Open for Innovation
KNIME

KNIME Explorer User Guide

KNIME AG, Zurich, Switzerland
Version 4.7 (last updated on 2018-11-06)

Table of Contents

Introduction

KNIME Explorer is part of the open source KNIME Analytics Platform application. It allows you to browse your workflows and to act upon them, for example through the context menu. You can look at the workflow projects that are stored in your current workspace and additionally look at multiple workflow repositories at the same time thus enabling you to share workflows and collaborate with colleagues using shared resources.

In KNIME Explorer you can "mount" the workflow repositories you want to work with. You can mount multiple repositories at the same time allowing you to work on workflows from different repositories simultaneously and to copy or move workflows from one repository to another.

By default only your current workspace is visible (mounted) in Explorer and depending on the product licenses you own, you can add TeamSpace repositories, which contain workflows stored in a local folder in the file system, or ServerSpace repositories, showing workflows stored on a KNIME Server. Depending on the type of the repository the functionality that is available may differ (for example workflows on a Server cannot be opened and directly modified).

In this guide the functionality that comes with KNIME Explorer is shown.

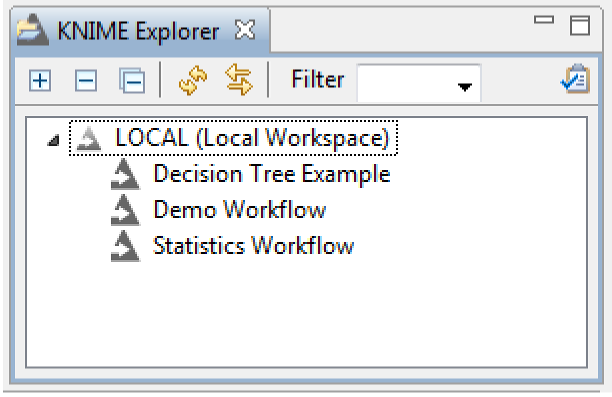
Installation

The KNIME Explorer view is already part of KNIME Analytics Platform and no additional installation steps are required. In order to open this view, select from the "**View**" menu the "**KNIME Explorer**" item. Only one KNIME Explorer view can be open at a time.

Common features

KNIME Explorer View

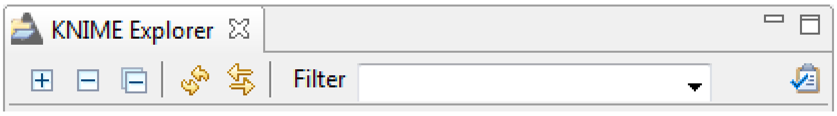
To open the KNIME Explorer view, click the "**View**" menu and select "**KNIME Explorer**". Explorer opens and shows the registered KNIME work locations. Alternatively, you can select "**View**" → "**Reset Perspective**" to restore the default KNIME view.



In order to add more content to the view, click the "configuration" icon ☑. You will be taken to the KNIME Explorer preferences page, which will be explained in the Enterprise features (TeamSpace and Server) section.

Explorer toolbar

At the top of the view are several GUI elements arranged in a toolbar:

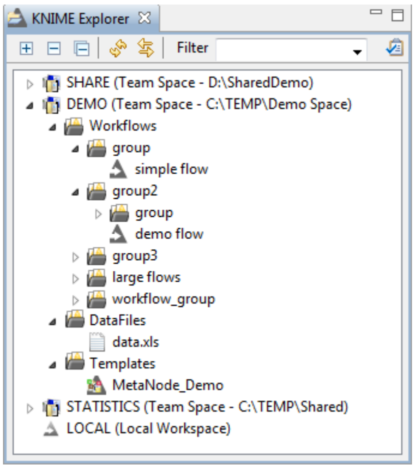


⊞ ⊟ ⊟	The (+) expands the selected element showing its content, while (-) collapses the element, hiding its children. The ⊟ collapses all elements in the view showing only the top level elements.
⟳	Refreshes the view, in case it is out-of-sync with the underlying file system.
⇆	If a workflow located in the Team Space is shown in an editor, this workflow is selected in the Team Space view.

Filter	If you add text to the field and press Enter, the Explorer will filter to items that contain the text in their name or are in a group containing the text in its name.
📋	Opens the Explorer preference page, allowing you to select the content displayed in the view or to add/remove mount points.

KNIME Explorer content

In addition to the repository entries we have discovered so far, there are four additional types of content that can be seen in KNIME Explorer. These are described in the table below:



KNIME Explorer

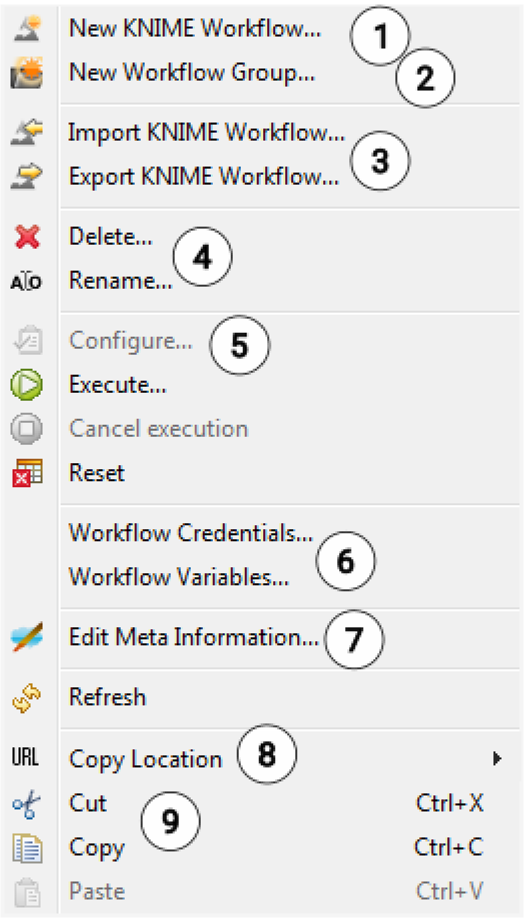
Filter

```
▷ SHARE (Team Space - D:\SharedDemo)
◢ DEMO (Team Space - C:\TEMP\Demo Space)
  ◢ Workflows
    ◢ group
        simple flow
    ◢ group2
      ▷ group
        demo flow
    ▷ group3
    ▷ large flows
    ▷ workflow_group
  ◢ DataFiles
      data.xls
  ◢ Templates
      MetaNode_Demo
▷ STATISTICS (Team Space - C:\TEMP\Shared)
  LOCAL (Local Workspace)
```

▲	Workflow	A collection of nodes used to analyze data in KNIME
📁	Workflow Group	A folder within the Explorer which can be used to store workflows, data files and Metanode Templates.
📄	Data File	Data files can be dragged to a workflow editor or a workflow group. File reader nodes can read data files stored in TeamSpace or KNIME Server (see section below).
▲	Metanode Templates	Metanodes containing a pre-configured workflow fragment can be added to both TeamSpace and ServerSpace. Once added to a workflow, they can be easily updated if the original Metanode in the repository changes.

Context menu

If you right-click in the view or on an item of the view, the following menu is shown. Several menu items are disabled, depending on whether they are applicable to the selected item or not.



1. Creates a new and empty workflow, places it in the selected workflow group (or folder) and opens it in a workflow editor.
2. Creates a workflow group.
3. Opens the workflow import or export wizard
4. Deletes or renames the selected item. If a workflow is currently edited by a user, or a workflow group contains an open workflow, it is locked and cannot be renamed nor deleted.
5. Opens the configuration dialog on the selected workflow or executes the entire workflow. Only enabled on workflows opened in an editor.
6. Opens the corresponding dialog allowing you to define workflow variables or credentials on the selected workflow.
7. Opens the meta information editor allowing entering information and a description associated with the selected workflow or workflow group.
8. Copies the URL (absolute or mount-point-relative) or local path of the selected item to the clipboard, i.e. for direct access to this source via system explorer or a terminal.
9. Copy/Cut and paste allow quick copy/move operations with hotkeys (Ctrl-c/x and Ctrl-v).

Explorer operations

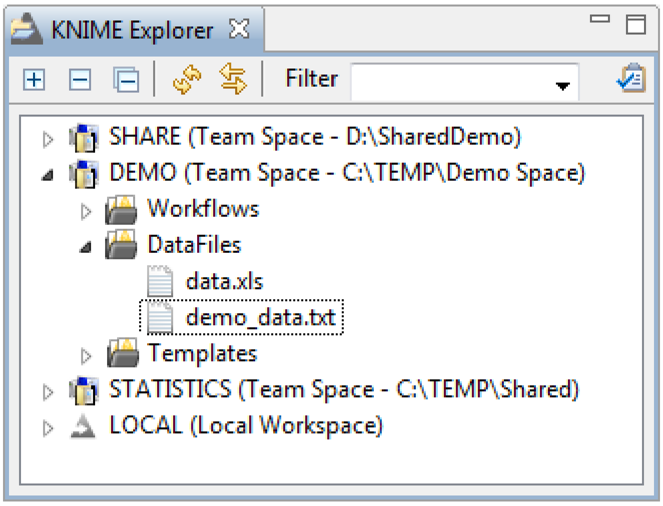
Move	To move an item, simply drag it and drop it to the desired location. You can move within the same mount point or move items between different shared resources.
Copy	Copying an item is the same process than moving it - just keep the "Ctrl"-key pressed during the drag & drop step. A little plus-sign next to the mouse cursor indicates the copy operation. The same restrictions with respect to what can be copied where apply. Additionally, Ctrl-c and Ctrl-v can be use to copy workflows between repositories.
Node Creation	If a data file of a supported type is dropped into the workflow editor, KNIME will add the data to your workflow using the appropriate file reading node.
Metanode Template creation	You can save a Metanode Template to your TeamSpace or ServerSpace repositories for later re-use. To do this, simply right-click on any Metanode and select "**Save as Metanode Template…**". The resulting dialog will let you choose a destination for your new Template. (More details can be found in the corresponding section below.)
Metanode Template usage	To use a Metanode stored in TeamSpace or on Server, drag&drop it in the workflow editor. A linked Metanode is added to the workbench. This instance is updated either manually through the context menu or when the workflow is loaded. The Metanode can also be unlinked from its original location, which makes it editable in the workflow directly.

URL-Scheme

References to files located in a remote repository use a URL. Both TeamSpace and ServerSpace features define a new URL scheme (`knime`), which denotes a resolution to KNIME Explorer. If you use the `knime` URL scheme, it must refer to the same mount point regardless of the operating system or the mounted content. When a data file is dropped into a KNIME workflow, the instantiated Reader is automatically configured to read from the `knime`-URL.

