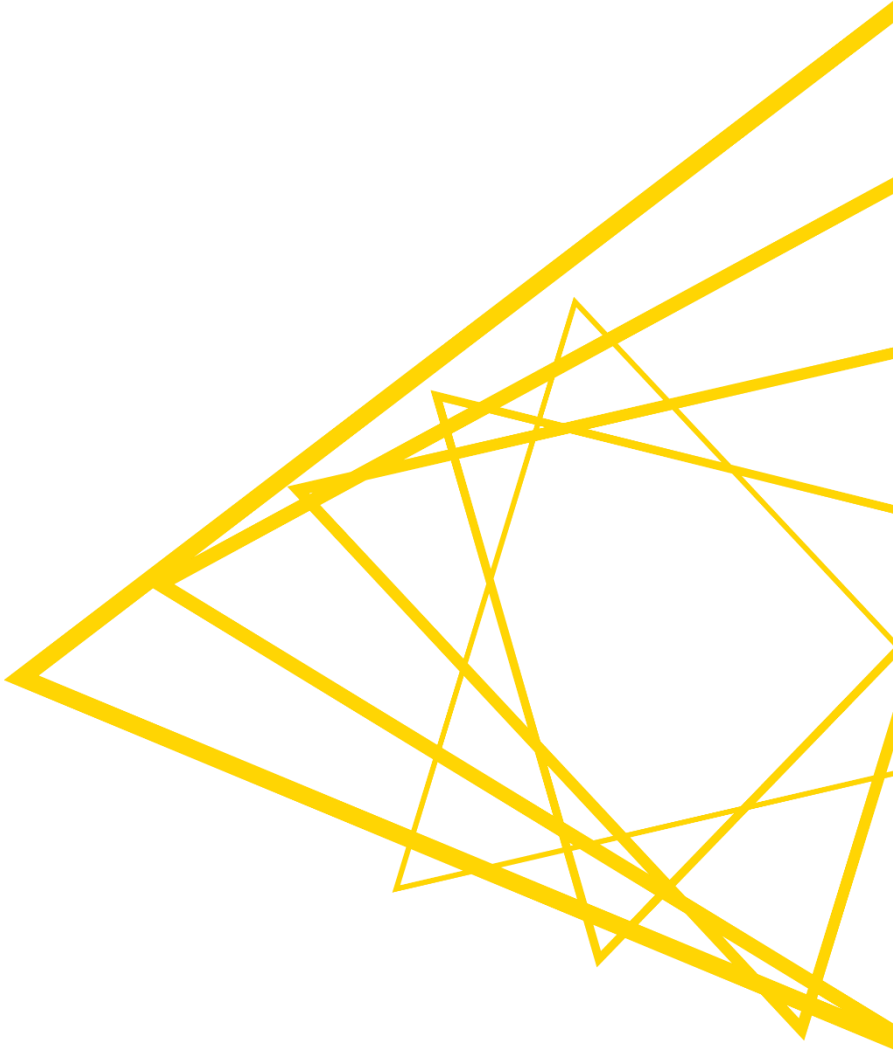
IMPOTANT: This component does not cover all edge-cases for complicated file formats.

KNIME Verified Components

- Verified Components reuse bundled functionalities, verified by KNIME experts
- Released and updated on the KNIME Hub

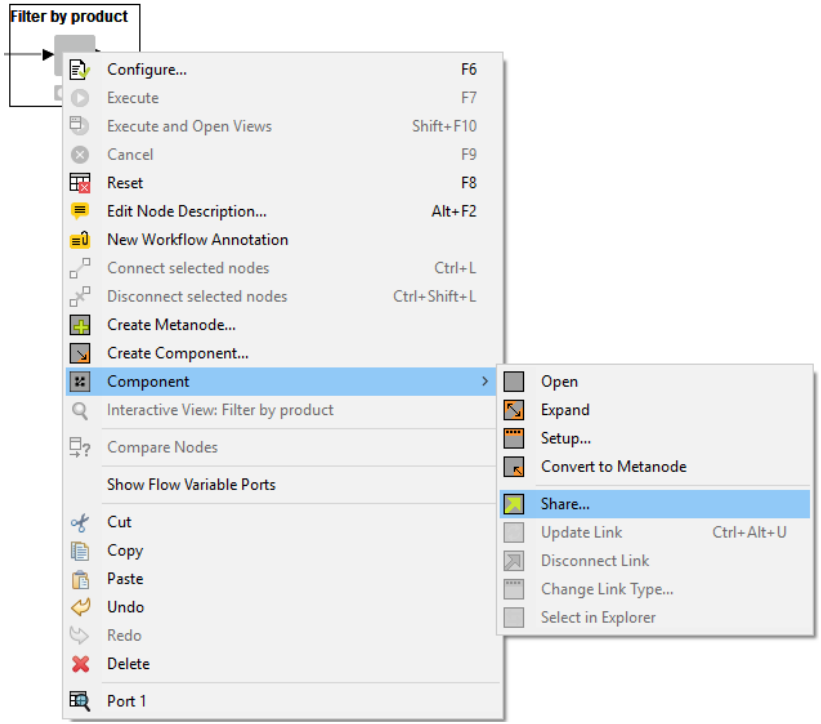


Shared Components



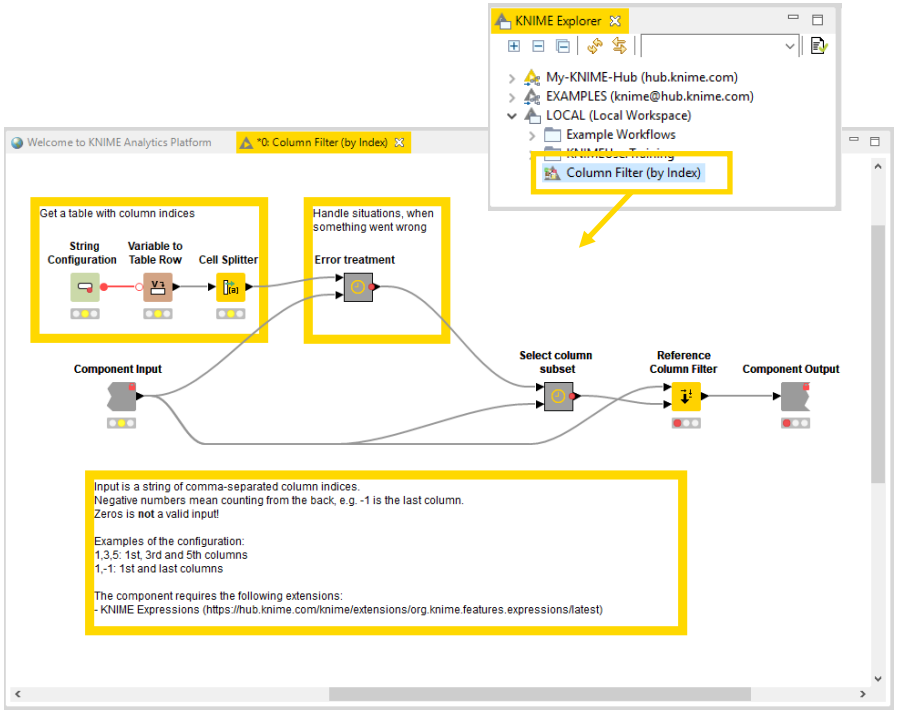
What is a shared component?

- Components can be saved in your KNIME workspace, on KNIME Server or KNIME Hub for later reuse
- To do this, simply right-click any component and select “Share…”
- Shared components are read-only instances of a component
- Public Shared Components are available on the EXAMPLES Server and on the KNIME Hub



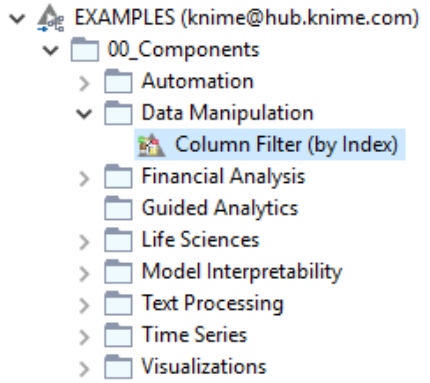
How can you edit a shared component?

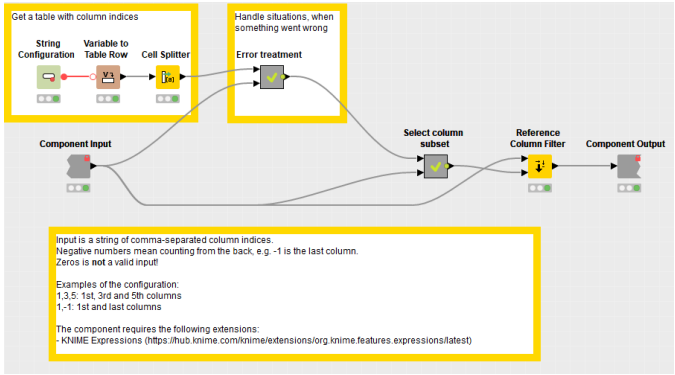
- Components can be edited using the Component Editor, similar to workflows
- To edit a component using the Component Editor, double-click the component in its location in the KNIME Explorer
- To ensure components are executable when opened in the Component Editor, choose the option to “Include input data with component” when sharing it



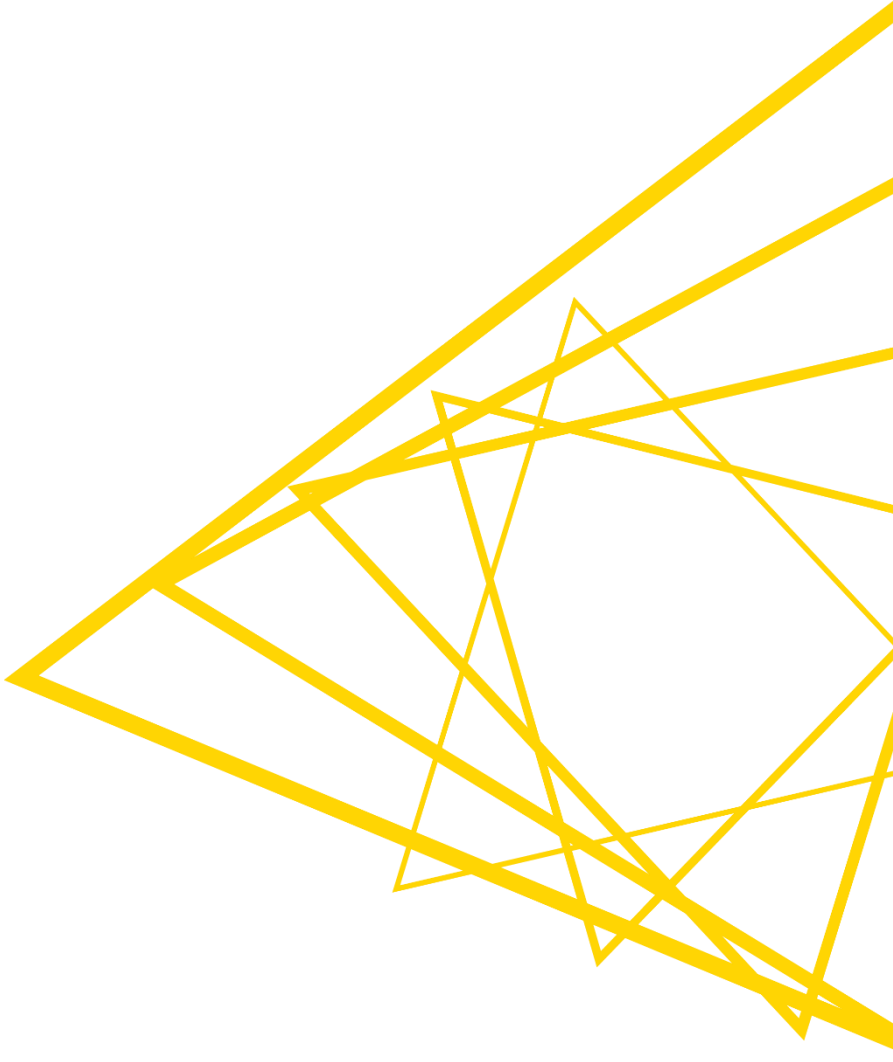
How can you use a shared component?

- To use a Shared Component, drag and drop it to your workflow editor
- Instances of Shared Components can be updated either manually or when the workflow is opened
- A Shared Component can also be unlinked from its original location, which makes it editable in the workflow directly
- Update Shared Components by overwriting them



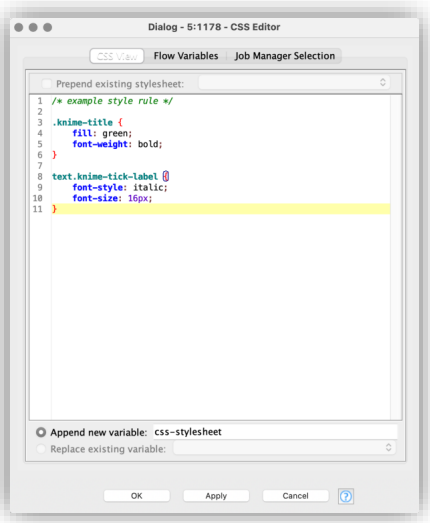


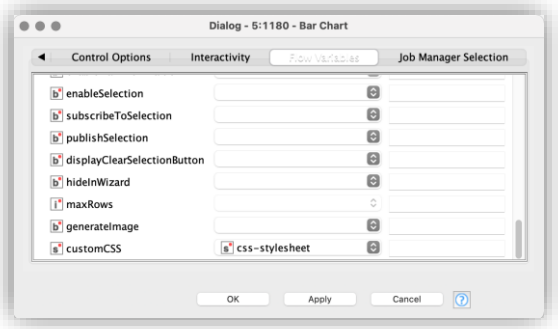
Customize Styling



CSS Editor

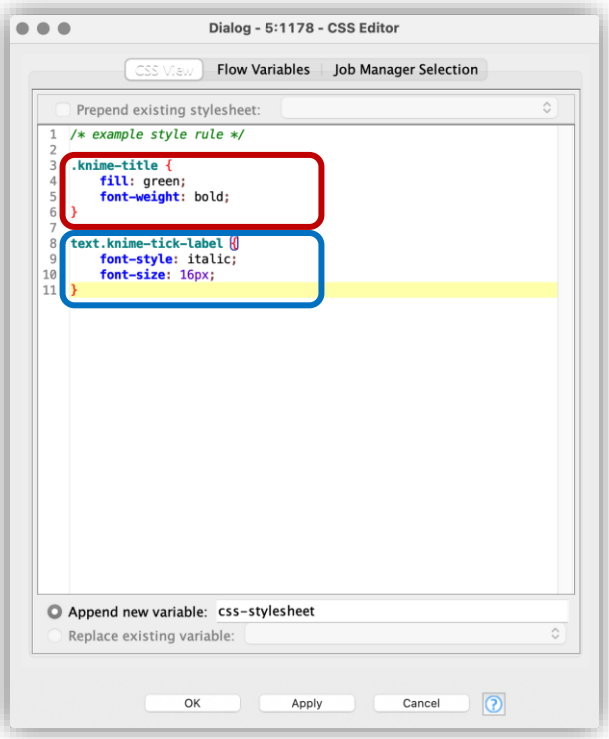
- With this node custom CSS styling for JavaScript views and Widgets can be created.
- Write CSS rules into the editor of the node dialog and set the output to create a new flow variable.