The general syntax of a URL referencing a data files in a KNIME repository is:

```
knime://<mount-ID>/<path-in-repository>/<filename>
```



SHARE (Team Space - D:\SharedDemo)
DEMO (Team Space - C:\TEMP\Demo Space)
Workflows
DataFiles
data.xls
demo_data.txt
Templates
STATISTICS (Team Space - C:\TEMP\Shared)
LOCAL (Local Workspace)

The scheme (the first element in the URL) must always be `knime:`. After the first slash, specify the mount ID of the content you want to read from. Subsequently, the path to the file in the repository including the filename is used to locate the file of interest. (To read from the selected file in the Explorer view shown above, the correct URL would be: `knime://DEMO/DataFiles/demo_data.txt`)

As long as you and your colleagues are using the same mount ID for your shared repository, you will be able to easily share items in this repository. Workflows that are used on different systems will always be able to reference the original data, enabling data sharing that is independent of the operating system KNIME Analytics Platform is running on.

KNIME supports three *relative URL* types (as opposed to absolute URLs), which is useful when you move workflows and data files or Metanode templates:

1. The *mountpoint-relative* URL: `knime://knime.mountpoint/<path-in-repository>/<filename>` resolves the filename by looking for the file in the same repository (mount point) the workflow is located in. This would be useful when you move the workflow together with its data files or Metanode Templates to a new repository - you do not have to adopt the URLs to the new location.
2. The *workflow-relative* URL: `knime//knime.workflow/../<path>/<filename>` resolves the path (that should contain `..` to refer to the parent folder) by starting at the workflow’s location. This could be useful if you store your data files or Metanode Templates in directories "parallel" to (on the same folder level than) your workflow folder. You should maintain the same folder structure when you copy or move flows with workflow-relative URLs.
3. The *node-relative* URL: `knime//knime.node/../<path>/<filename>` resolves the path (that should contain `..` to refer to the parent folder) by starting at the current node’s location. This could be useful if you store your data files inside a Metanode that you want to distribute to other users.

4. The *absolute* URL:
   `knime://<mount-ID>/<path>/<filename>` is the choice if you always want to refer to this particular item. Even if the workflow is moved or copied, the URL always points to this item located in the referenced mount point with the specified path. If the workflow is uploaded to a Server, the mount-ID with the corresponding file must be available on that Server environment in order for the URL to be resolved correctly.

Enterprise features (TeamSpace and Server)

KNIME TeamSpace and KNIME Server are commercial features of the KNIME product family. KNIME TeamSpace facilitates the use of KNIME Analytics Platform in small groups by allowing you to share your workflows and data files easily between KNIME Analytics Platform applications using shared repositories that may contain workflows, data files and Metanodes. KNIME Server has some additional features. Specifically, in addition to having access to a shared repository, KNIME Server enables server-side execution of workflows, user access permissions and the ability to run workflows via KNIME WebPortal.

Prerequisites

This guide refers to features available in the latest versions of KNIME TeamSpace 3.8 and KNIME Server 4.4, which require KNIME Analytics Platform 3.3 or higher.

Installing KNIME TeamSpace features

Start KNIME Analytics Platform as the user that owns the installation folder. From the "**File**" menu, select "**Preferences**" and browse to "**Install/Update**" → "**Available Software Sites**". Check the "**KNIME Store <version> Update Site**" and click "**OK**". After that you can choose "**Install KNIME Extensions…**" from the "**File**" menu. Select "**KNIME TeamSpace**" and "**KNIME Report Batch Execution**" from the category "**KNIME TeamSpace Extensions (license required)**", click "**Next**", accept the license and finish the installation. The new feature will only be available after a restart of KNIME Analytics Platform.

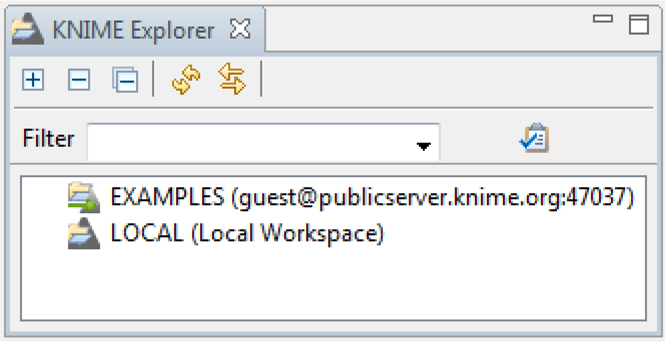
Installing KNIME Server features

Start KNIME as the user that owns the installation folder. From the "**File**" menu, select "**Preferences**" and browse to "**Install/Update**" → "**Available Software Sites**". Check the "**KNIME Analytics Platform <version> Update Site**" and click "**OK**". After that you can choose "**Install KNIME Extensions…**" from the "**File**" menu. Select "**KNIME ServerSpace**" from the "**KNIME Server Connector**" category and click "**Next**", accept the license terms and finish the installation. The new feature will only be available after a restart of KNIME Analytics Platform.

Adding repositories

By default only the LOCAL and the EXAMPLES workspaces are displayed in the Explorer view.

The LOCAL workspace shows all workflows and groups of the current workspace. The EXAMPLES workspace provides access to the KNIME Public Server with its example workflows.



EXAMPLES (guest@publicserver.knime.org:47037)
LOCAL (Local Workspace)

If you want to add a workflow repository to the view, you need to register a new mount point.

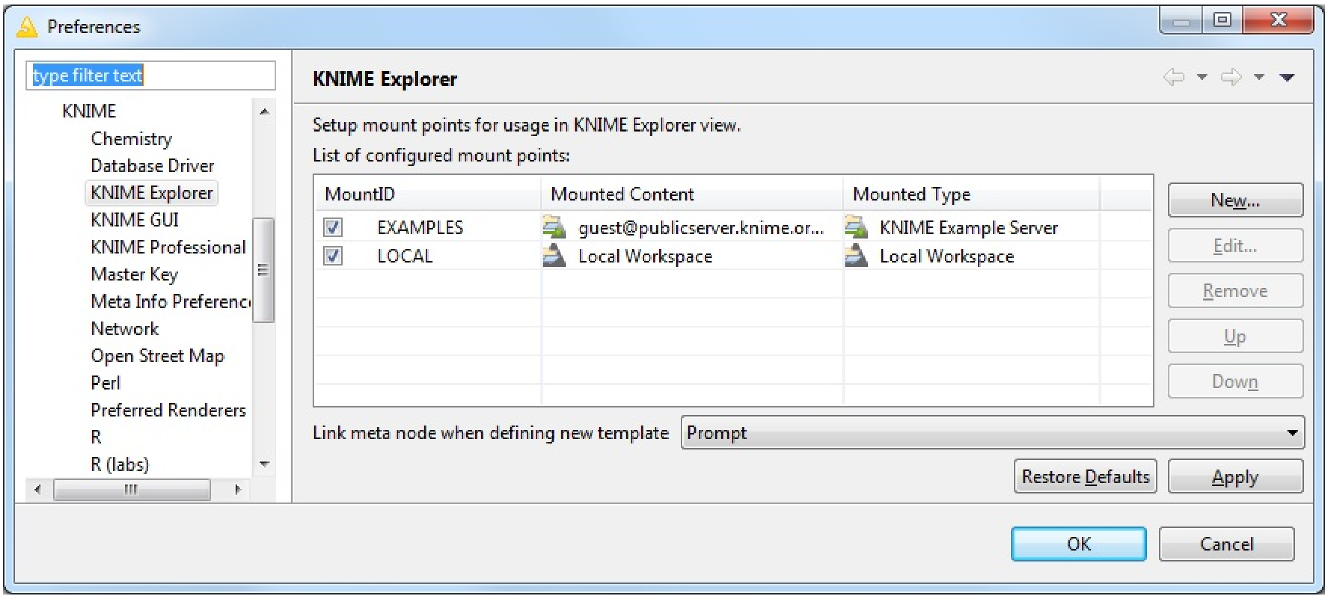
Mount points

Workflow repositories that should be accessible from KNIME Analytics Platform are called mount points. Mount points can be displayed in the KNIME Explorer view.

Each mount point consists of the **location** of the workflow repository (that is either the path to the folder or the address of the server) and a "**mount ID**".

The mount ID is used to reference files or workflows located under the new mount point.

Especially if workflows are shared and files are accessed throughout different mount points, it is important to associate the correct shared folder with the mount ID. If nodes in a workflow use an URL with the `knime` protocol to read from the shared space (see section below), it will always reference the same file, as long as the same shared folder is mounted. If a mount point is removed while a workflow node reads from a file stored in the mount point, the reading will fail. If a different shared folder is mounted with the same mount ID, all URL references will try to access the file in this different folder. If it does not exist, the operation fails.



KNIME Explorer

Setup mount points for usage in KNIME Explorer view.

List of configured mount points:

MountID	Mounted Content	Mounted Type
☑ EXAMPLES	guest@publicserver.knime.or...	KNIME Example Server
☑ LOCAL	Local Workspace	Local Workspace

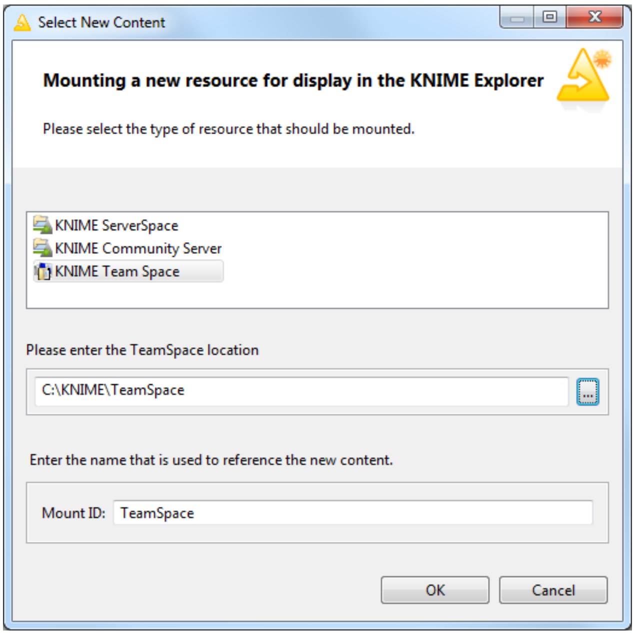
Link meta node when defining new template: Prompt

New mount points are defined in a preference page: From the "**File**" menu, select "**Preferences**", then select "**KNIME Explorer**" in the category "**KNIME**".

The KNIME Explorer preference page shows all currently defined mount points. The LOCAL workspace (which is the place where the KNIME Analytics Platform base version stores the workflows) and EXAMPLES workspace are included by default.

To link a new workspace to Explorer, click "**New…**".

In the dialog that opens, select the type of mount point to add and enter the additional information as required.



Mounting a new resource for display in the KNIME Explorer

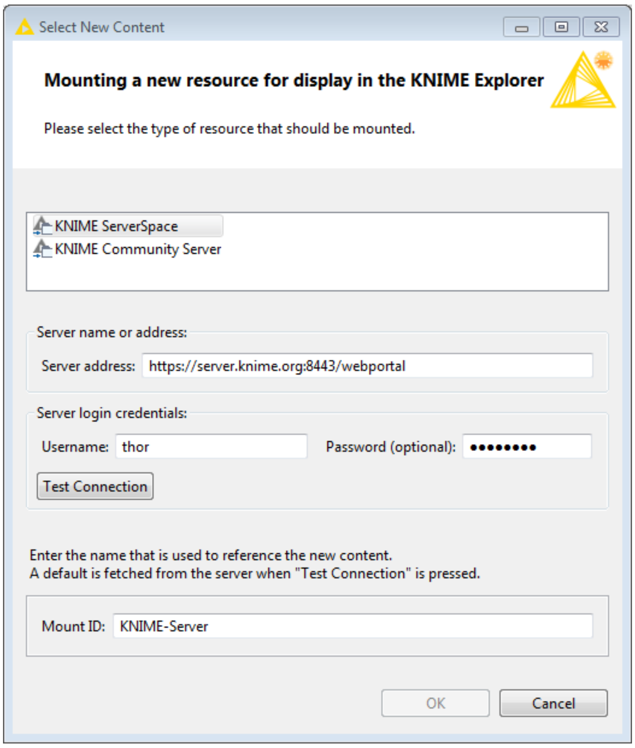
Please select the type of resource that should be mounted.

Please enter the TeamSpace location: C:\KNIME\TeamSpace

Enter the name that is used to reference the new content.

Mount ID: TeamSpace

For a new TeamSpace workspace simply select the root folder of your repository. If a default mount ID has been set for that location it is automatically entered as mount ID, otherwise the selected folder name. In case that a default ID has not been set, you will be asked to create it when creating the mount point for the first time.



Select New Content

Mounting a new resource for display in the KNIME Explorer

Please select the type of resource that should be mounted.

- KNIME ServerSpace
- KNIME Community Server

Server name or address:

Server address: https://server.knime.org:8443/webportal

Server login credentials:

Username: thor   Password (optional): ••••••••

Enter the name that is used to reference the new content.
A default is fetched from the server when "Test Connection" is pressed.

Mount ID: KNIME-Server

To add a new KNIME Server repository, select "**KNIME ServerSpace**" and enter the Server’s name or address in the appropriate field. For KNIME Servers until version 3.10 the "**Server name**" is simply the hostname (with optional port). Starting with version 4.3 you need to specify the `http` or `https` address of KNIME WebPortal, e.g. `http://hostname:8080/knime/` or similar. The protocol, hostname and port need to be adjusted. If you have a standard setup you can also omit the path part (`knime`) but it is recommended to always specify the complete address.

To test connectivity, credentials and fetch a default mount ID from Server, enter your user name and password and click the "**Test Connection**" button. You can use your own mount ID be changing the value in the field below.

To change an already existing and configured mount point, select the resource to change and click the "**Edit…**" button.

To hide configured mount points from the Explorer view without deleting them, simply uncheck them on the preference page.

After you have added all mount points needed, you can put them in order by selecting one and clicking "**Up**" or "**Down**" to move it up or down in the list. The mount points are displayed in this order in the KNIME Explorer view.

> When sharing Metanodes within a common mount point, all KNIME Analytics Platform and Server instances connected to this mount point need to use the same mount point ID.

Mount table

The mount table defined in the preference page (see above) is saved automatically when KNIME Analytics Platform is closed. When you export the preferences (menu "**File**" → "**Export**

Preferences…"), this table is exported with all other settings and if the preferences are imported into another KNIME workspace, the mount table is effectively transferred.

If nodes in the workflow reference data files (or Metanodes) in a mounted repository, the references are resolved against the mount table (see section below). This resolution is done by mount ID, so it is important to note that only IDs which are defined in the mount table can be resolved: if the mount ID is changed or disabled in the mount table, a "**file not found**" error will appear.

Licenses

TeamSpace only works with a valid license file. Create a folder named `licenses` in your KNIME Analytics Platform installation folder, if it does not exist already. Copy your license file from the KNIME TeamSpace distribution to this folder. Finally, restart KNIME Analytics Platform in order to activate your license.

Note that certain features (data files and linked Metanodes) will also become available in the local workspace, when a TeamSpace license is installed.

If you do not have a license file, or it is not working correctly, please contact KNIME by sending an email to contact@knime.com or to your dedicated KNIME support specialist. You can check your license using the KNIME License view available in the menu "**View**" → "**Other**" → "**KNIME Views**".

If you get the error message "No license found", please check the name of the license folder, it must be named `licenses`, and be located in the KNIME Analytics Platform installation folder. Finally, the license file must end with `.xml` and have read permissions for the user running KNIME Analytics Platform.

Licensing for KNIME ServerSpace is controlled on the server itself. If you do not know your username and password, please contact your local KNIME administrator.

Data files in remote repositories

Files can be stored in workflow groups together with workflows in both KNIME TeamSpace and KNIME ServerSpace repositories. Add a file to a remote repository by dragging it from a file explorer window (the "Windows Explorer" in Windows) to the desired group within the repository.

You can also copy a file with the operating system to the mounted shared folder. After you click refresh 🔄 in the TeamSpace Explorer, the file is displayed. Note that only files with a registered extension can be read from within a KNIME workflow.

Dropping files in the workflow editor

Files stored in a remote repository can be dragged to an open workflow editor. The node that is associated with the file's extension is instantiated and configured to read from the dropped file. If the node is not executable - that is, it needs more settings in its dialog - the configuration dialog will be opened. Registered file extensions and their associated reader nodes are fixed and cannot be adjusted by the user.

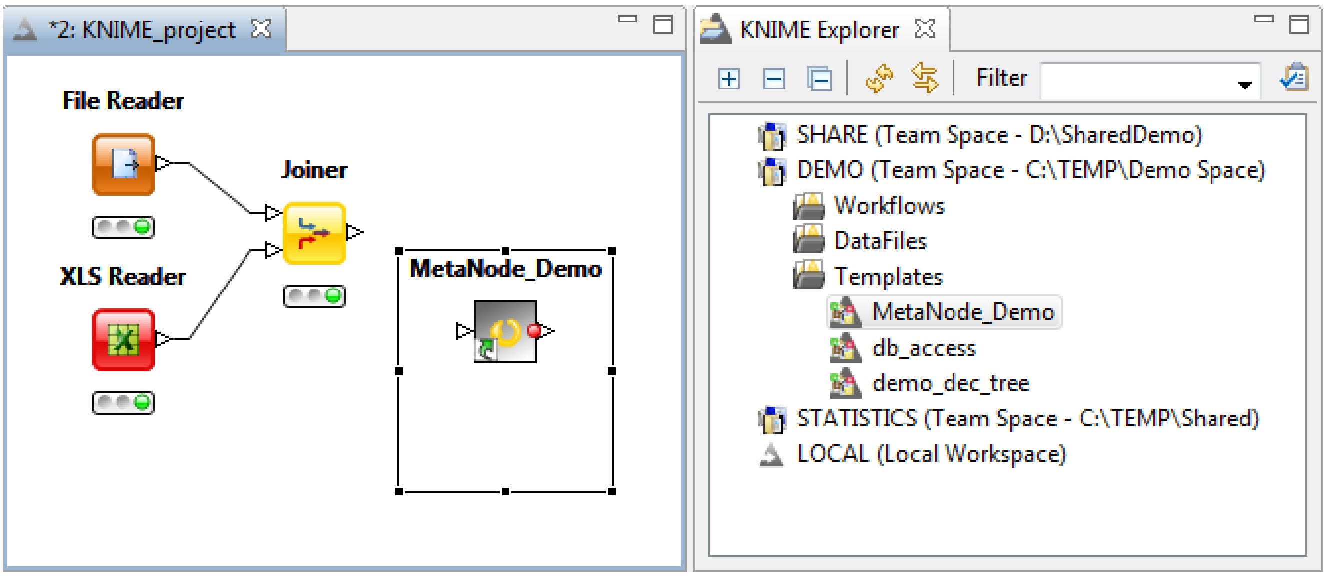
Metanode Templates in remote repositories

TeamSpace and ServerSpace repositories can be used to store reusable Metanodes. Metanodes stored in a remote repository allow complicated subroutines in KNIME Analytics Platform to be shared among your colleagues. This enables increased productivity through a reduced duplication of efforts and by allowing fixes and improvements to these subroutines to be pushed out to all users of a particular Metanode.

Metanodes stored in workflow groups in TeamSpace or on Server. They are displayed with this icon:

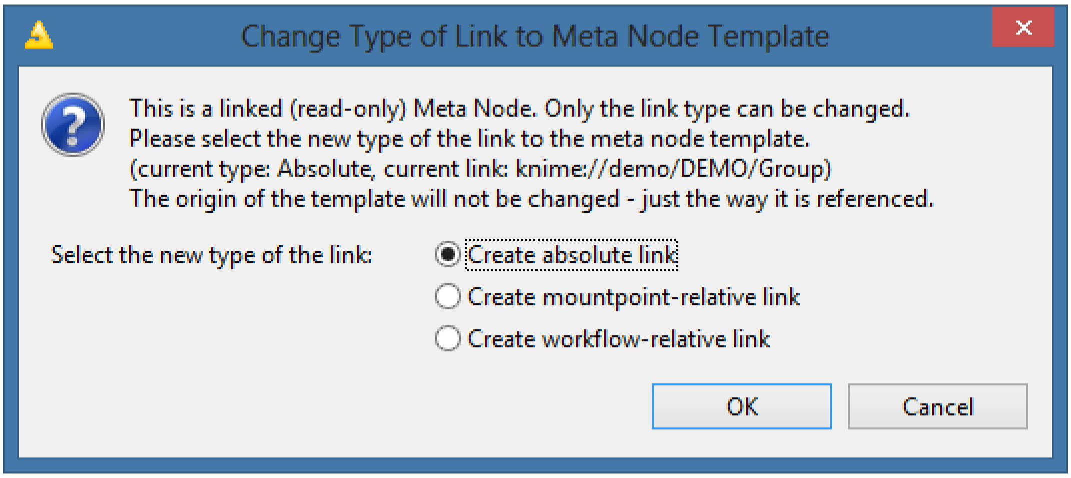
Using Metanode Templates

To use a Metanode from your remote repository, simply drag it to the workflow editor, which will then insert a linked Metanode reference. The advantage of a linked reference is that it can be used to update your working copy of the node should the original be changed. A Metanode linked to a shared repository is read-only, but can be unlinked from its Metanode Template via the Metanode context menu command: "**Disconnect Metanode Link**".