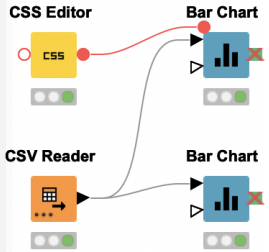


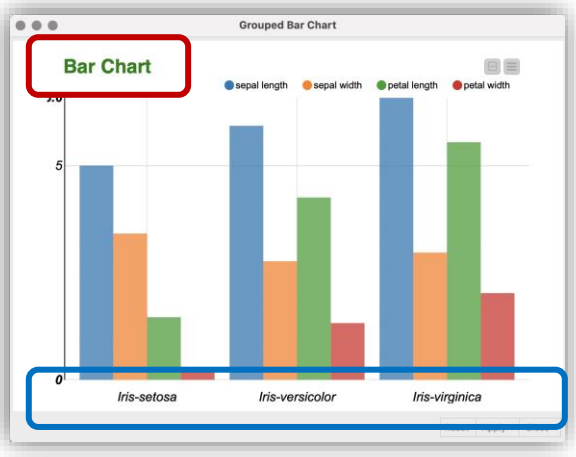


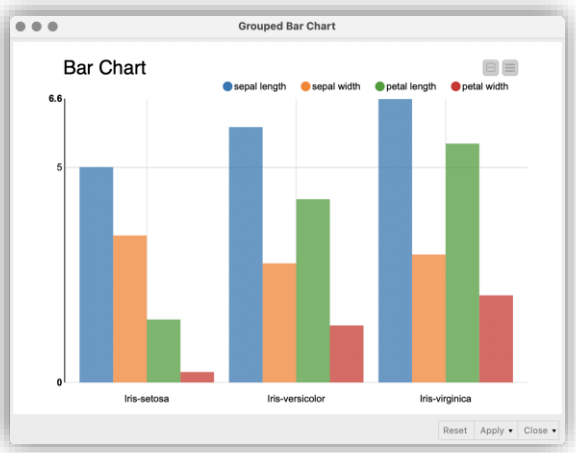
Documentation: https://docs.knime.com/2021-06/analytics_platform_css_guide/index.html#_css_styling_in_knime_analytics_platform_4_2

CSS Editor





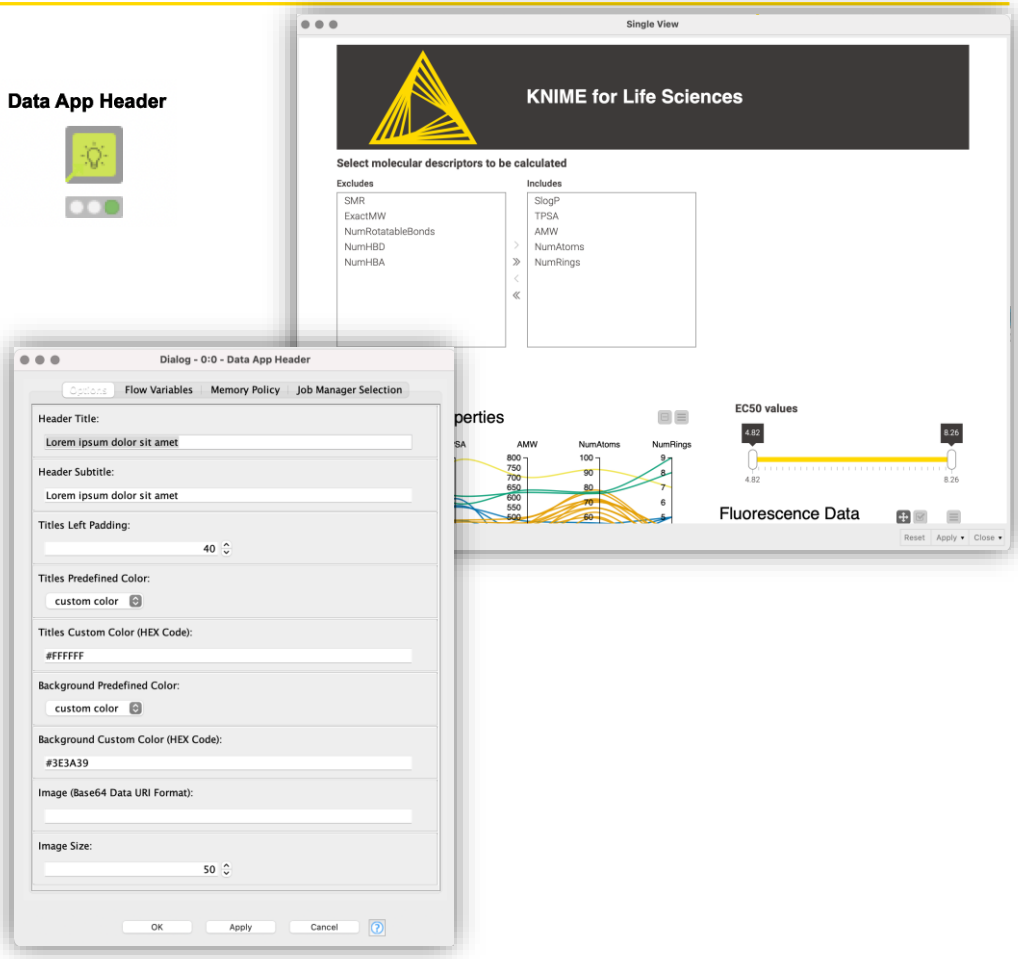




Verified Component: Data App Header

- Creates an engaging header for your data app

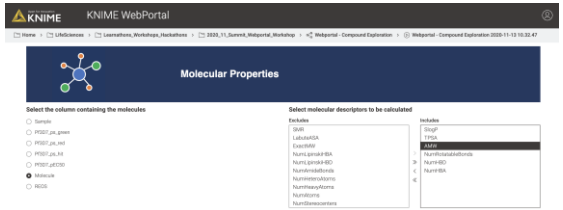
- Configurable
  - Title
  - Text Color
  - Background Color
  - Image as Base64 format

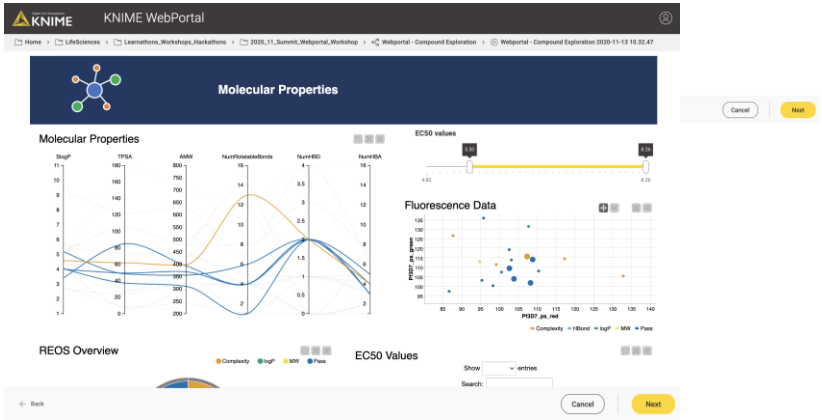


Theming

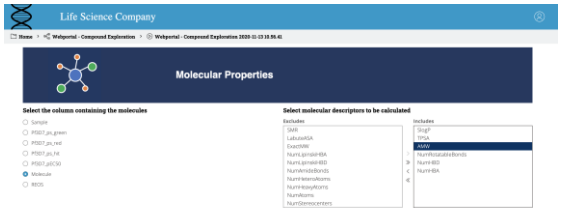
- Apply a customized theme to the WebPortal (see Documentation)
- Admin rights necessary

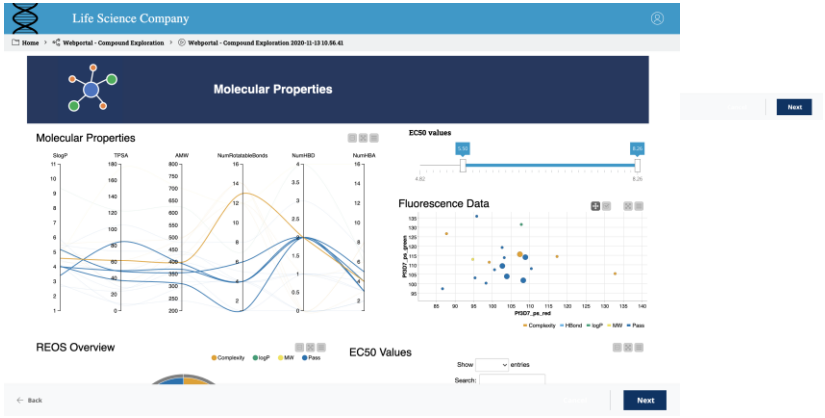
Default Theme





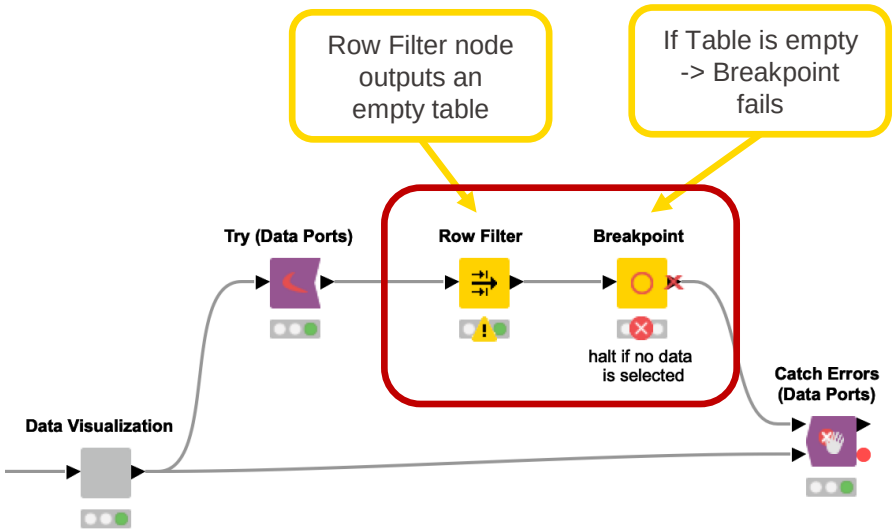
Customized Theme





Best Practice #1: for data app developer

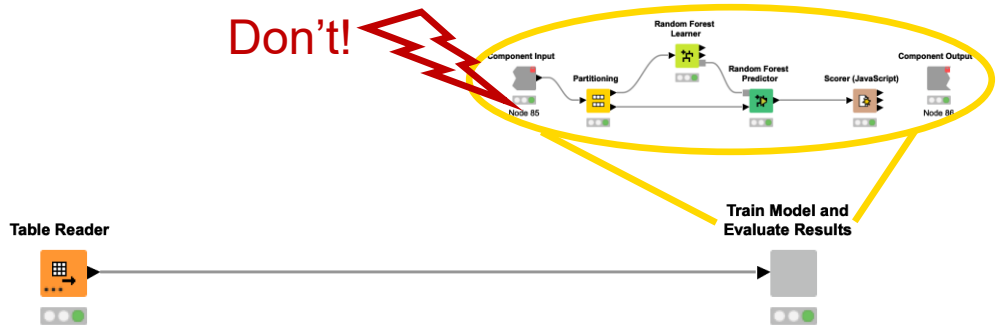
- Error Handling: In cases where the user does not select any data points
  - Usually, you apply a Row Filter node to only keep the selected data points
  - To account for the case that the user does not select any data point, use the Try-Catch Motif

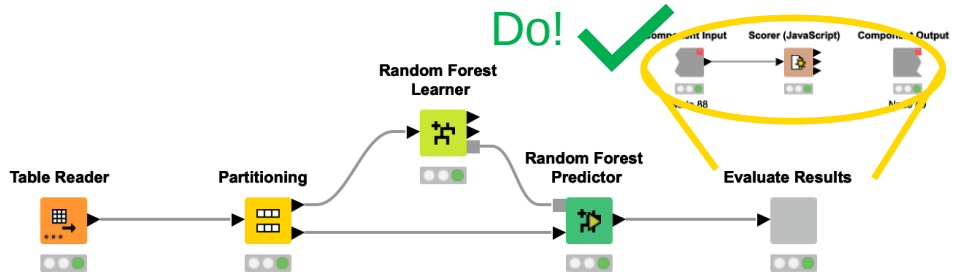


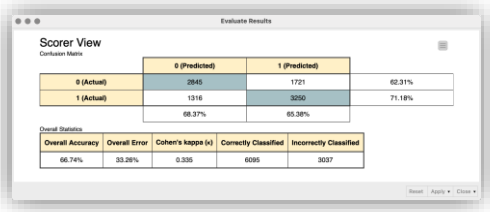
Best Practice #2: for data app developer

- Components are re-executed after user interaction/inputs

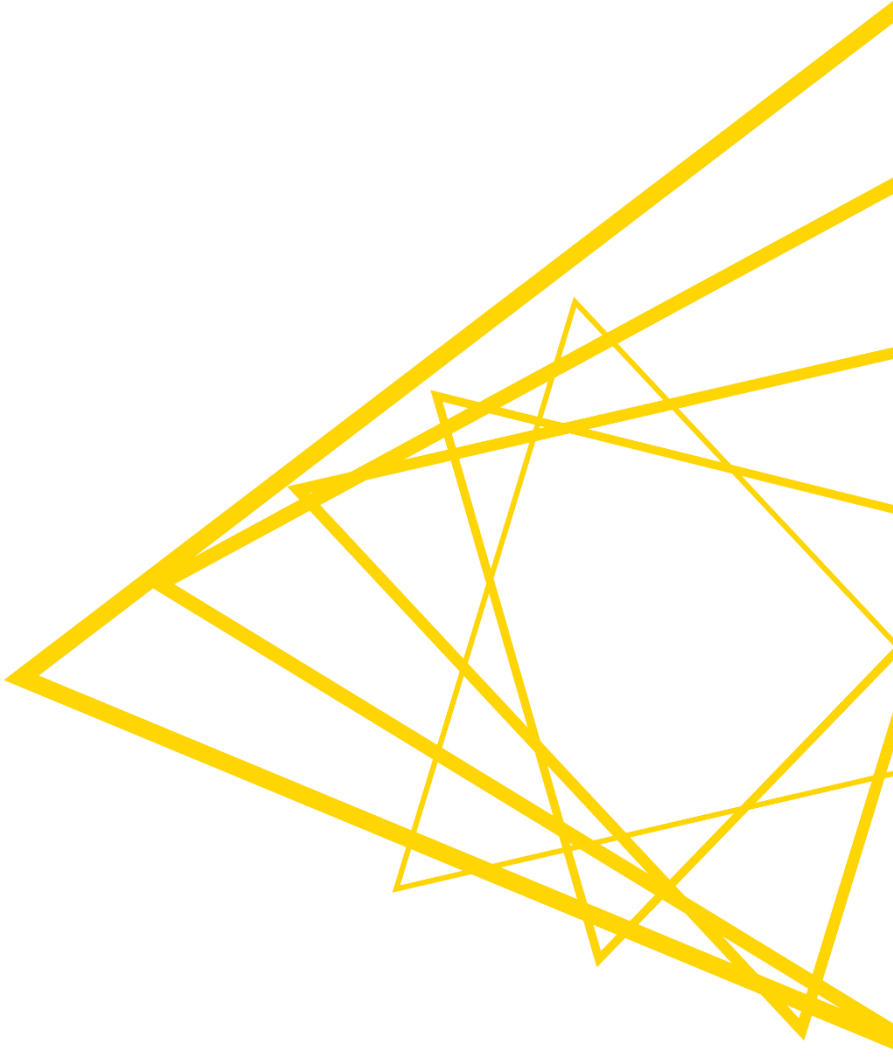
- IMPORTANT:
  - Keep complex/computationally-extensive nodes outside of the component if possible.
  - Only contain widgets or visualization nodes in component







Review and Exercises



Exercise 04. Data app – File Upload+Download

- Open Exercise 04. Data app – File Upload+Download:
  - Extend the data app from the previous exercise to
    - provide a file upload to load data into the app and
    - Provide a file download to enable the user to download the results

  - **Deploy workflow on the KNIME Server and run the data app in the browser**

Config Details – Access KNIME Server

- KNIME Server Address: http://3.123.122.13
- KNIME WebPortal Address: http://3.123.122.13/knime