Metanodes in a workflow with an arrow in the lower left corner of the node icon are "linked"

Metanodes.



Change Type of Link to Meta Node Template

This is a linked (read-only) Meta Node. Only the link type can be changed.
Please select the new type of the link to the meta node template.
(current type: Absolute, current link: knime://demo/DEMO/Group)
The origin of the template will not be changed - just the way it is referenced.

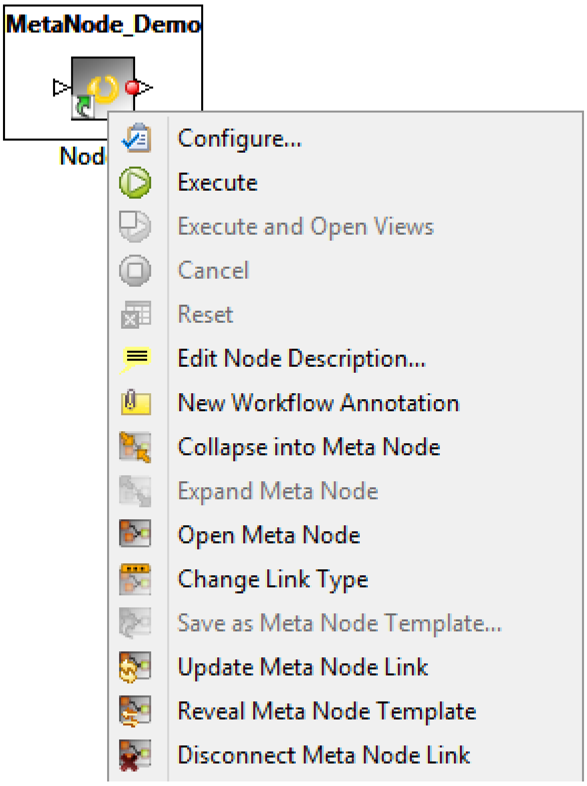
Select the new type of the link:

● Create absolute link
○ Create mountpoint-relative link
○ Create workflow-relative link

By default linked Metanodes have an absolute URL as link back to their original Template. You can change the type of the link by selecting "**Change Link Type**" from the context menu of the Metanode. In the subsequent dialog you can select one of the three types "mountpoint-relative", "workflow-relative", and "absolute" link (details see above).

Linked Metanodes

Linked Metanodes are read-only instantiations of a Metanode Template. If you open the editor for this new Metanode copy (by double-clicking it for example), the editor has a gray background, indicating that it is indeed read-only: you may open the configuration dialogs on the nodes and inspect the settings but you may not apply changes to configuration dialogs or adjust node inputs or outputs.



If you open the context menu on a Metanode and select "**Update Metanode Link**", KNIME Analytics Platform will check if the original Metanode Template has changed since the copy in the workflow was last updated. If an update is available, you may choose to overwrite your local copy. If you choose to not update, the link icon in the node will change from green to red. The red arrow serves as a reminder that the Template has changed and that the Metanode is currently not in sync with those changes.

If you select "**Reveal Metanode Template**" from the context menu, the original template of the linked Metanode is selected in Explorer, showing the actual source of the content of the Metanode.

Workflow loading with Linked Metanodes

Upon loading a workflow containing one or more linked Metanodes, KNIME Analytics Platform will ask if you would like to check for newer versions of your linked Metanodes. Choosing yes will allow you to see which nodes have available updates and give you the opportunity to implement those updates in your current workflow. Choosing No will simply flag all of your nodes as potentially outdated and allow you to proceed with your analysis normally. You will subsequently need to manually update your Metanode links when this becomes convenient.

Disconnecting Linked Metanodes

Should you need to edit your linked Metanode you will first need to disconnect it from its linked Template. From the context menu of the node select "**Disconnect Metanode Link**". The little link arrow in the node’s icon will disappear.

You can now edit the Metanode and save changes. It is a regular Metanode now. You cannot automatically check for new versions of the original template anymore. You can in turn use this modified Metanode to create a new Metanode Template or override an existing one, see section below.

Creating a Metanode Template

Any Metanode in a workflow can become a Metanode Template when it is stored in TeamSpace.

Be careful to create the subflow contained in the Metanode in a way so that it will work in other workflows and in other KNIME environments. For example, try to avoid hard coded paths to files or directories. Instead, use URLs that reference files in the remote repository using the scheme identified in the previous section. All nodes should be pre-configured to run with the expected data input this step is very important because linked Metanodes are read-only for their users.

To streamline the adoption of your Metanodes, consider adding detailed custom node descriptions. From the context menu select: "**Edit Node Description…**". Your node description should contain information regarding the purpose of your node, the configuration options that you have exposed (see next section), and details regarding expected inputs and outputs. You may also change the name of the node. This is what users see, when they connect to TeamSpace. Note that it is not advised to use non-standard characters: stick with numbers, spaces and underscores in order to avoid problems.

Finally, you may right-click on your Metanode and select "**Save as Metanode Template…**". The resulting dialog will prompt you to choose a location where to save the Metanode Template. After creating the Template in TeamSpace another dialog opens. You can now add a link from the Metanode in your workflow to the newly created Template in TeamSpace. This way you can update your local copy of the Metanode whenever the Template in TeamSpace is updated. If you choose to link your Metanode with the new Template, you can select what kind of link you want to create, either absolute, mountpoint- or workflow-relative (details see above).

Using Quickform nodes in Metanode Templates

Linked Metanodes are read-only instances spawned from a Template in a remote repository. They cannot be changed or configured by the end user. However, in order to provide some degree of flexibility in Metanode Templates it is possible to parameterize those using Quickform nodes. When the Template contains Quickform nodes the Metanode will construct a configuration dialog based off of the defined these nodes. For more information on this topic, please refer to the KNIME WebPortal documentation.

Additional TeamSpace features

Report batch executor

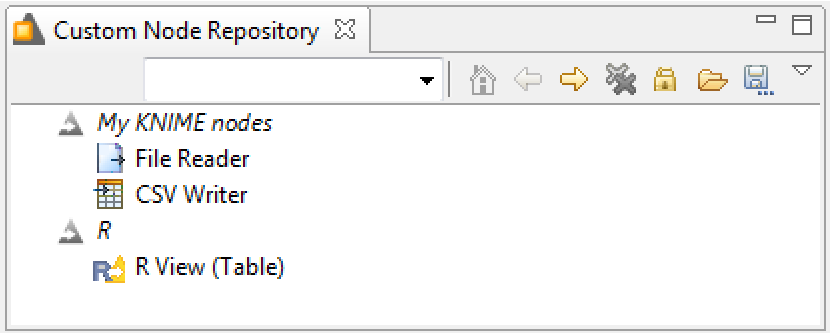
From a command line tool, it is possible to invoke KNIME Analytics Platform as a headless application:

```
./knime -application com.knime.product.reportbatch.KNIME_REPORT_BATCH_APPLICATION
```

This command will provide an overview of the functionality of this tool including all available options.

Custom node repository

KNIME TeamSpace allows you to create a customized node repository, whereby only selected nodes from the overall node collection are included. This enables administrators of larger deployments to filter (or re-order) the list of nodes available to the average end user.



To use this feature enable the corresponding view by selecting "**Other**" from the "**View**" menu and browse to the "**KNIME Views**" folder.

You can use the custom node repository to create your own collection of nodes, separate from the general node repository, which lists all available views.

Create groups and add nodes by simply dragging them in from the Node Repository view. The configuration can be exported into an `xml` file and later on be imported using the corresponding view actions.

In order to persistently save this configuration and setting it as default export it to `<knime-installation>/customNodeRepository.xml`. This file can be further modified using a system editor to, e.g. set the label associated with the view window (default is "Custom Node Repository"). If present, KNIME uses this file to define the content of the standard node repository. Users of KNIME will only be able to instantiate nodes that are contained in this customized view (though workflows containing other nodes will still load successfully.)

Workflow Difference

Introduction

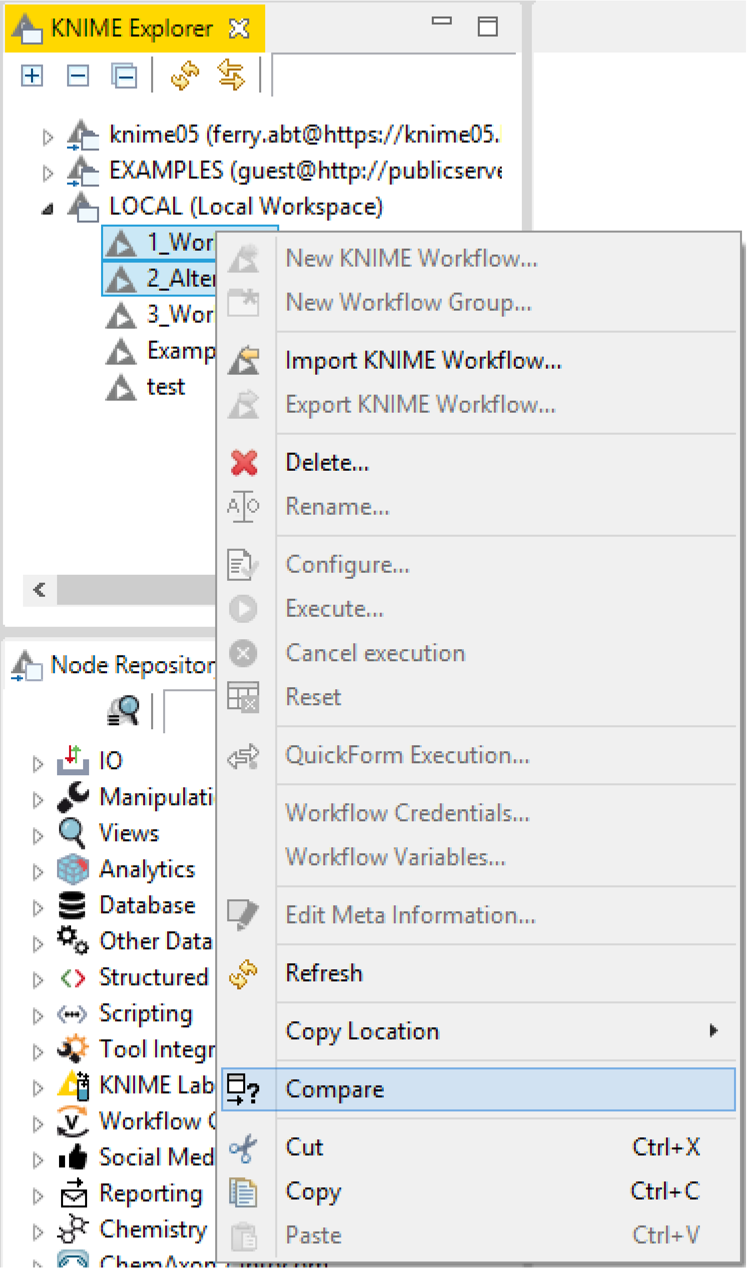
The Workflow Difference feature provides tools and views that compare workflow structures and node settings. Workflow Difference allows you to view changes in different versions of a workflow. The feature allows users to spot insertions, deletions, substitutions or similar/combined changes of nodes. The node settings comparison makes it possible to track changes in the configuration of a node.