Login credential: *firstname.lastname* (*)

Password: *knime*

*for double names and double surnames the whitespace has been removed

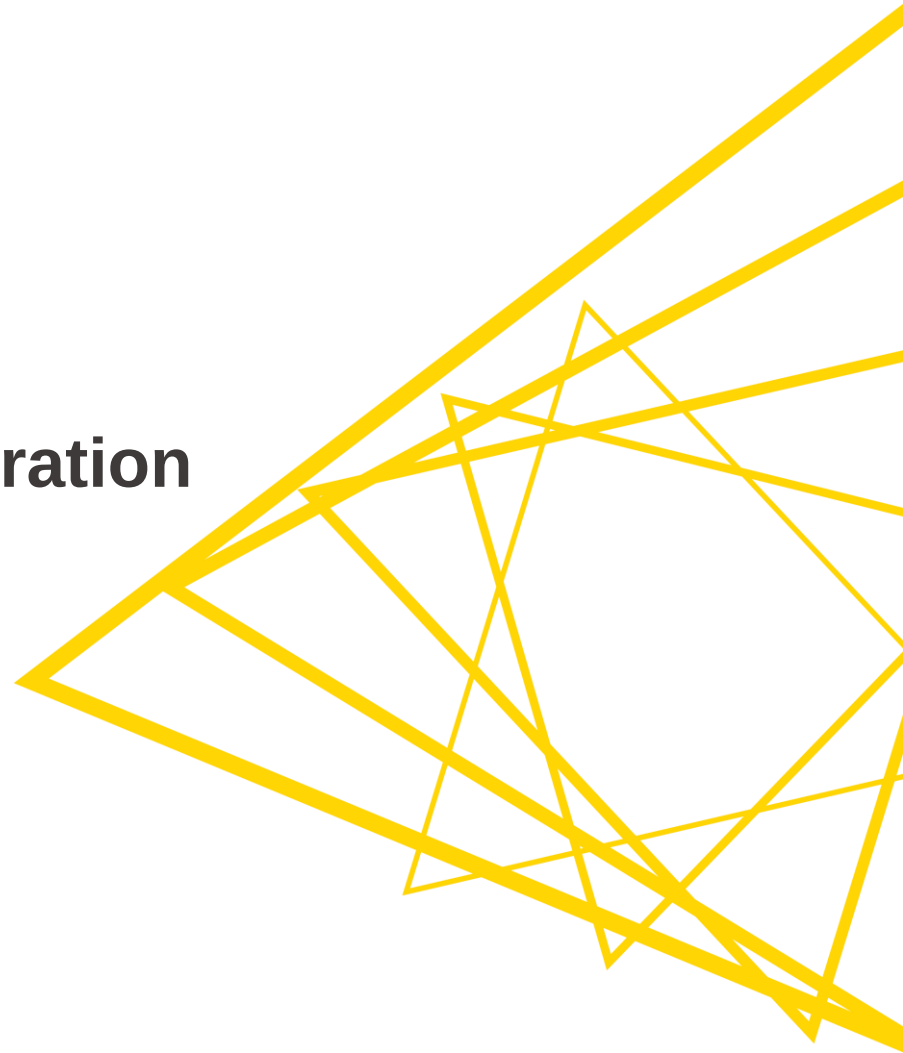
Open for Innovation
KNIME

Thank You!

Open for Innovation
KNIME

[L3-PC] KNIME Server: Productionizing and Collaboration

KNIME GmbH



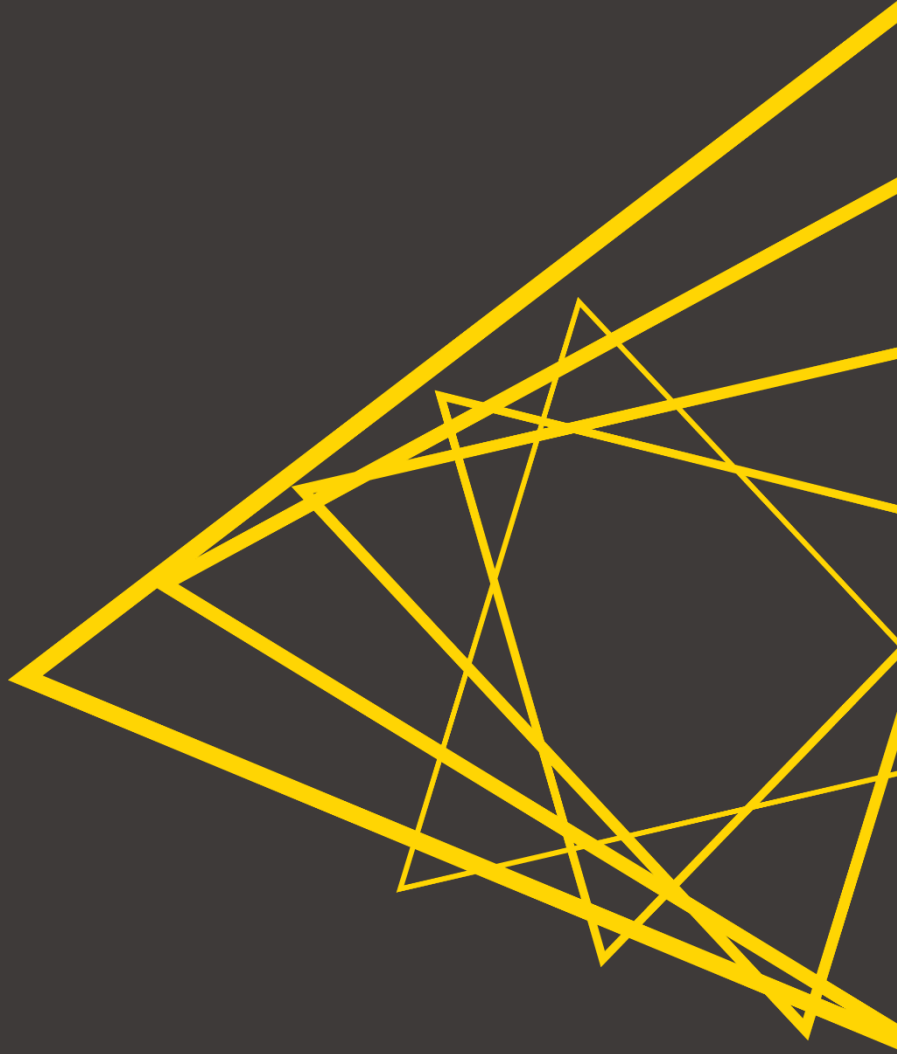
Structure of the Course

Session	Topic	Duration
Session 1	Introduction to KNIME Software & Working with KNIME Server	75 min
Session 2	Creating Data Apps with Components	75 min
Session 3	Deploying WebPortal Data Apps	75 min
Session 4	**Integrated Deployment, REST API & Administration**	75 min
Session 5	Wrap Up Session	15 min

Structure of each session

- Discussion of past exercises
- Course
- Introduction of next exercises

KNIME Server REST API



KNIME Server REST API

- Integrate KNIME Server functionality with IT infrastructure
- REST is a design pattern used for building networked applications
- REST = Representational State Transfer
  - Communication based on HTTP
  - Usually clear text
- Execute workflows, check server status, set permissions, and more
- Entry point for the REST Interface is https://server-address/knime/rest/

See Blog Posts for detailed tutorials:

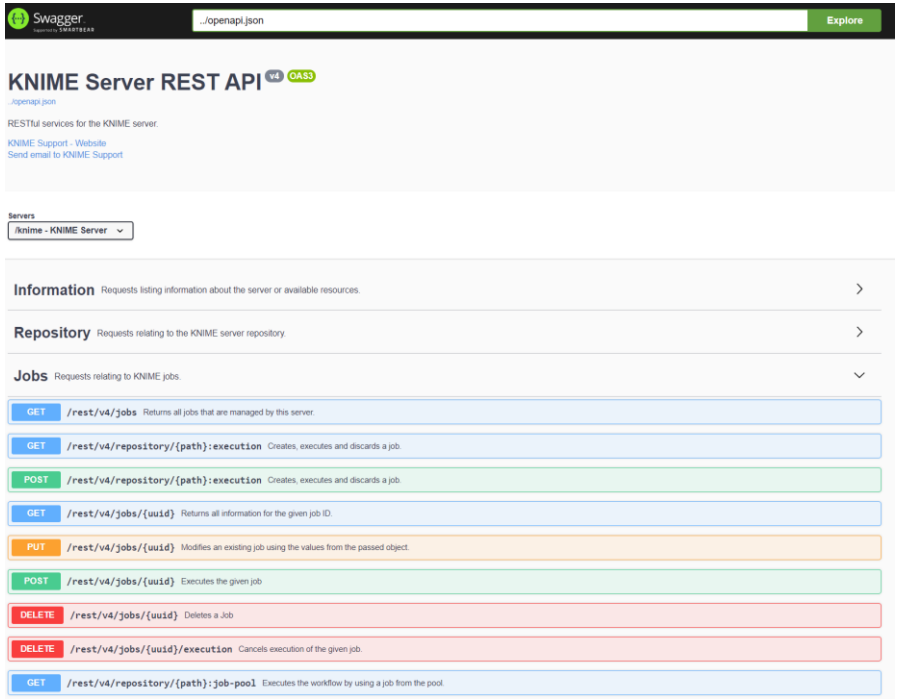
https://www.knime.org/blog/giving-the-knime-server-a-rest

https://www.knime.org/blog/the-knime-server-rest-api

https://tech.knime.org/wiki/using-knime-server-rest-api-for-file-uploads-and-downloads

KNIME Server REST API

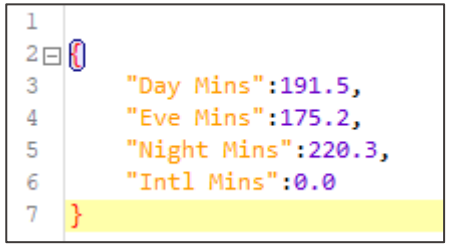
- All server functionality available via REST API
- Programmatically control KNIME Server
  - Upload/download/delete resources
    - Staging from Dev to Prod
  - Upload licenses
  - Empty trash/restore items
  - Execute workflows
  - Schedule jobs
  - Set permissions
  - Create users & groups, etc.
- Documentation available at: /knime/rest/doc/index.html

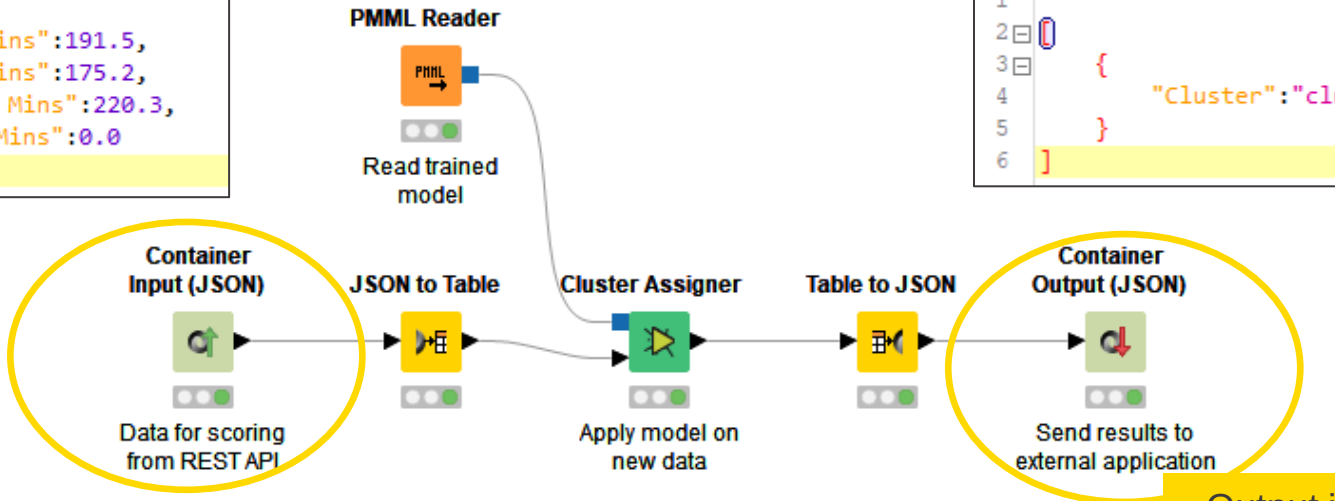


KNIME Server REST API

- Enables external integration
- Build applications around KNIME Server
- e.g. for Microservices and real time scoring

Deploy KNIME workflows as web services



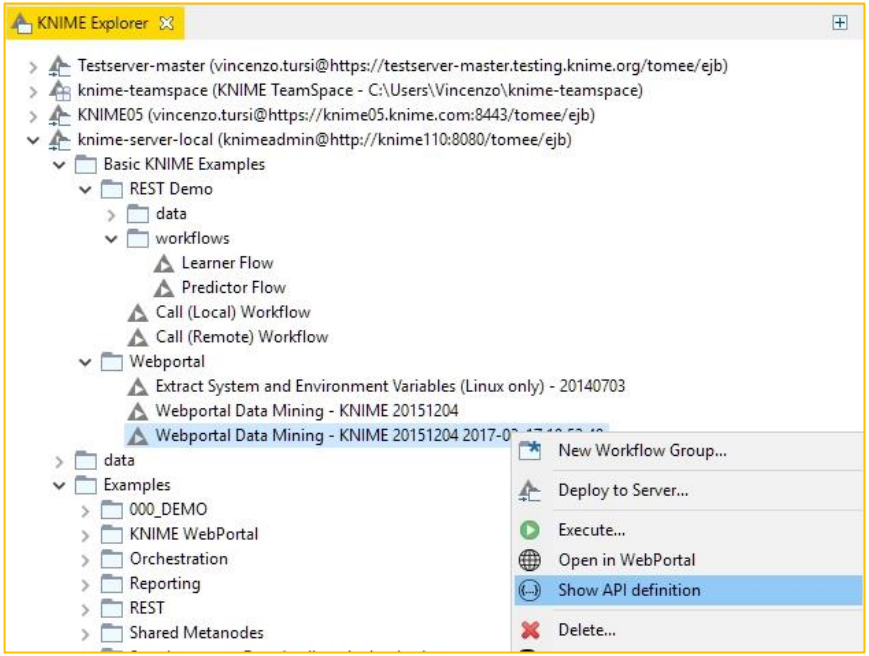


Input data

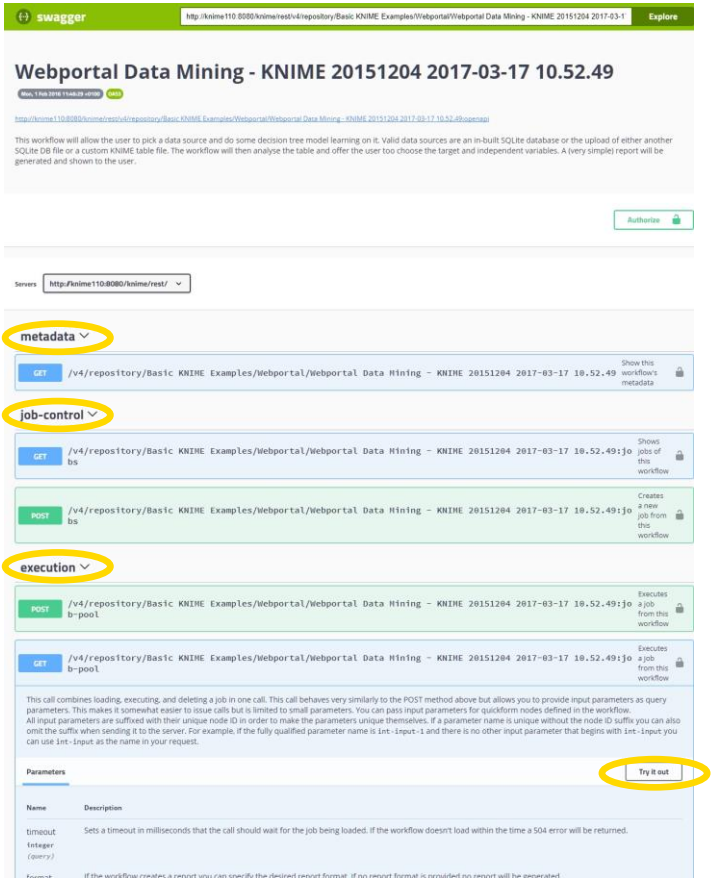
Output in server response

SwaggerUI definitions of individual workflows (KNIME Server *Medium* and *Large*)

- One of the key functionality of KNIME Server is to allow individual workflows to be exposed as REST endpoints
- SwaggerUI interface allows to document and test your web services
- “Show API definition” is available in the context menu item on the KNIME Explorer. Selecting this option opens the SwaggerUI page for that service in your web browser.