Installing Workflow Difference

KNIME Workflow Difference is part of the KNIME Personal Productivity Tools feature which is part of KNIME Analytics Platform. In case you do not have it installed, please follow the steps below

1. Start KNIME Analytics Platform as the user that owns the installation folder.
2. Go to the "**File**" menu and select "**Install KNIME Extensions…**".
3. Select "**KNIME Personal Productivity Tools**". Click "**Next**", accept the license, and complete the installation.
4. KNIME Workflow Difference will be available as soon as you have restarted KNIME.

Workflow Comparison

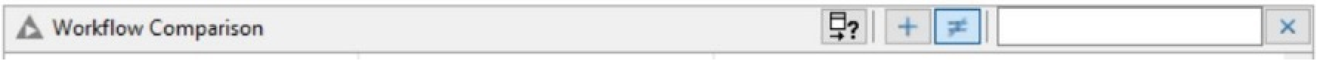


A Workflow Comparison can be triggered from every view that shows multiple workflows, e. g. KNIME Explorer and Server History.

In order to compare two workflows, select the two workflows and then click "**Compare**" in the context menu.

It is also possible to compare a workflow with itself. With this option, you are given a list of nodes from which you can choose the two nodes you want to compare (See Node Comparison).

To make it easier to see which changes have been identified there are three buttons in the upper right corner of the Workflow Comparison view.

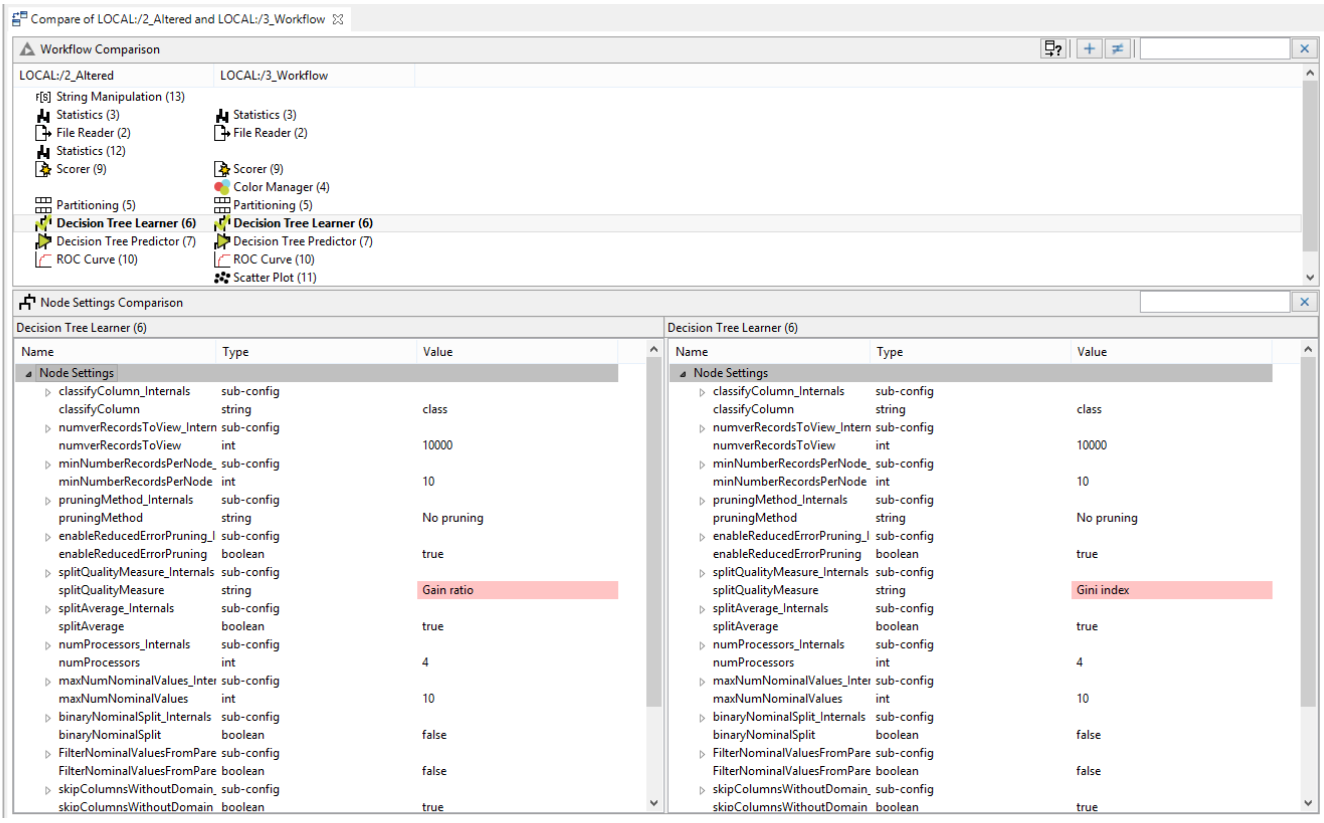


These buttons (from left to right) filter the list to:

- perform a Node Comparison of the selected nodes (see Node Comparison)
- show the added or removed nodes only
- hide the nodes with equal settings

Additionally you can see a search field, which makes it easy to check a special node or node type for changes. The ✕ clears the search field and displays all nodes.

Workflow Comparison is structure based. This means, that not only workflows, but also Metanode Templates, snapshots, Metanodes, WrappedMetanodes, and everything else that acts as a workflow can be compared with each other. This basically includes everything that can be seen in KNIME Explorer and Server History (except data).



Compare of LOCAL:/2_Altered and LOCAL:/3_Workflow

Workflow Comparison

LOCAL:/2_Altered	LOCAL:/3_Workflow
String Manipulation (13)	
Statistics (3)	Statistics (3)
File Reader (2)	File Reader (2)
Statistics (12)	
Scorer (9)	Scorer (9)
	Color Manager (4)
Partitioning (5)	Partitioning (5)
Decision Tree Learner (6)	**Decision Tree Learner (6)**
Decision Tree Predictor (7)	Decision Tree Predictor (7)
ROC Curve (10)	ROC Curve (10)
	Scatter Plot (11)

Node Settings Comparison

Decision Tree Learner (6)

Name	Type	Value
Node Settings		
classifyColumn_Internals	sub-config	
classifyColumn	string	class
numverRecordsToView_Intern	sub-config	
numverRecordsToView	int	10000
minNumberRecordsPerNode_	sub-config	
minNumberRecordsPerNode	int	10
pruningMethod_Internals	sub-config	
pruningMethod	string	No pruning
enableReducedErrorPruning_I	sub-config	
enableReducedErrorPruning	boolean	true
splitQualityMeasure_Internals	sub-config	
splitQualityMeasure	string	Gain ratio
splitAverage_Internals	sub-config	
splitAverage	boolean	true
numProcessors_Internals	sub-config	
numProcessors	int	4
maxNumNominalValues_Inter	sub-config	
maxNumNominalValues	int	10
binaryNominalSplit_Internals	sub-config	
binaryNominalSplit	boolean	false
FilterNominalValuesFromPare	sub-config	
FilterNominalValuesFromPare	boolean	false
skipColumnsWithoutDomain_	sub-config	
skipColumnsWithoutDomain	boolean	true

Decision Tree Learner (6)

Name	Type	Value
Node Settings		
classifyColumn_Internals	sub-config	
classifyColumn	string	class
numverRecordsToView_Intern	sub-config	
numverRecordsToView	int	10000
minNumberRecordsPerNode_	sub-config	
minNumberRecordsPerNode	int	10
pruningMethod_Internals	sub-config	
pruningMethod	string	No pruning
enableReducedErrorPruning_I	sub-config	
enableReducedErrorPruning	boolean	true
splitQualityMeasure_Internals	sub-config	
splitQualityMeasure	string	Gini index
splitAverage_Internals	sub-config	
splitAverage	boolean	true
numProcessors_Internals	sub-config	
numProcessors	int	4
maxNumNominalValues_Inter	sub-config	
maxNumNominalValues	int	10
binaryNominalSplit_Internals	sub-config	
binaryNominalSplit	boolean	false
FilterNominalValuesFromPare	sub-config	
FilterNominalValuesFromPare	boolean	false
skipColumnsWithoutDomain_	sub-config	
skipColumnsWithoutDomain	boolean	true

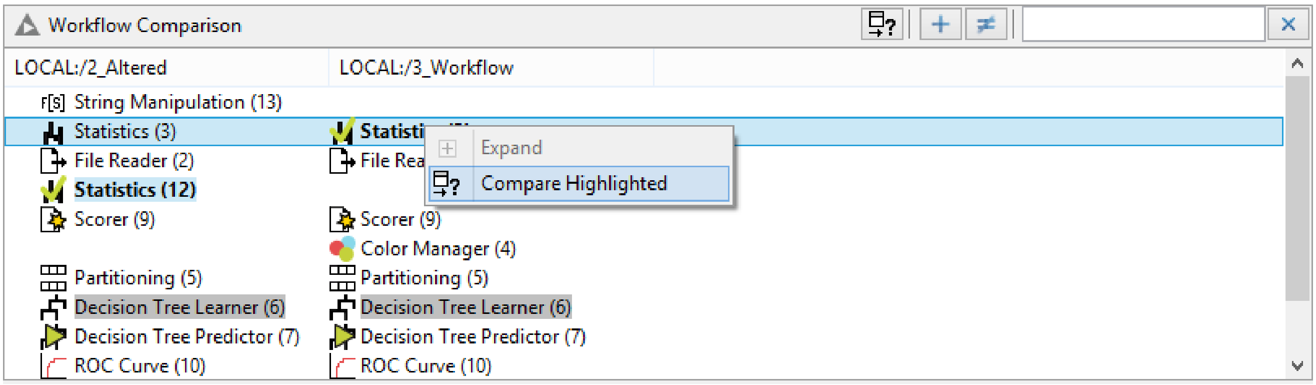
Workflow Comparison focuses on the functional structure of a workflow: Metanodes are expanded for the comparison; what you see in the Workflow Compare view is the content of the Metanodes. Wrapped Metanodes and Encrypted Metanodes, on the other hand, are treated as normal nodes (their content does not appear in the view).