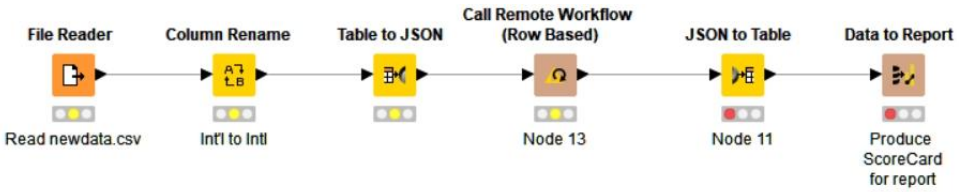


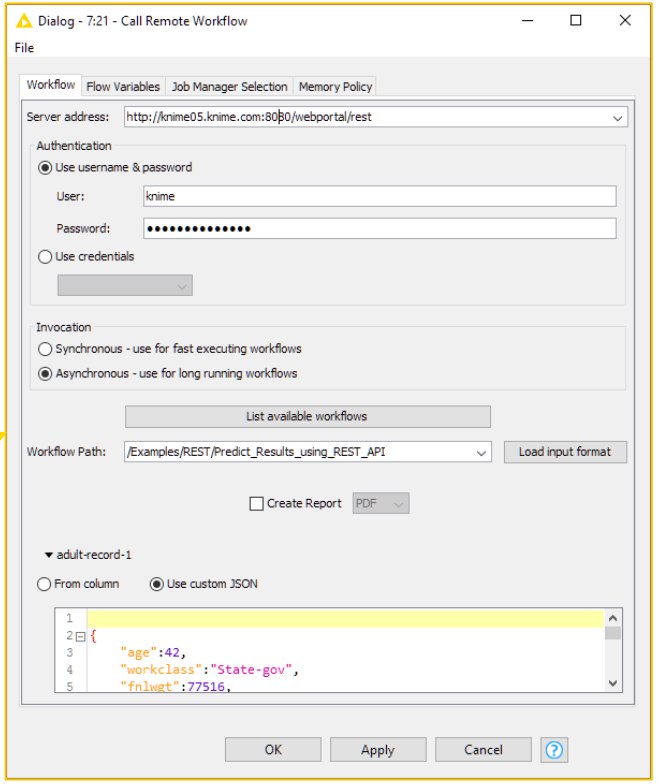
SwaggerUI definitions of individual workflows



REST API – Use Case: Workflows Calling Workflows

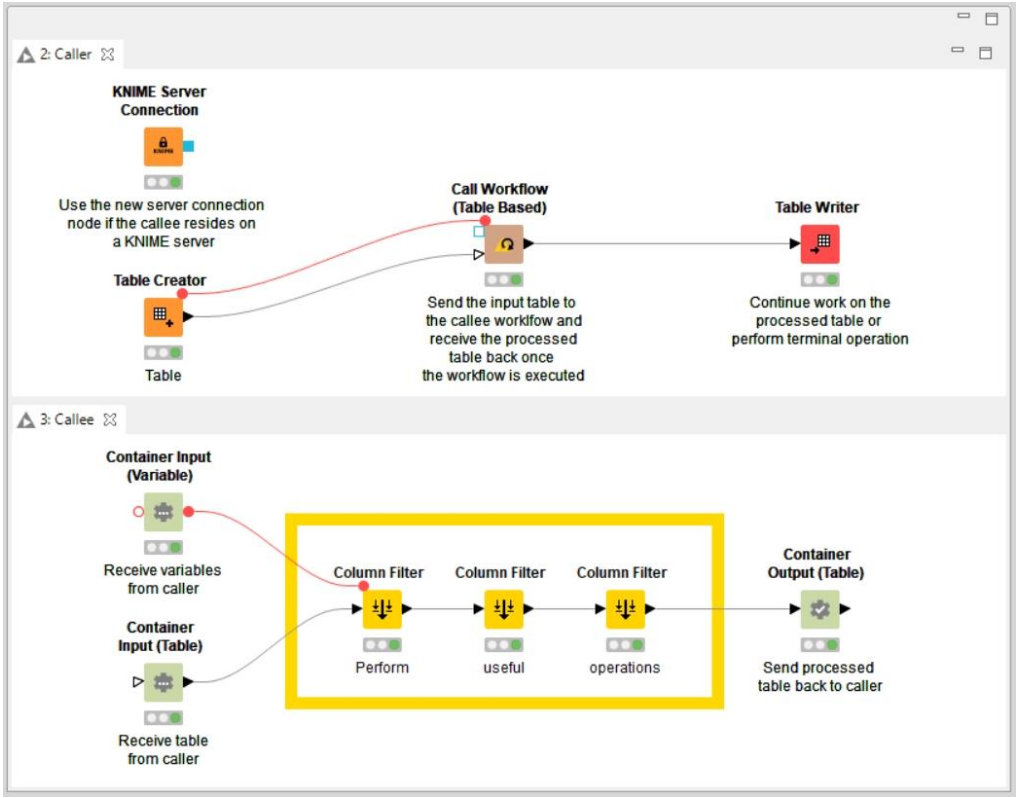
- Execute the workflow via Call Remote Workflow node
  - Analyses input parameters
  - Prepare input data accordingly
  - Executes job and gets back results





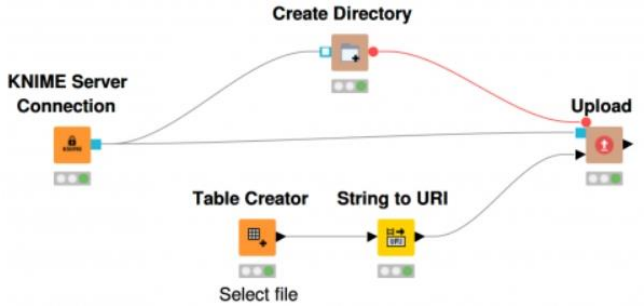
New Node: Call Workflow (Table Based)

- Makes it easier to call other workflows using an entire KNIME table
- A caller workflow can send a table and flow variables to a callee workflow and receive a table from the callee via the Container Input (Table) node



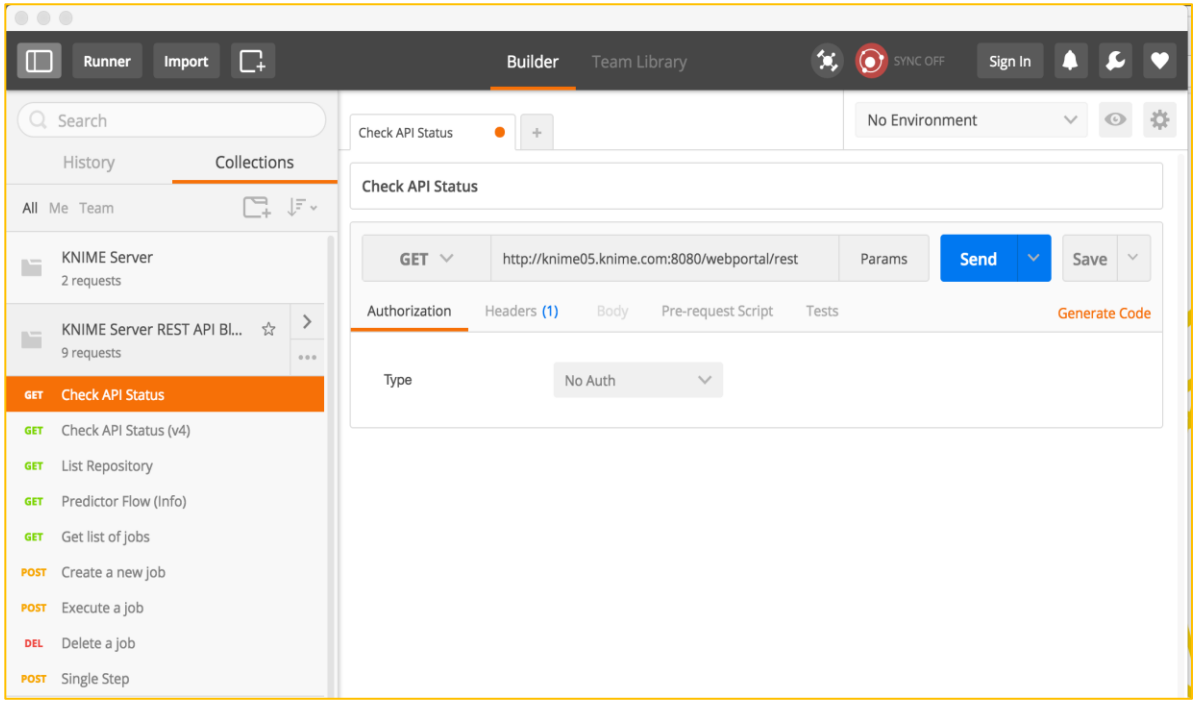
New Node: KNIME Server Connection

- Allows to connect to KNIME Server
- After a connection has been established, any of the remote file handling nodes can be used with the connected server
- The server connection can also be used together with the Call Workflow (Table Based) node in order to run workflows that are shared via a KNIME Server

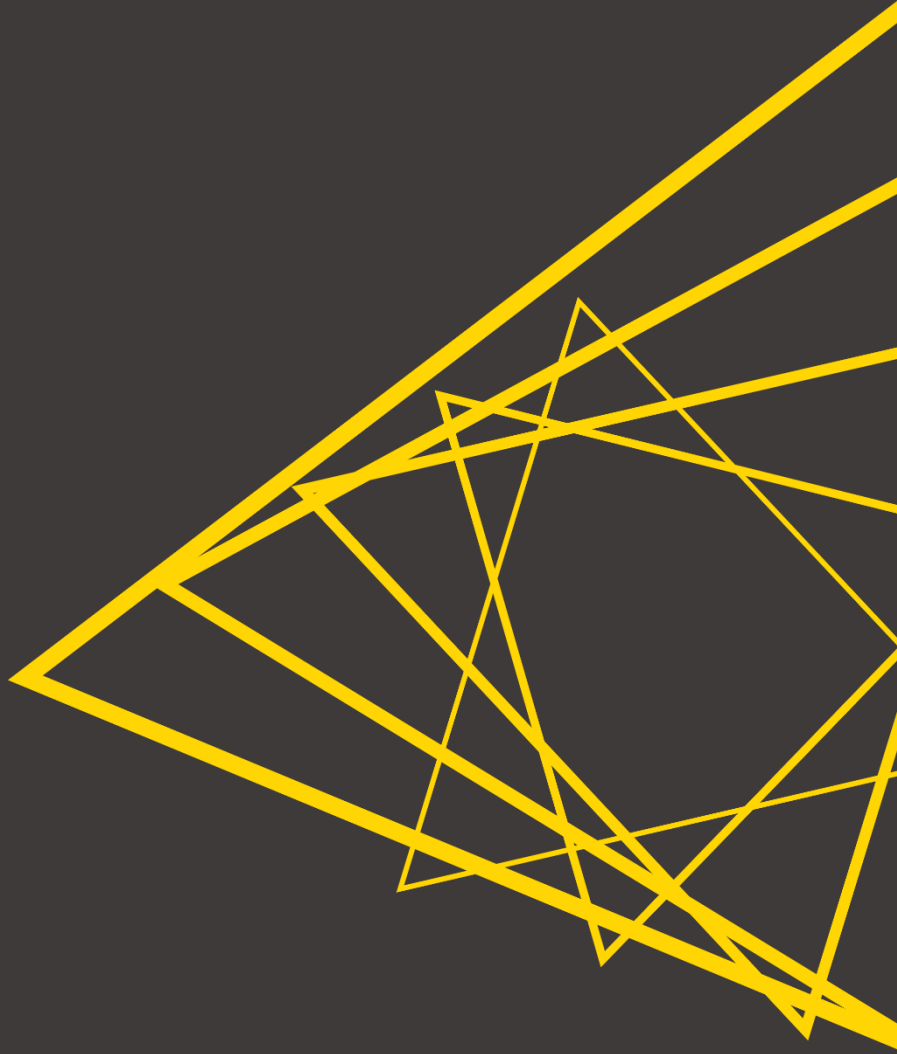


REST API – Use Case: Integration with External Applications

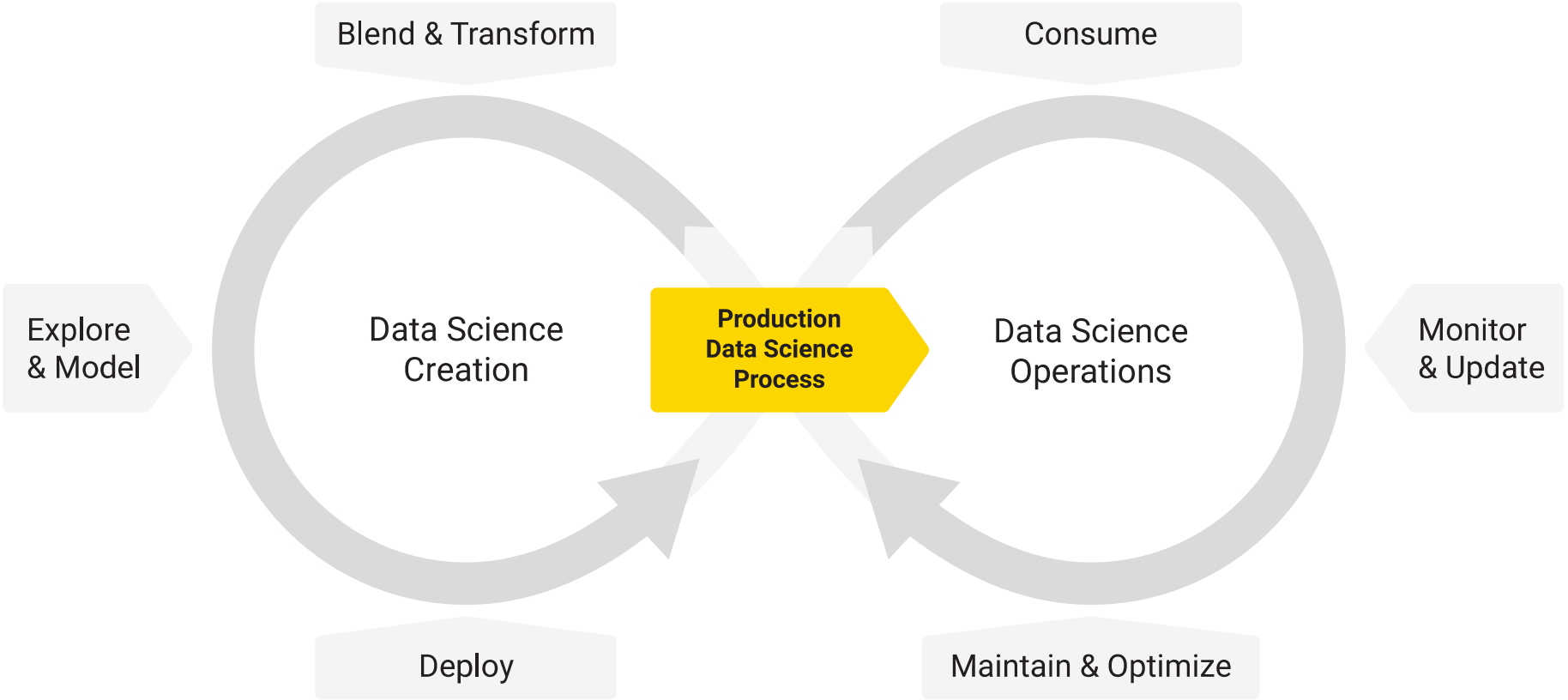
- The workflow can also be executed by external tools such as Postman or Curl for debugging purposes
- KNIME Server as backend for third party analytical applications



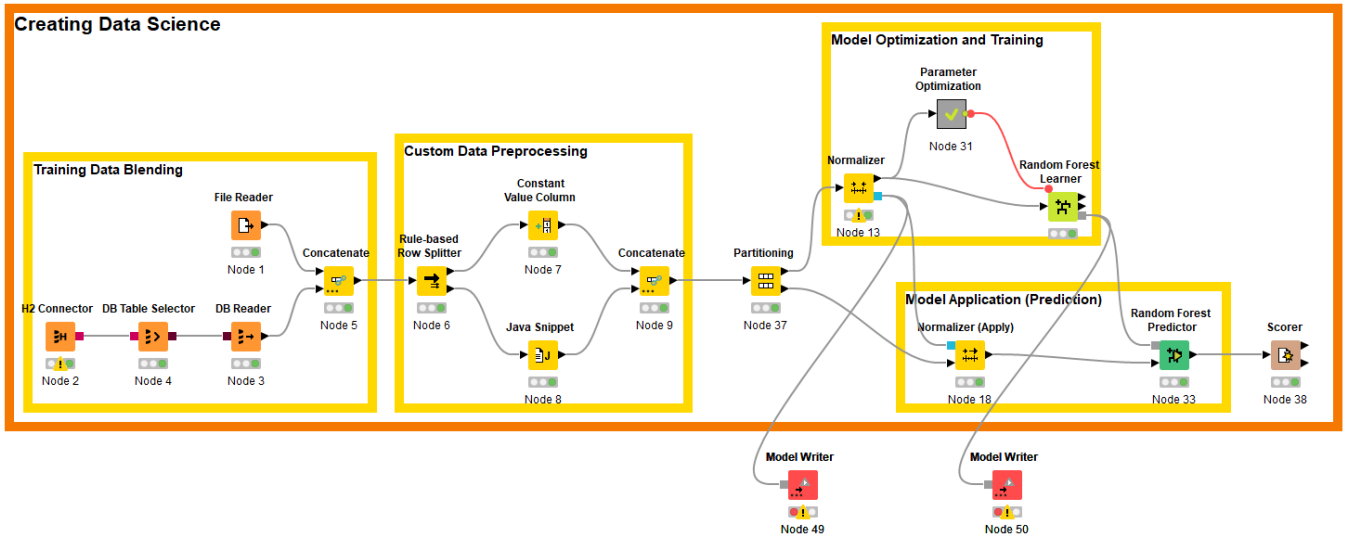
Integrated Deployment



Data Science: Development != Deployment



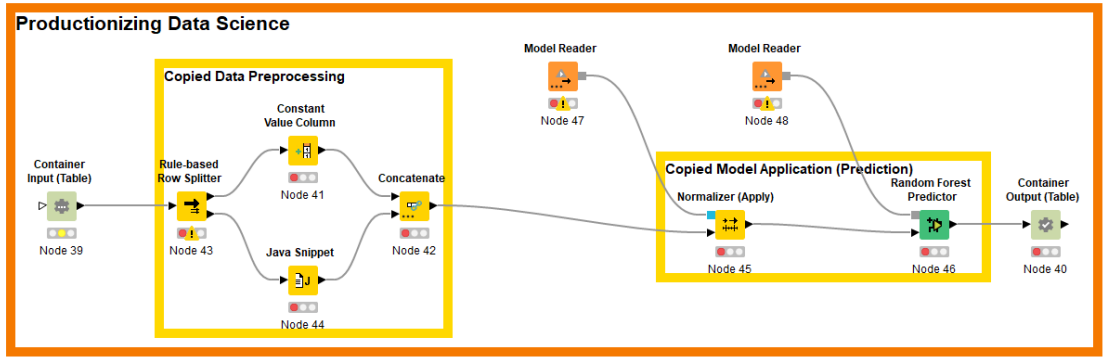
Moving Data Science into Production



Issues:

- Development =! Deployment
- Needs Copy/Paste, Rewrite
- Transport of models is non-trivial

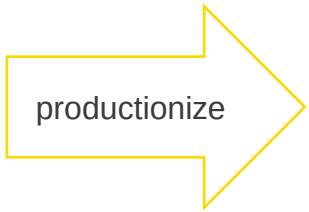
⇒ Inefficient & Error prone



Side Note: It’s not just a Visual Workflow Issue

Creating Data Science

```
# read data
raw_target_data = read_xls_data()
# remove duplicates, handle missing values:
target_data = basic_data_cleanup_with_pandas(raw_target_data)
raw_feature_data = fetch_db_data_using_psycopg2()
# remove duplicates, handle missing values:
feature_data = basic_data_cleanup_with_pandas(raw_feature_data)

# basic feature engineering with sklearn
feature_scaler = sklearn.preprocessing.StandardScaler().fit(feature_data)
standardized_features = feature_scaler.transform(feature_data)
filtered_feature_data = variance_feature_filter_with_sklearn(standardized_features,target_data)

# build model with sklearn
training_feature_data,testing_feature_data,training_target_data,testing_target_data =
split_data_with_sklearn(filtered_feature_data,target_data)
RF_params = RF_hyperparameter_search_with_sklearn(training_feature_data,training_target_data)
trained_RF = build_RF_using_params(training_feature_data,training_target_data,RF_params)

# validate model
generate_validation_report(trained_RF,testing_feature_data,testing_target_data)

#--------------------
# save models
save_models_with_joblib((feature_scaler,trained_RF))
```

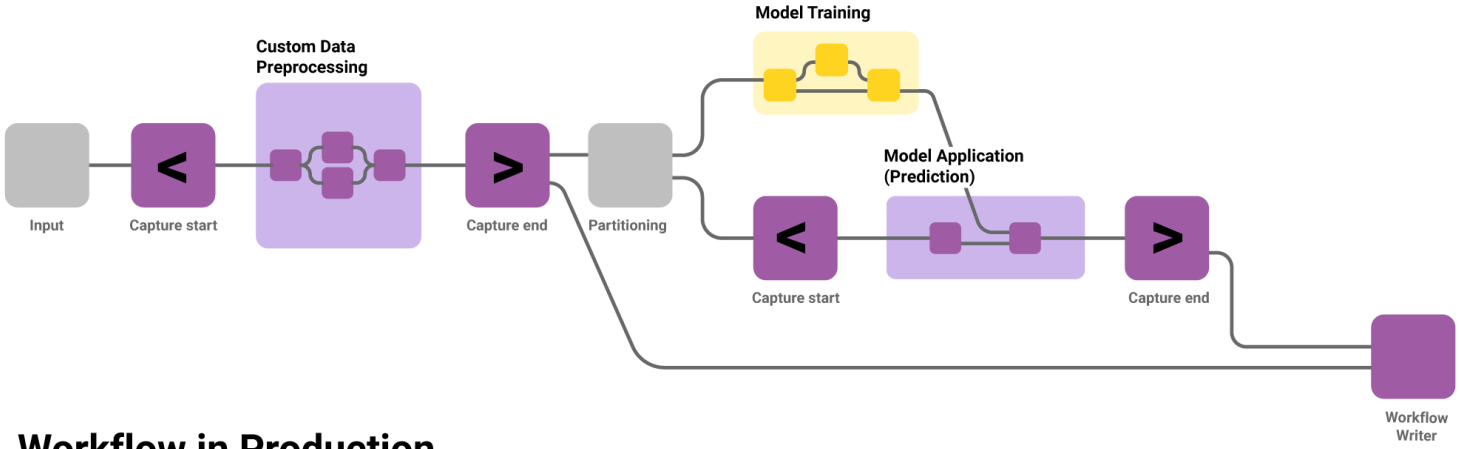


Data Science in Production

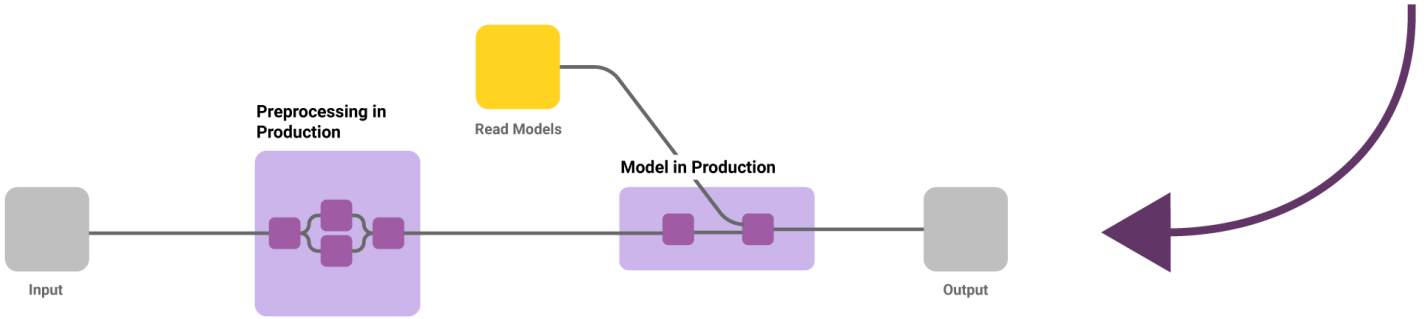
```
# Predictions, running as a flask service

# load saved components
feature_scaler,trained_RF = load_models_with_joblib()

# read and prepare data
raw_prediction_data = get_dataframe_from_request()
prediction_data = basic_data_cleanup_with_pandas(raw_prediction_data)
scaled_data = feature.scaler.transform(prediction_data)

# generate predictions
predictions = trained_RF.predict(scaled_data)
prediction_probs = trained_RF.predict_proba(scaled_data)
predictions = join_tables_with_pandas(predictions,prediction_probs)

# return results from the service:
return_dataframe_to_service(predictions)
```