If a Wrapped Metanode has been changed, it is highlighted in the Workflow Compare view, and the Node Comparison shows two additional entries: "Wrapped Metanode Content Hash" and "Wrapped Metanode Internal Settings Hash". These two numbers change whenever the content of the Wrapped Metanode (e.g. node insertion/substitution) or the settings of an internal node change, respectively.

The structural comparison is based on a sequential alignment with respect to the attributes (neighborhood, settings, etc.) of a node. It is designed to identify changes in a workflow. It might still be used to find similarities/common parts in any two workflows, however the usefulness of the results of these comparisons is often limited.

Node Comparison

Node Comparison is an additional view to Workflow Comparison. There are two ways to compare two nodes. You can either double-click both nodes in the Workflow Comparison view, or you can select them (single click) and then click either the "**Compare**" button in the view or right-click them to select "**Compare Highlighted**" in the context menu.



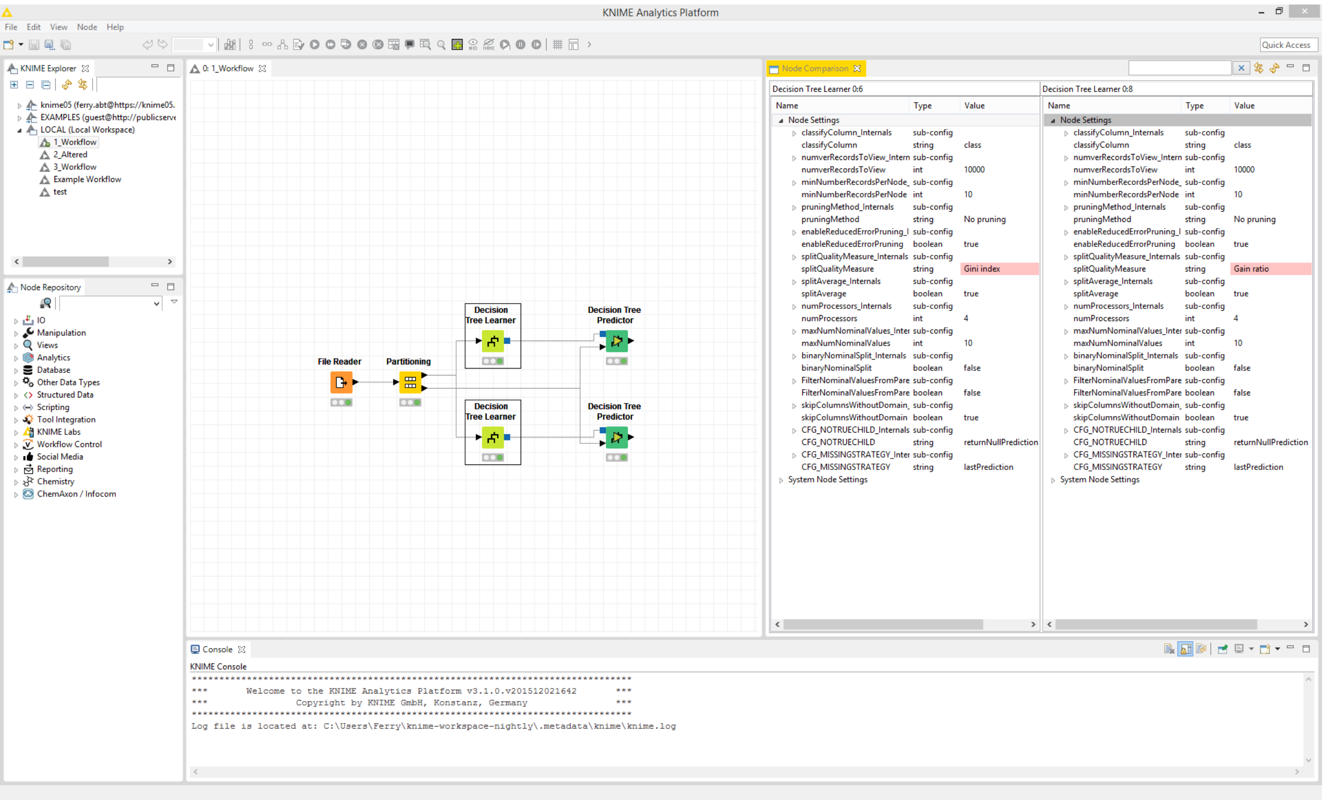
LOCAL:/2_Altered	LOCAL:/3_Workflow
String Manipulation (13)	
Statistics (3)	**Statisti**
File Reader (2)	File Rea
Statistics (12)	
Scorer (9)	Scorer (9)
	Color Manager (4)
Partitioning (5)	Partitioning (5)
Decision Tree Learner (6)	Decision Tree Learner (6)
Decision Tree Predictor (7)	Decision Tree Predictor (7)
ROC Curve (10)	ROC Curve (10)

The selected nodes are written in bold and include a green checkmark on the icon.

Node Comparison shows the settings of the nodes in two trees. Differing values and settings, which are not present in both nodes, are highlighted in red. If a changed setting is nested, the parent setting is highlighted in gray to indicate the hidden change. Click the arrow to expand the setting and show all nested settings.

To look for a specific setting the user can type the name into the search field in the upper right corner. This filters the list to show only those settings that contain the search query. As in Workflow Comparison, ✕ clears the search.

To compare the settings of two nodes in the same workflow (for example to compare two similar branches) the user can either compare the workflow with itself to retrieve the list of nodes, or open the workflow in the editor, select the two nodes of interest and select "**Compare**" in the context menu. This opens the Node Compare view. This view is identical to the one in Workflow Comparison, but is an independent view. This means the settings are displayed only as long as the view is open or until the user triggers a new comparison.



Decision Tree Learner 0:6

Name	Type	Value
Node Settings		
classifyColumn_Internals	sub-config	
classifyColumn	string	class
numverRecordsToView_Intern	sub-config	
numverRecordsToView	int	10000
minNumberRecordsPerNode_	sub-config	
minNumberRecordsPerNode	int	10
pruningMethod_Internals	sub-config	
pruningMethod	string	No pruning
enableReducedErrorPruning_I	sub-config	
enableReducedErrorPruning	boolean	true
splitQualityMeasure_Internals	sub-config	
splitQualityMeasure	string	Gini index
splitAverage_Internals	sub-config	
splitAverage	boolean	true
numProcessors_Internals	sub-config	
numProcessors	int	4
maxNumNominalValues_Inter	sub-config	
maxNumNominalValues	int	10
binaryNominalSplit_Internals	sub-config	
binaryNominalSplit	boolean	false
FilterNominalValuesFromPare	sub-config	
FilterNominalValuesFromPare	boolean	false
skipColumnsWithoutDomain_	sub-config	
skipColumnsWithoutDomain	boolean	true
CFG_NOTRUECHILD_Internals	sub-config	
CFG_NOTRUECHILD	string	returnNullPrediction
CFG_MISSINGSTRATEGY_Inter	sub-config	
CFG_MISSINGSTRATEGY	string	lastPrediction
System Node Settings		

Decision Tree Learner 0:8

Name	Type	Value
Node Settings		
classifyColumn_Internals	sub-config	
classifyColumn	string	class
numverRecordsToView_Intern	sub-config	
numverRecordsToView	int	10000
minNumberRecordsPerNode_	sub-config	
minNumberRecordsPerNode	int	10
pruningMethod_Internals	sub-config	
pruningMethod	string	No pruning
enableReducedErrorPruning_I	sub-config	
enableReducedErrorPruning	boolean	true
splitQualityMeasure_Internals	sub-config	
splitQualityMeasure	string	Gain ratio
splitAverage_Internals	sub-config	
splitAverage	boolean	true
numProcessors_Internals	sub-config	
numProcessors	int	4
maxNumNominalValues_Inter	sub-config	
maxNumNominalValues	int	10
binaryNominalSplit_Internals	sub-config	
binaryNominalSplit	boolean	false
FilterNominalValuesFromPare	sub-config	
FilterNominalValuesFromPare	boolean	false
skipColumnsWithoutDomain_	sub-config	
skipColumnsWithoutDomain	boolean	true
CFG_NOTRUECHILD_Internals	sub-config	
CFG_NOTRUECHILD	string	returnNullPrediction
CFG_MISSINGSTRATEGY_Inter	sub-config	
CFG_MISSINGSTRATEGY	string	lastPrediction
System Node Settings		

KNIME Console

```
*******************************************************************************
***       Welcome to the KNIME Analytics Platform v3.1.0.v201512021642      ***
***                 Copyright by KNIME GmbH, Konstanz, Germany              ***
*******************************************************************************
Log file is located at: C:\Users\Ferry\knime-workspace-nightly\.metadata\knime\knime.log
```

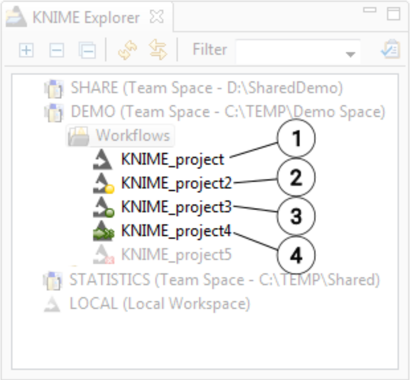
A subtle but powerful difference between Node Comparison in Workflow Comparison and the independent view is the toolbar, which has two additional buttons. The right button refreshes the view, retrieving the settings from the workflow again. This is useful for example if you find a difference between two nodes which should actually be identical. After identifying and changing the setting, click "**Refresh**" to show the new settings and confirm the new equality. The second button enables you to find the compared nodes in the workflow. If the workflow is still open in an editor, the nodes are selected and scrolled into the viewport.

Additional Server features

Teamspace vs Serverspace

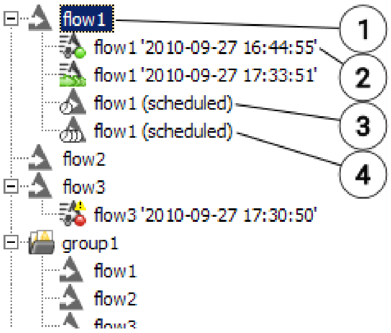
Here we outline the different types of remote repositories (TeamSpace and Server)

In a nutshell, TeamSpace provides a shared resource where workflows may be executed locally:



1. A workflow not opened by this KNIME Analytics Platform instance.
2. The workflow is not fully executed.
3. All nodes in the workflow are executed.
4. The workflow is currently executing.