Integrated Deployment

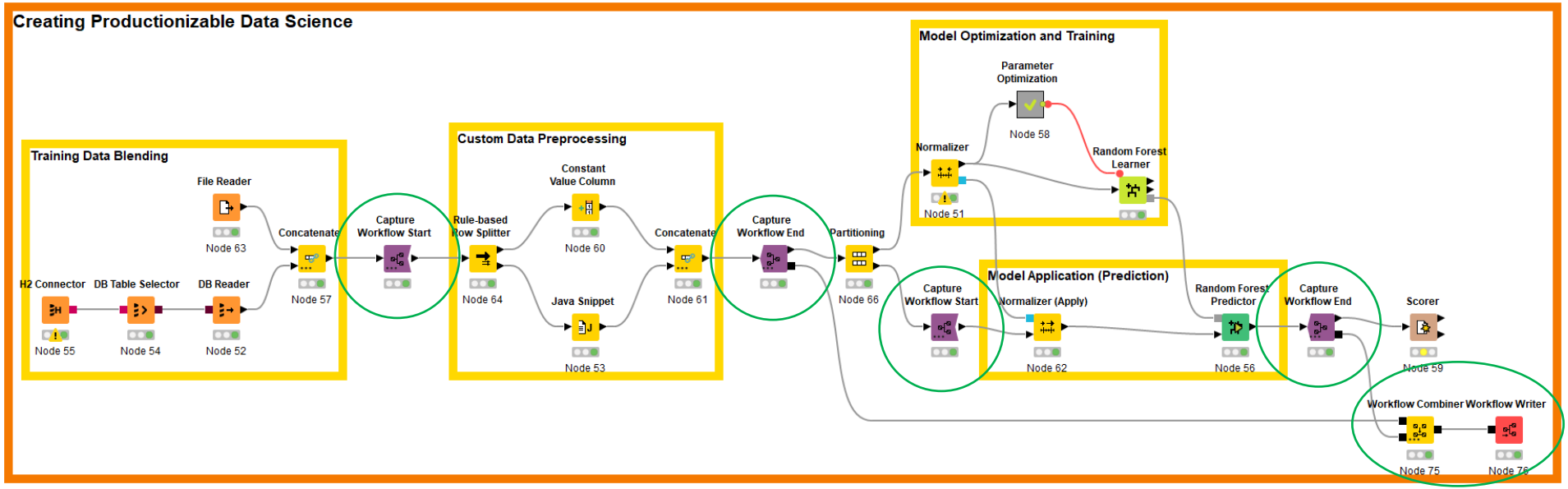
Creating Prediction Model



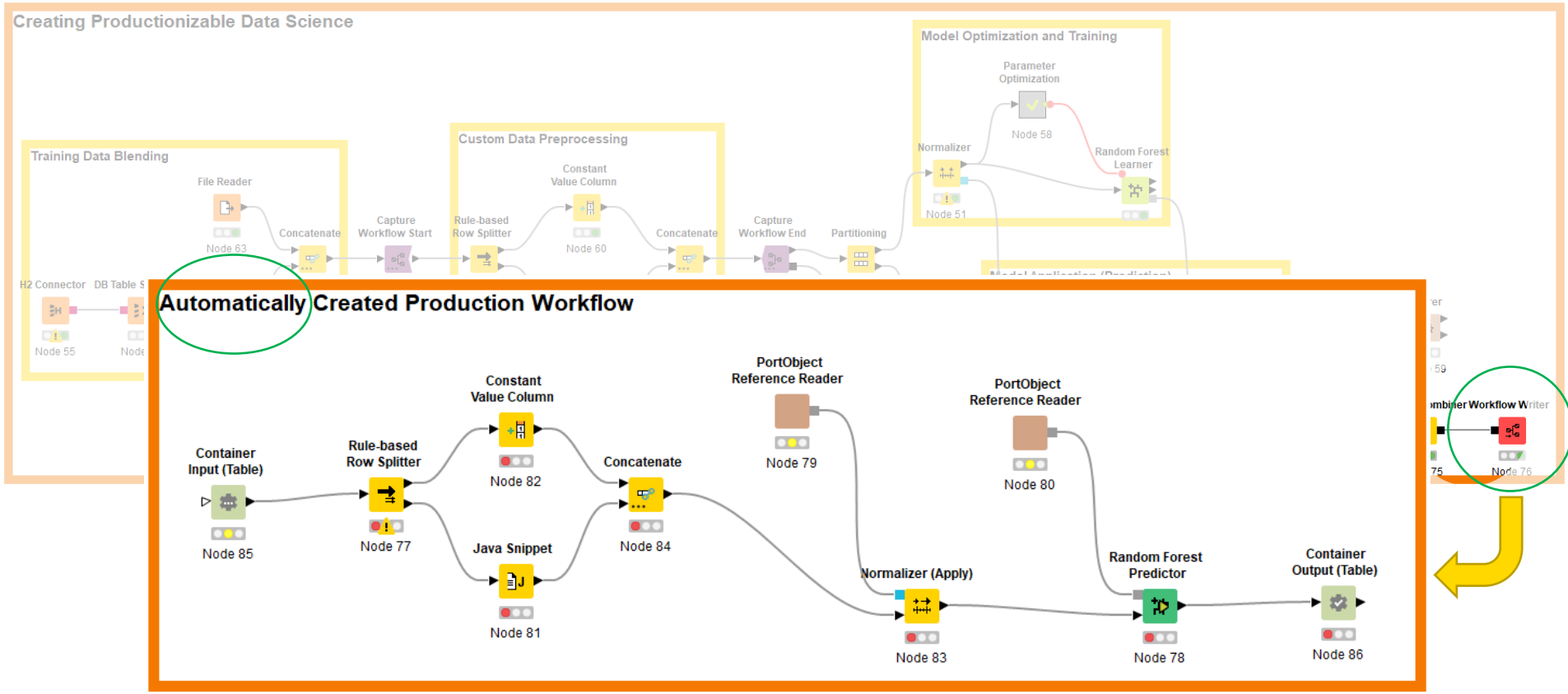
Workflow in Production



Capture and Combine Parts of a Workflow

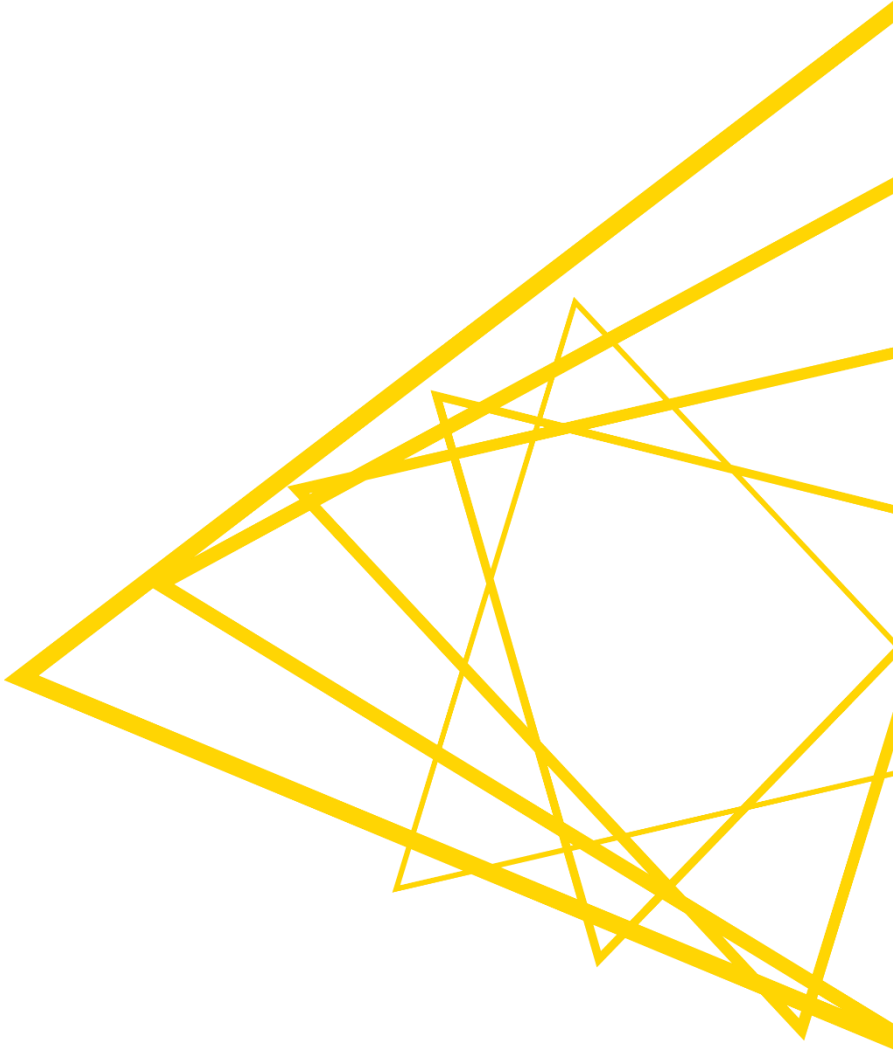


Write Production Workflow from Captured Parts

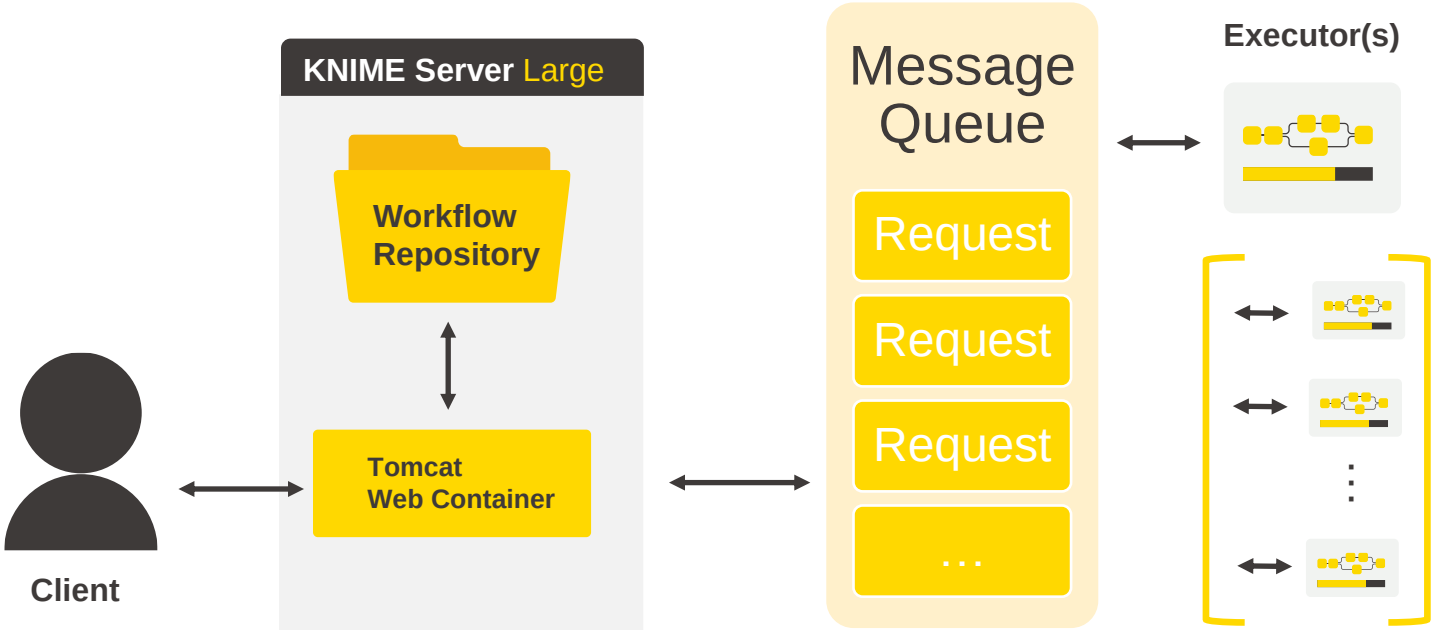


KNIME Server Administration

Architecture



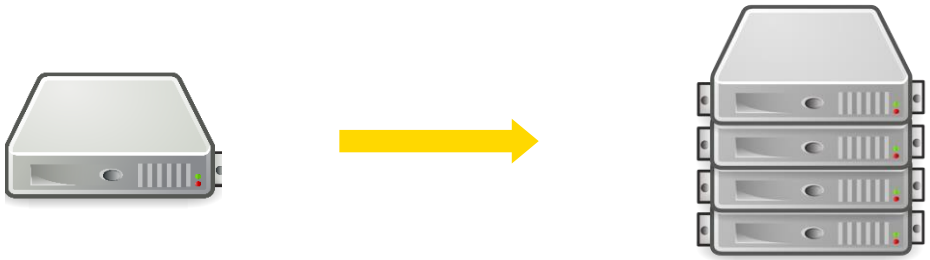
Central Components of KNIME Server



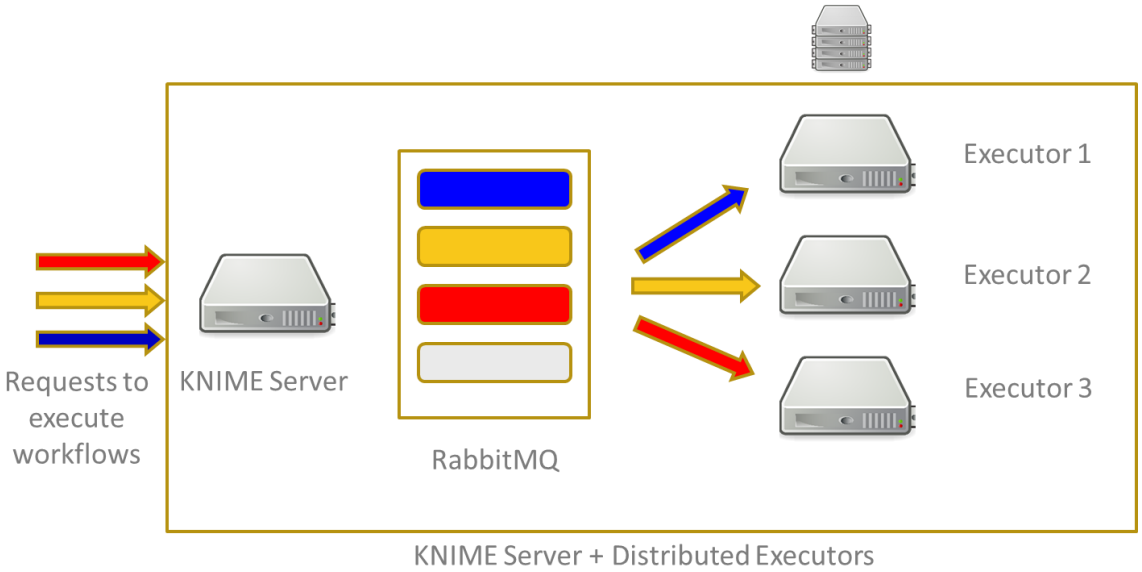
KNIME Server – Scaling Options

- KNIME Server supports ‘Scale Up’

- KNIME Executors allows ‘Scale Out’



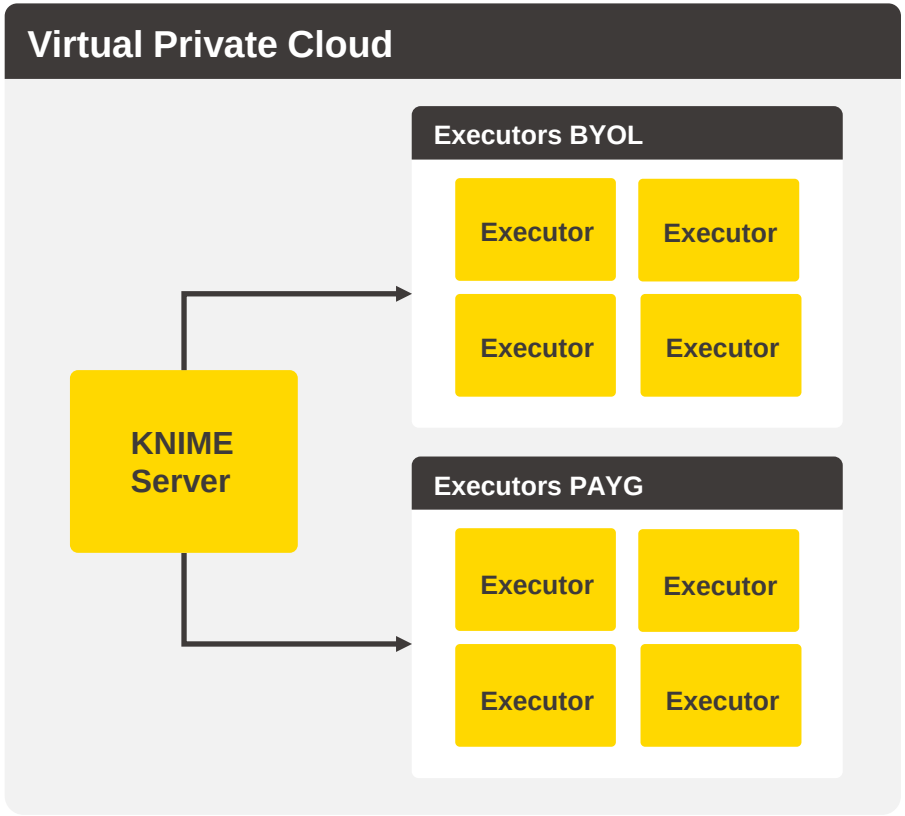
Scaling Out - KNIME Executors - *KNIME Server Large*



- Run multiple executors on independent hardware
- The distributed Executors rely on the KNIME REST interface and on a message queueing system called RabbitMQ

Flexible Cloud deployments to meet computing needs

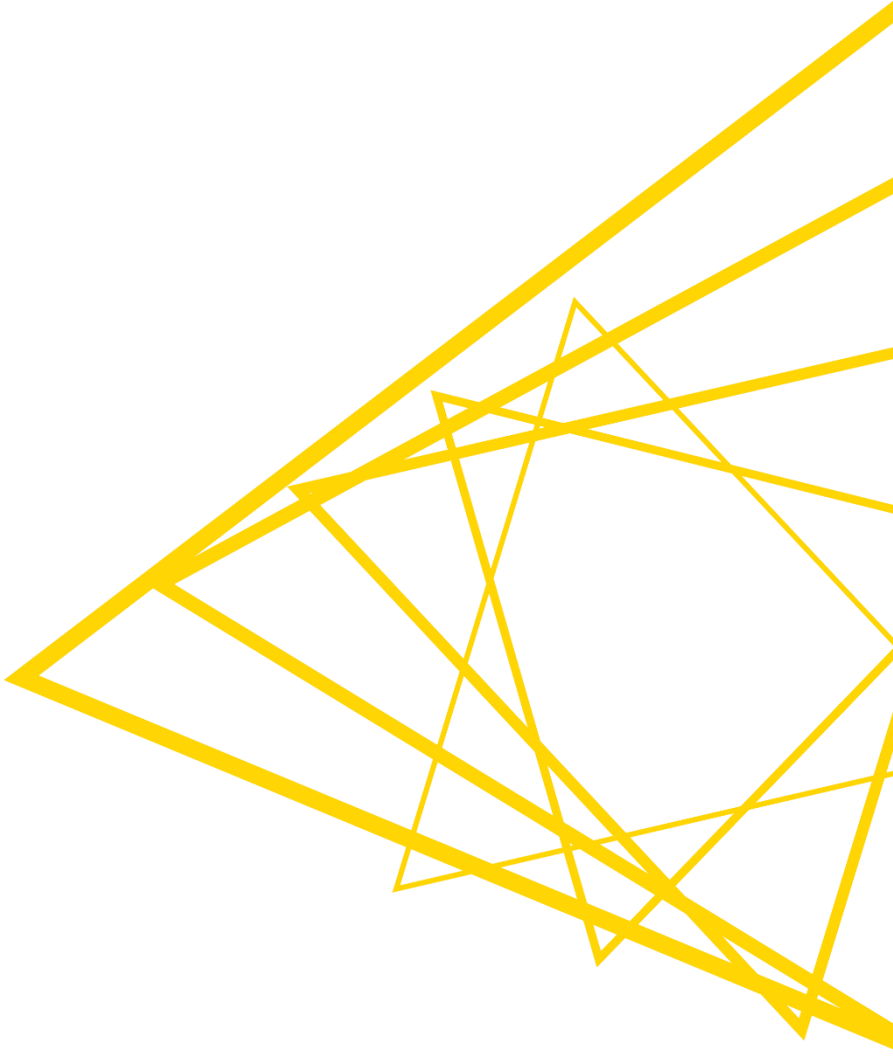
Mixed Cloud Usage



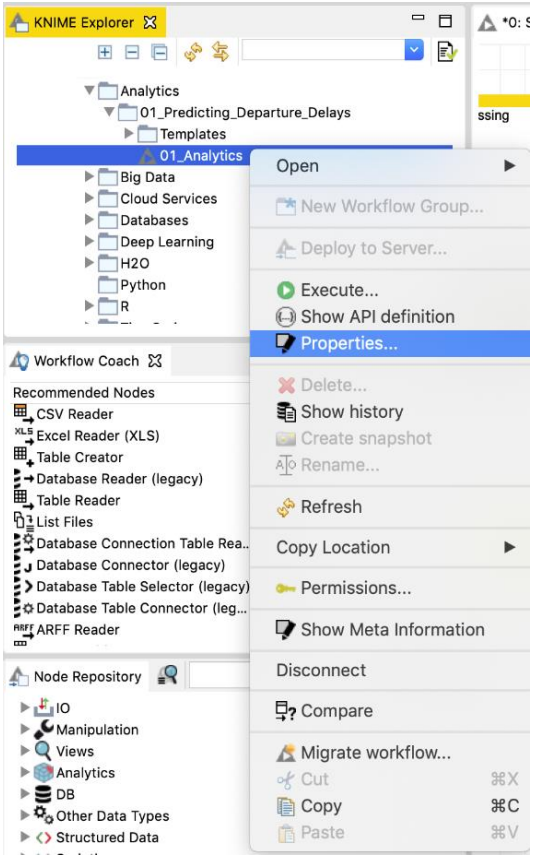
Features:

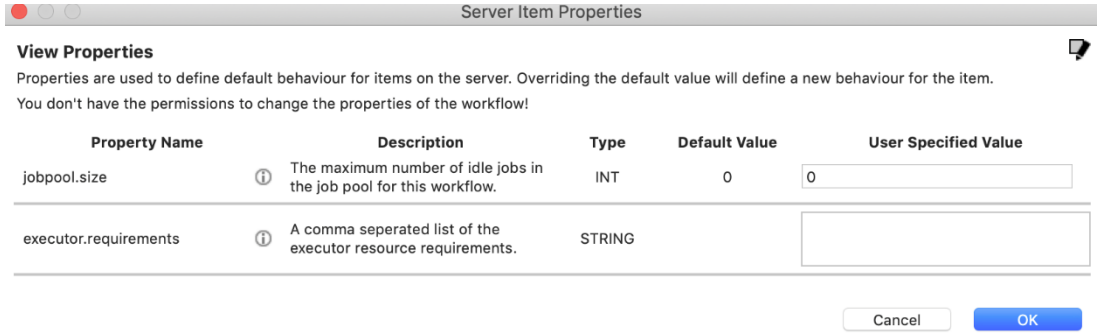
- Supplement traditionally licensed Executors with Pay-as-you-Go (PAYG) model
- Meet periodic demand peaks
- Fulfill need for speciality hardware (e.g. GPU‘s)
- Meet budgeting needs

Properties



Set Properties

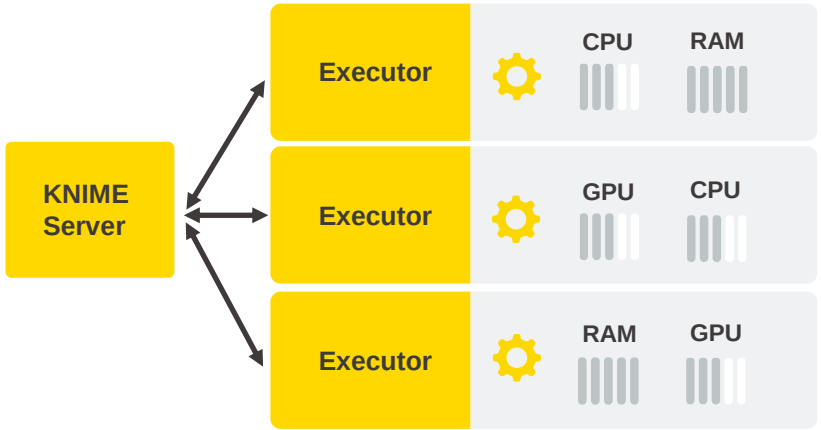




Job Pools

- Allows frequently executed workflows to stay in memory
- Advantage: Eliminates the overhead of loading the workflow in an Executor for each execution
- Useful when job loading time is large compared to execution time
- A job pool can be enabled by setting a property on the workflow that should be pooled

Workflow Pinning



Features:

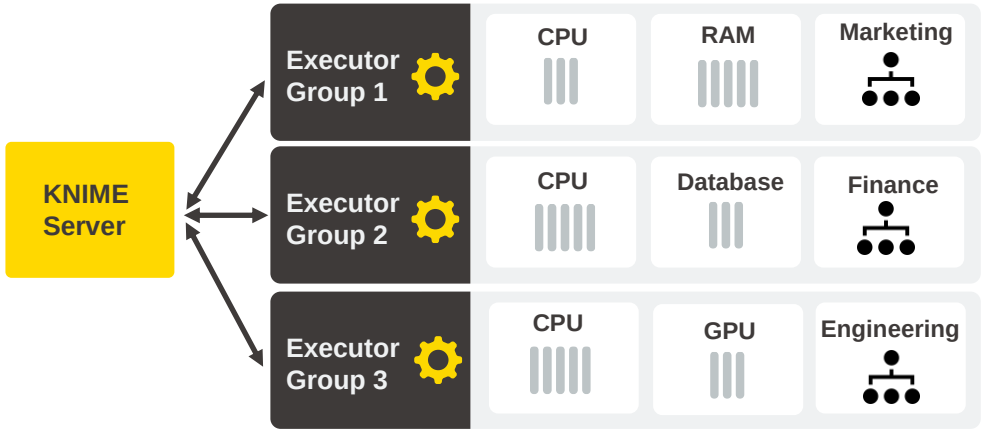
- Match workflow needs to Executor capabilities
- Partition compute resources by capability, department, usage, …
- Workflow needs determined by workflow publisher

Workflow Pinning

- Workflow pinning is useful when using multiple KNIME Executors
- For a workflow that needs certain system requirements (e.g. specific hardware (like GPU), extensions, or system environments (like Linux) it is possible to define such Executor requirements
- Only Executors that fulfill the requirements will accept and execute the workflow job
- The system admin of the Executors must specify a property for each executor separately
- The requirements can be defined by setting a property on a workflow, which is a simple comma-separated list of user defined values.

Executor Groups

Supporting Executors in the Enterprise



Features:

- Logical groupings of Executors
- Match users/groups to Executor Groups
- Partition compute resources by groups, department, …
- Partitioning managed by Server administrators

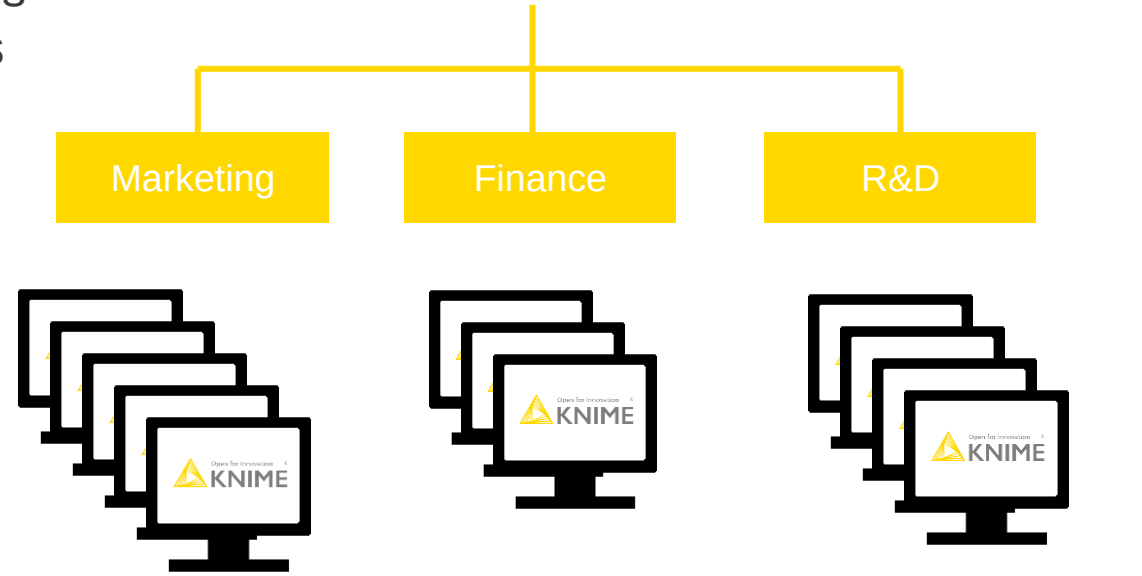
KNIME Server Management Services

About Managing Preferences

- Management (Client Preferences) makes IT operations easier by centrally managing KNIME Analytics Platform preferences
- **Solution:** Management (Preferences) part of the Management Tools available with KNIME Server Medium and KNIME Server Large
- **Features:**
  - Easier IT Operations.
  - Manage Analytics Platform preferences centrally
  - Include dependencies – e.g. driver files.
  - Deliver updates to configurations automatically
- Simplifies Executor setup in distributed Executor environments

Management (Client Preferences)

- Different departments/teams have different requirements
- Multiple OS deployments
  - Windows 7
  - Windows 10
  - Linux
  - macOS

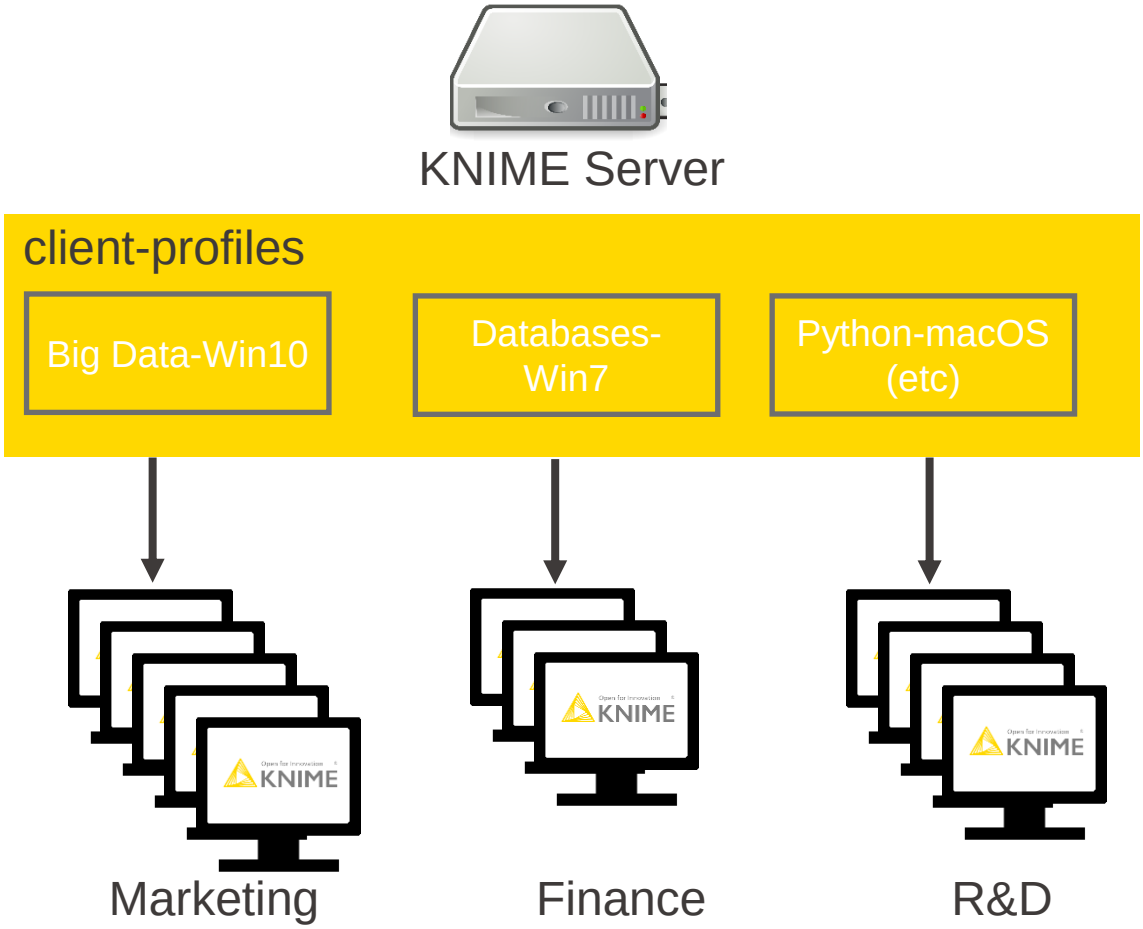


Windows 10
Hive
Spark

Windows 7
Oracle
MS Access

Linux and macOS
Python
R

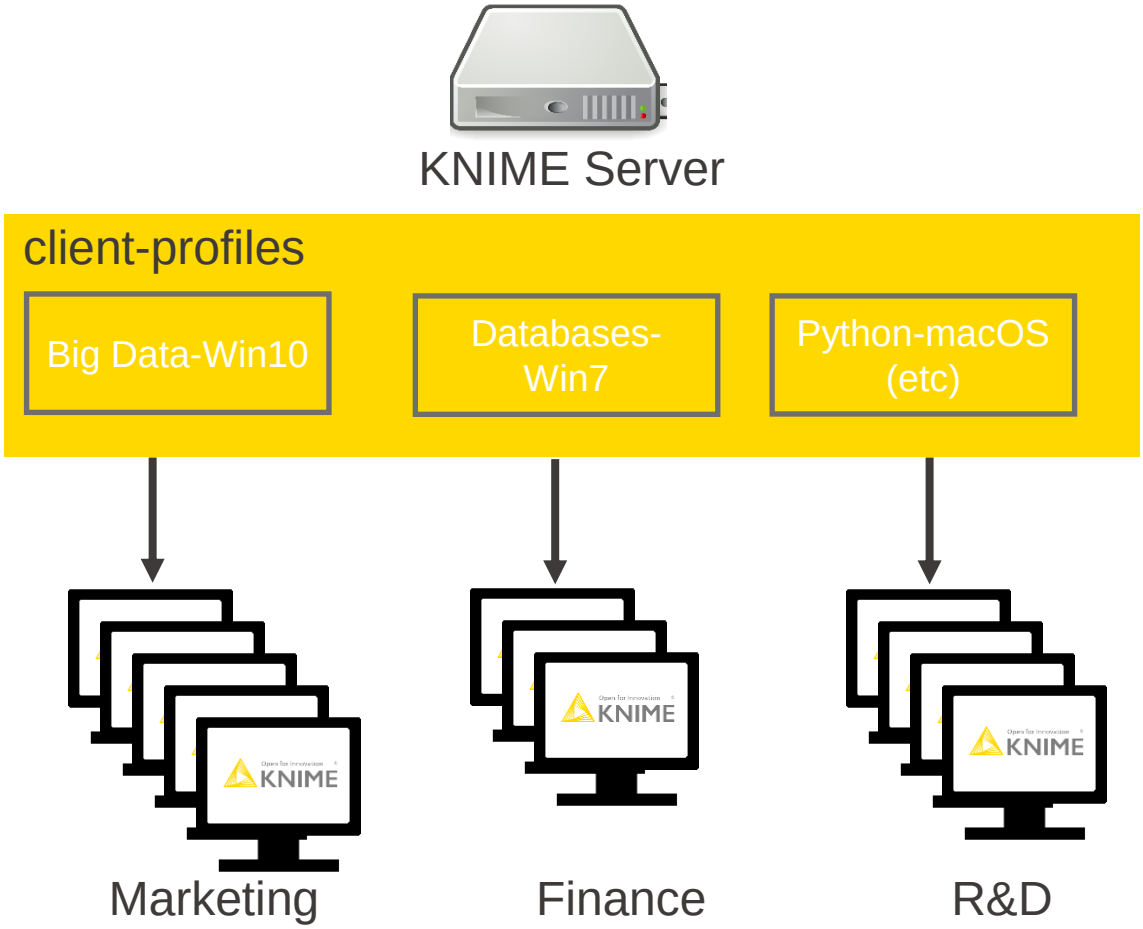
Management (Client Preferences)