KNIME Server enables user access permissions, server-side execution, and scheduling for more complex environments:

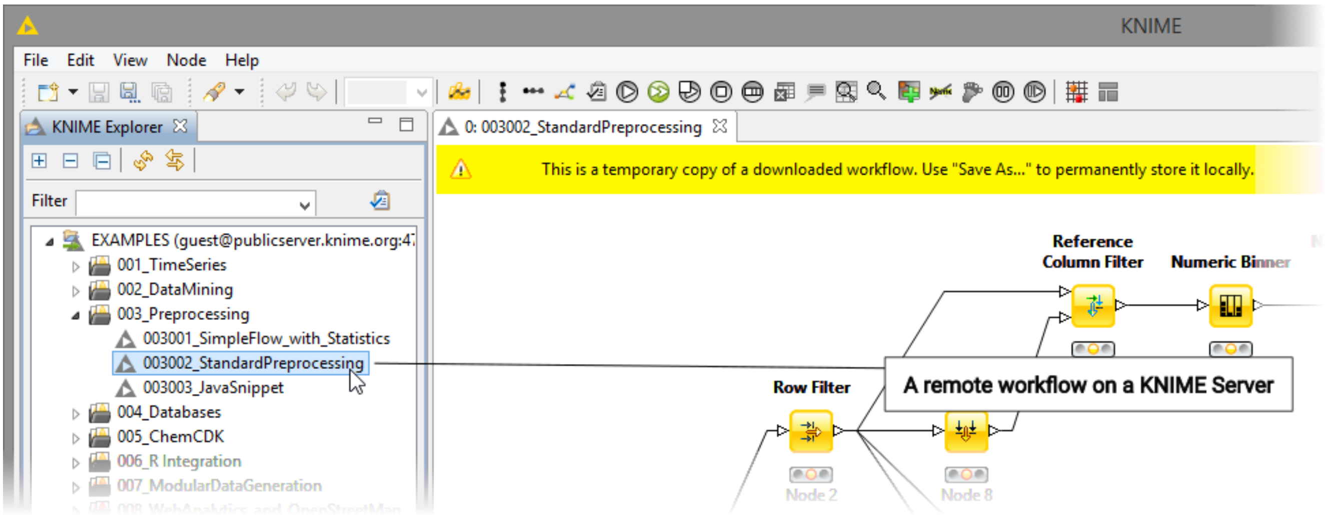

flow1
flow1 '2010-09-27 16:44:55'
flow1 '2010-09-27 17:33:51'
flow1 (scheduled)
flow1 (scheduled)
flow2
flow3
flow3 '2010-09-27 17:30:50'
group1
flow1
flow2
flow3
1
2
3
4

1. A remote workflow on KNIME Server
2. A successfully executed remote workflow
3. A workflow job scheduled for execution
4. A workflow scheduled for periodic execution

If a workflow in TeamSpace is opened by one user, it is locked for all other users; actually it is locked for all other KNIME Analytics Platform instances, i.e. the same user cannot open it with another KNIME Analytics Platform instance either. The workflow cannot be opened or modified by any other user while it is in use by one user. Also the workflow groups that contain it cannot be modified, i.e. not renamed or deleted nor moved or copied.

Inspecting a workflow from the Server

You cannot open a workflow on Server directly. You need to download it to the client before you can open it in an editor. By **double-clicking** a workflow on Server, the client downloads it (to a temporary location) and opens it automatically afterwards.

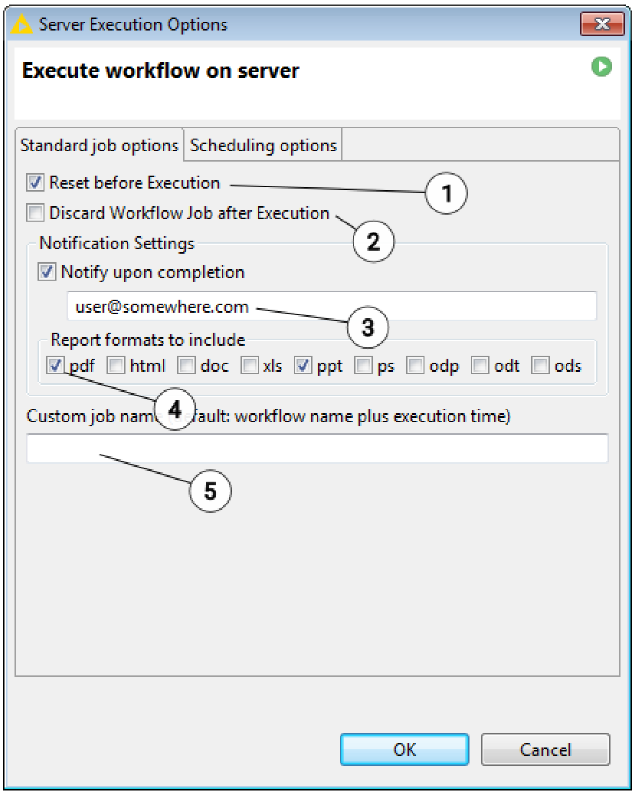


This is a temporary copy of a downloaded workflow. Use "Save As..." to permanently store it locally.

The yellow bar at the top of the editor indicates that this is a temporarily downloaded Server workflow. After changing the workflow it can be saved as any other workflow but you have to confirm that you want to overwrite the existing workflow on Server. Alternatively you may also save the workflow to a different location via the "**File**" → "**Save As…**" menu (or the corresponding toolbar button).

Executing a workflow on the Server

If "**Execute…**" is selected from the context menu of a workflow, the dialog shown below appears. In the first tab you set general options for execution of the job, such as if it should be reset before execution, if the job should be discarded immediately after execution, or you can configure notifications.



Server Execution Options

Execute workflow on server

Standard job options | Scheduling options

☑ Reset before Execution

□ Discard Workflow Job after Execution

Notification Settings

☑ Notify upon completion

user@somewhere.com

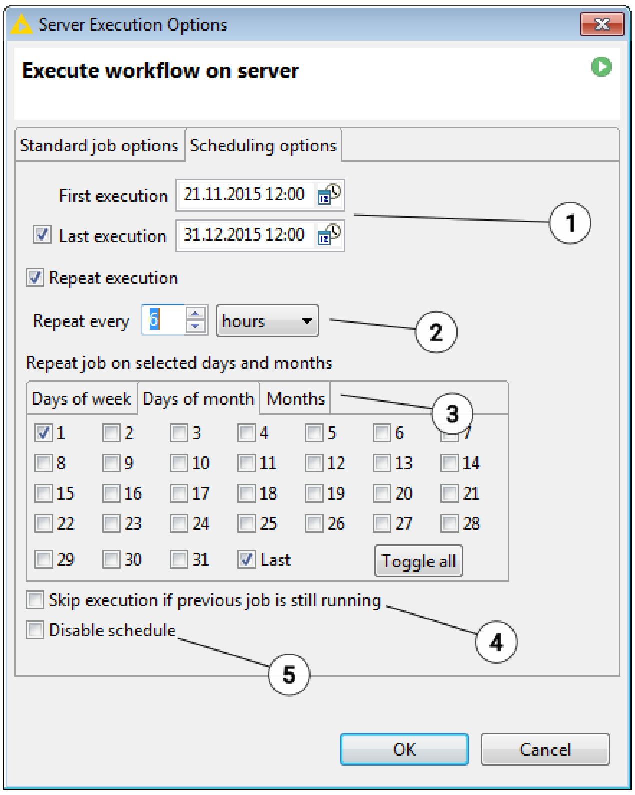
Report formats to include

☑ pdf □ html □ doc □ xls ☑ ppt □ ps □ odp □ odt □ ods

Custom job nam ault: workflow name plus execution time)

1. Check to reset workflow before execution. All nodes are reset (including File and Database Reader nodes, etc.).
2. If selected, the executed job is deleted immediately after successful execution, without saving it.
3. Enter one or multiple email addresses (separated by comma) to which a notification is sent after the workflow execution has been finished.
4. If the workflow contains a report, you can select in which formats the report should be attached to the notification emails.
5. The name of the workflow job as it is displayed in the server view. By default this is the name of the workflow. For repeating jobs, the execution date is always appended to the name.

By default, the job is executed immediately after you leave the dialog with OK. On the second tab you can specify options for a scheduled job. This includes start time, if the job should be executed repeatedly and when, or if execution should be skipped if a previous job is still running.



Server Execution Options

Execute workflow on server

Standard job options | Scheduling options

First execution 21.11.2015 12:00

☑ Last execution 31.12.2015 12:00

☑ Repeat execution

Repeat every 6 hours

Repeat job on selected days and months

Days of week | Days of month | Months

☑ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐ 6 ☐ 7
☐ 8 ☐ 9 ☐ 10 ☐ 11 ☐ 12 ☐ 13 ☐ 14
☐ 15 ☐ 16 ☐ 17 ☐ 18 ☐ 19 ☐ 20 ☐ 21
☐ 22 ☐ 23 ☐ 24 ☐ 25 ☐ 26 ☐ 27 ☐ 28
☐ 29 ☐ 30 ☐ 31 ☑ Last

☐ Skip execution if previous job is still running

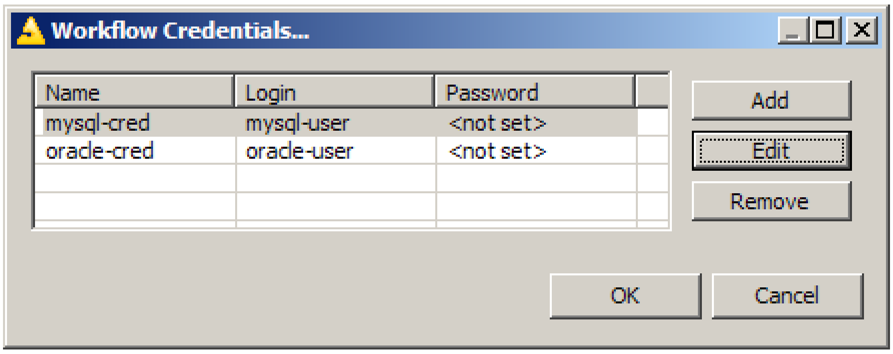
☐ Disable schedule

1. Here you can specify the first and optional the last execution date of the scheduled job (in case it is a repeating job).
2. 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).
3. By default repeating jobs are executed on every day. Here you can filter if 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 (which depends on the month).
4. Scheduled jobs can be disabled temporarily.
5. If the previous job is still running when the next execution should start, you can select to skip this execution.