- Client-profiles
  - Python-Linux
  - Python-macOS
  - R-Linux
  - Databases-Win7
  - Big Data-Win10
  - **Executor**

Profiles can include: Preferences, drivers, and more

Management (Client Preferences)



Client-profiles

- Default
  - default.epf
- Python-Linux
  - python.epf
  - py-linux.sh
- Python-macOS
  - python-mac.epf
  - py-mac.sh
- R-macOS
  - R-mac.epf
- Databases
  - db.epf
  - oracle.jdbc
  - msaccess.jdbc
- Big Data
  - bigdata.epf

Management (Client Preferences)

1. Knime.ini

By adding lines to the knime.ini (file available in the same directory as the KNIME Analytics Platform executable)

On application startup of KNIME Analytics Platform, KNIME Server is queried for the specified preference profiles. Preferences are applied before finishing startup

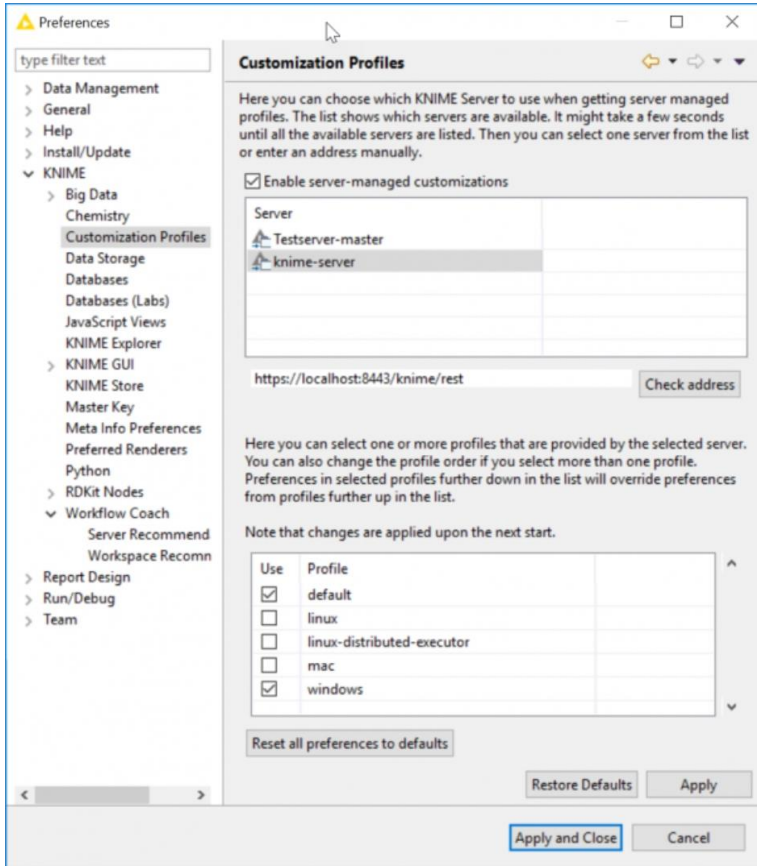
knime.ini

```
1  -vm
2  plugins/org.knime.binary.jre.win32.x86_64_1.8.0.152-01/jre/bin
3  -profileLocation
4  https://localhost:8443/knime/rest/v4/profiles/contents
5  -profileList
6  default,windows
```

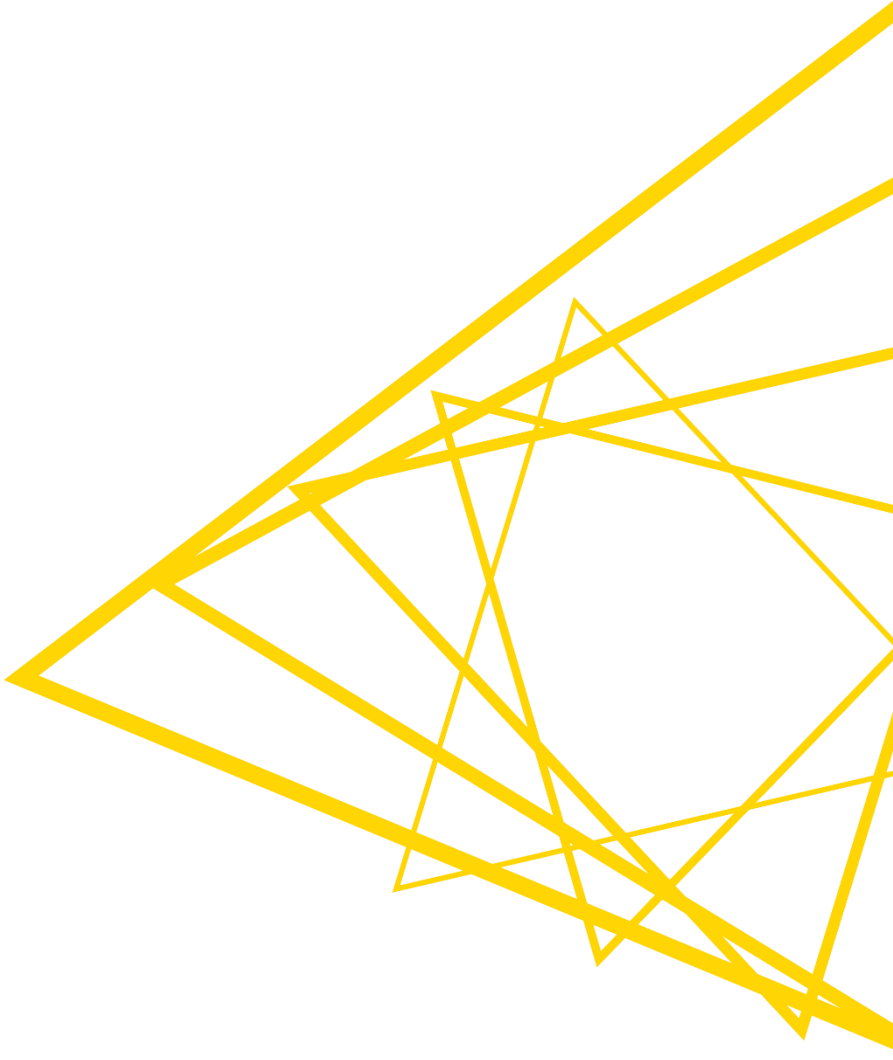
Management (Client Preferences)

2. Manually set a single preference

Allows users to browse available profiles and add them on demand.

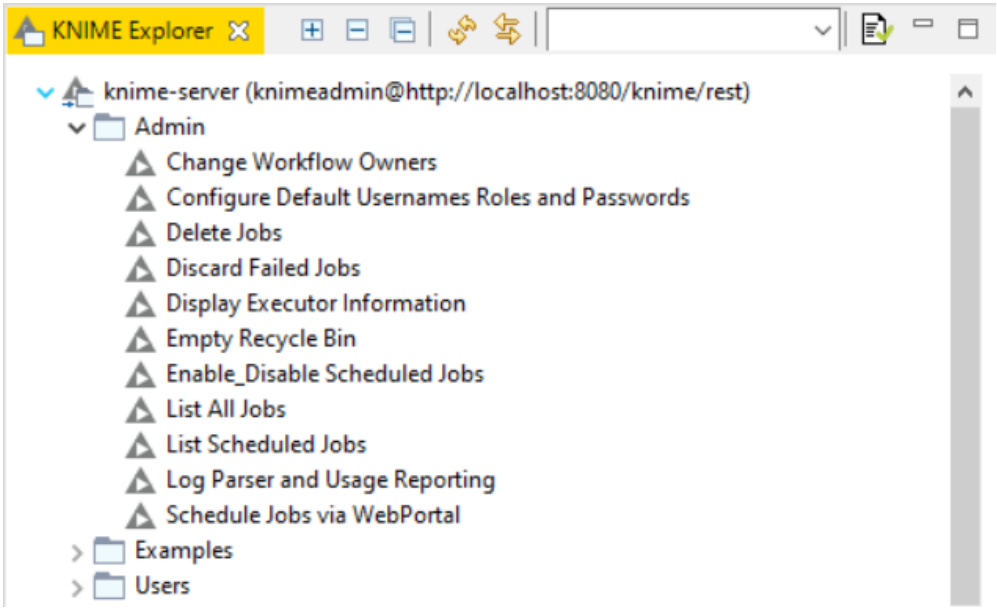


KNIME Server Maintenance

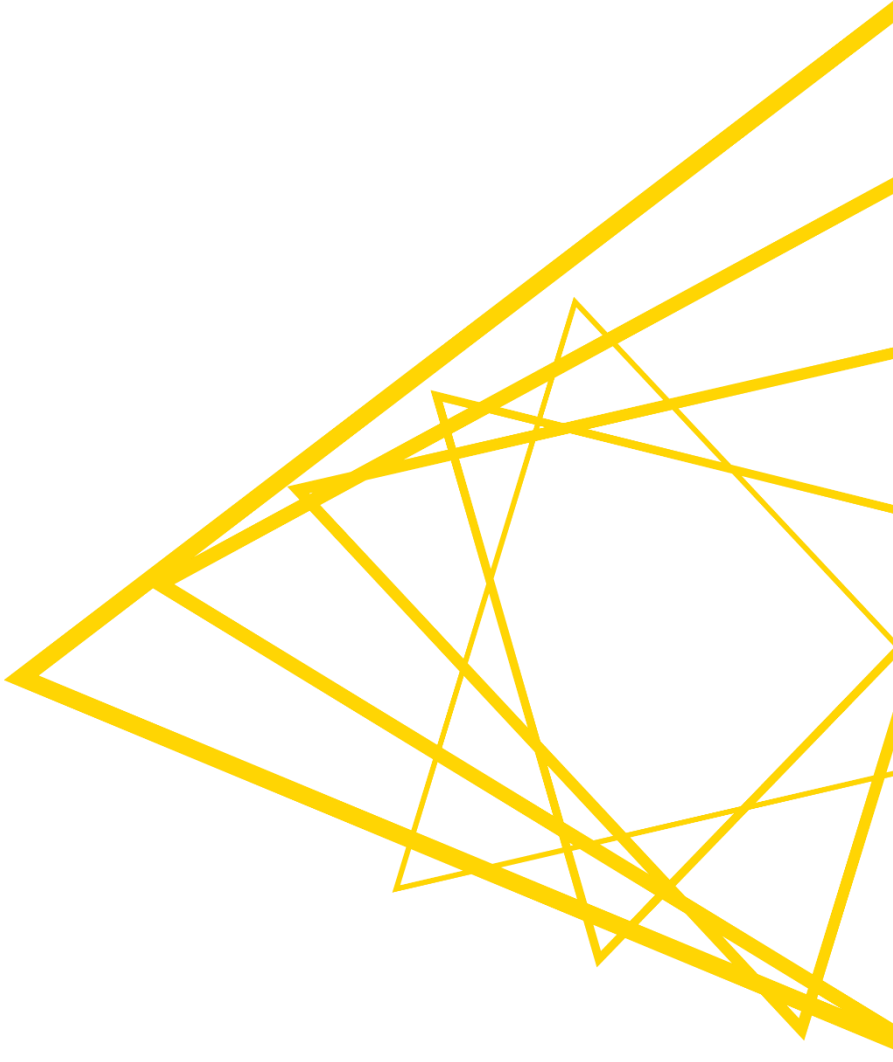


Admin Workflows

- KNIME Server comes with a set of admin workflows
- Admin workflows make use of KNIME Server REST API
- Allows automated KNIME Server maintenance



Review and Exercises



Explore KNIME Server REST API (1/2)

- Access KNIME Server REST API documentation pages: http://3.123.122.13/knime/rest/doc/index.html
- Explore different endpoints

Explore KNIME Server REST API (2/2)

Path to the workflow: *Examples* → *REST* → *Predict Results Using REST API*

- Right click menu → Show API Definition
- Explore Execution Endpoint: GET Request
- Try out and execute from browser

Integrated Deployment (1/2)

Path to the workflow: *Exercises → 05. Modelling Workflow with Integrated Deployment*

- Capture workflow parts using Capture Workflow Start & Capture Workflow End nodes for
  - Data Preparation
  - Scoring
- Combine captures using Workflow Combiner node
- Write combined deployment workflow relative to the modeling workflow

Integrated Deployment (2/2)

- Deploy this workflow to your user folder on KNIME Server
- Execute workflow on KNIME Server
- Inspect resulting deployment workflow

Config Details – Access KNIME Server

- KNIME Server Address: http://3.123.122.13
- KNIME WebPortal Address: http://3.123.122.13/knime

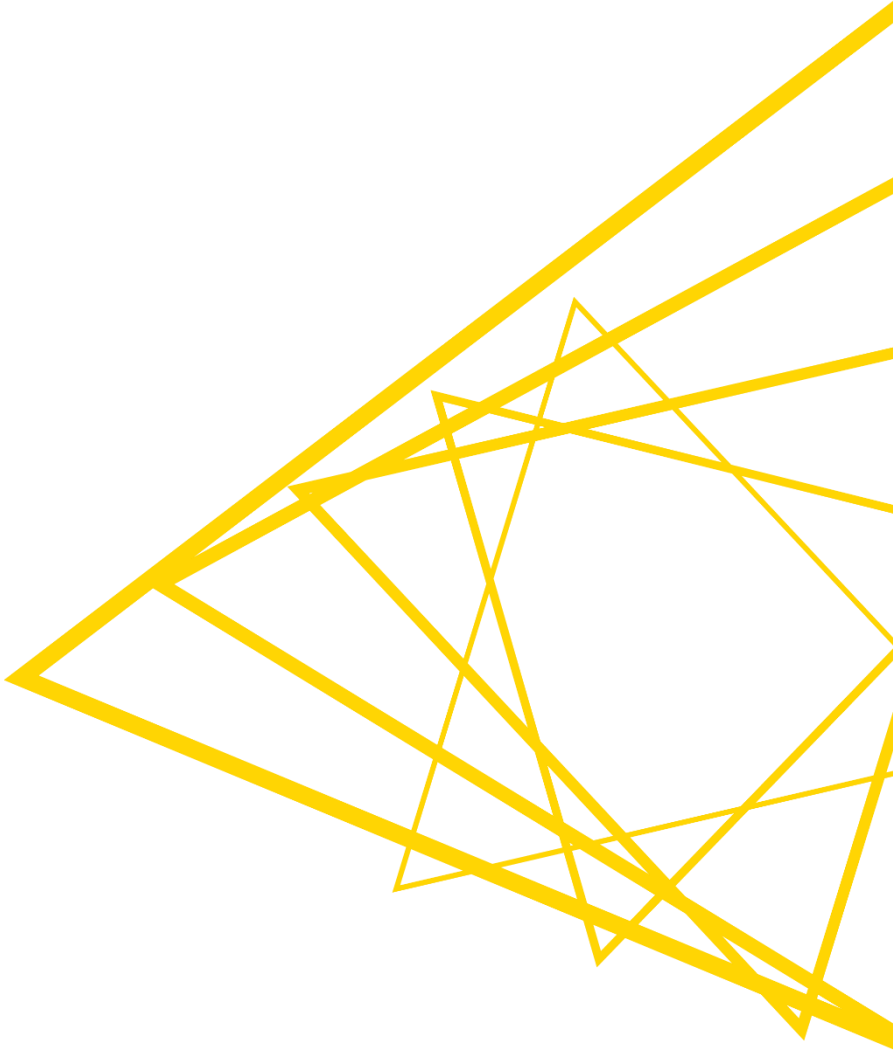
Login credential: *firstname.lastname*

Password: *knime*

Additional Info

- We will keep KNIME Server up and running for an additional week to let you play around a little bit more with it
- Interested in a trial license? Just send me an email at

Q&A



Open for Innovation
KNIME

Thank You!