If you click "**OK**", the workflow is loaded into the KNIME executor on Server.

Scheduled jobs are children of their corresponding workflow in the Server’s repository. While non-repeating scheduled jobs are automatically removed from the time table after their execution, repeating jobs exist until they are deleted by the user.

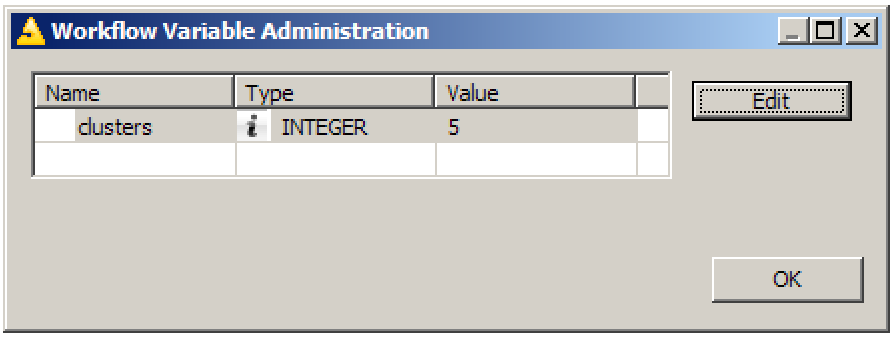
If the flow contains nodes that require credentials (user name and password) for them to log in to another system (for example Database Nodes), these credentials are usually stored as workflow credentials and the following dialog allows entering the credentials for this run:



Name	Login	Password
mysql-cred	mysql-user	<not set>
oracle-cred	oracle-user	<not set>

Select the credentials you want to change and click "**Edit**" to enter username and password.

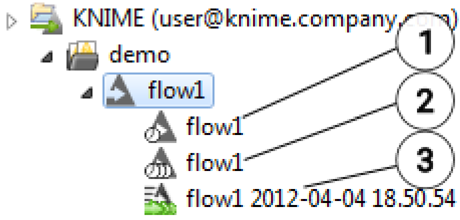
If the workflow contains flow variables, a dialog shows, allowing you to enter new values for these variables:



Name	Type	Value
clusters	INTEGER	5

Select the variable you want to change, click "**Edit**" and enter the new value. After you click "**OK**" in this dialog, the execution starts.

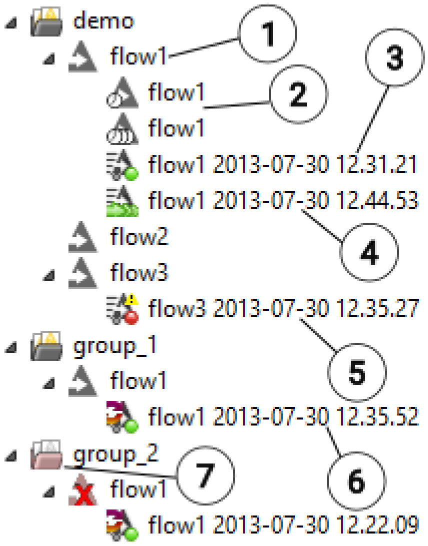
Executing workflows are displayed as "Workflow Jobs" in the server view. They show as children of their workflow with a decorator indicating their status. A flow can be executed simultaneously multiple times, creating multiple workflow jobs.



1. A scheduled workflow job (executing once at the specified time).
2. A flow scheduled for periodic execution.
3. A currently executing flow.

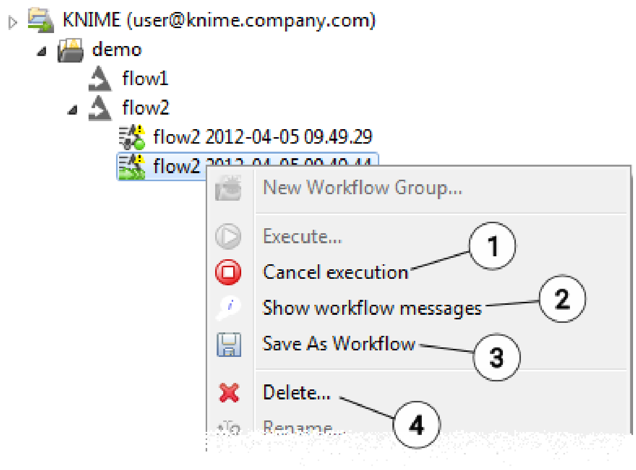
Please note, with multiple KNIME Analytics Platform instances for flow execution, one of the limited number of Analytics Platform instances is blocked as long as the job stays in memory. Consider checking "discard after execute" (especially with repeating scheduled jobs) in order to remove jobs that successfully executed and free resources.

Workflow job status


demo
flow1
flow1
flow1
flow1 2013-07-30 12.31.21
flow1 2013-07-30 12.44.53
flow2
flow3
flow3 2013-07-30 12.35.27
group_1
flow1
flow1 2013-07-30 12.35.52
group_2
flow1
flow1 2013-07-30 12.22.09
1
2
3
4
5
6
7

1. Workflow jobs are children of their corresponding workflow in the Server's repository.
2. Scheduled jobs (scheduled for single and periodic execution).
3. A successfully executed job: all nodes are executed. The flow stays in memory until it is removed manually through the context menu.
4. This job is currently executing in the KNIME executor on Server. The date and time it started is added to its (default) name.
5. This workflow job finished unsuccessfully. An error occurred. The yellow warning icon indicates node messages. They can be looked at through the context menu.
6. This icon indicates that the original workflow was been overwritten. The job executed from a previous version of the workflow.
7. If a workflow (and/or a workflow group) is deleted and jobs executing the flow still exist, a placeholder icon is created for the flow (or the group)

Workflow Jobs stay in the main memory of the server after execution (unless you checked "discard after execution") until you remove them manually. Right-click on the icon of the workflow job to open the context-menu:



1. Cancels workflow job execution (only available if the job is still executing).
2. Shows the messages that occurred in the workflow. Only available if a yellow warning icon shows at the workflow job.
3. Saves the workflow job as workflow in the Server's repository. (Only available if you have "read"/"download" access to the original workflow.)
4. Deletes the workflow job from the memory of the server (this does not affect the workflow in the Server's repository this job was generated from). All data is lost, if the flow was not saved before.

The result of the workflow job can currently only be inspected by saving it as workflow in the repository - this can also be achieved by dragging & dropping it into a different location inside the server repository - and then downloading that flow to your local workspace and opening it there. Currently it is not possible to download the workflow job directly.

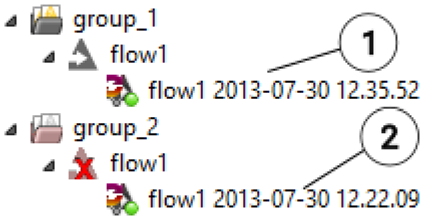
If Server is not configured to run multiple KNIME executor instances (see KNIME Server Installation Guide), a flow is executed on the server under the same user the server was started with. If it writes to a file, this user must have write permissions at the destination location. If the flow submits jobs into a cluster (separate KNIME Cluster Execution extension) they are submitted by this user.

Deleting workflows with jobs

If you overwrite a workflow with jobs, a dialog will open informing you about the update of a flow that is still in use on Server (possibly by other users whose jobs you do not see in your Explorer view). You get the details about the existing jobs and can then decide to overwrite the flow. The jobs that ran on the previous version of the workflow will not be affected by the update. That is, they will continue the execution on a temporary version of the original workflow. All workflow execution schedules will remain and be translated to use the newer version of the workflow automatically. The icons of jobs that execute from the original version of the workflow will be decorated to indicate their obsolescence.

If you move a workflow with existing jobs, you will be presented with a confirmation dialog which lists the details of these jobs as well as scheduled executions, even if they are owned by other users and not typically visible. If you decide to continue, the scheduled executions are also moved. Currently running executions will continue to execute in their original location using a temporary copy of the workflow. In the original location a placeholder icon will represent the deleted version of the workflow until it is no longer needed.

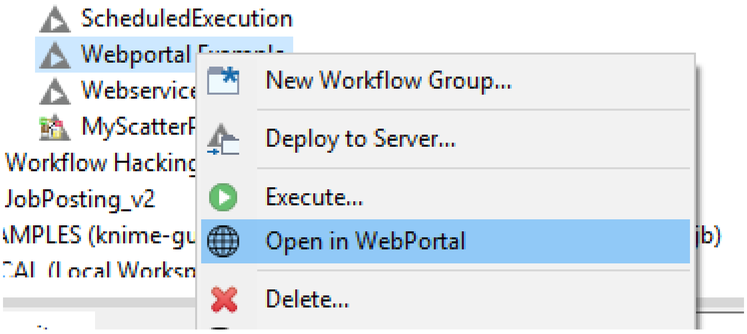
If you delete a workflow you will get a confirmation dialog with details about jobs in the executor and scheduled executions (possibly from other users) that still use this workflow. If you chose to delete the workflow the jobs that currently execute the workflow are not affected. They finish the execution on a temporary copy of the workflow. All scheduled executions of the workflow are deleted. If executing jobs exist, a new icon replaces the deleted flow, symbolizing the executed flow.



1. An "outdated" job executing a previous version of the flow1.
2. Placeholder icons are created for jobs executing deleted flows. "group_2" and "flow1" are deleted from the Server repository, but appear in the view to host the job.

Open in WebPortal

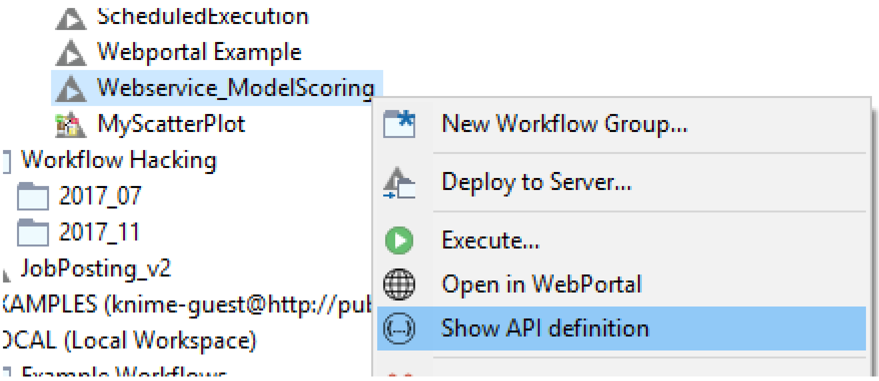
In case your server license covers KNIME WebPortal you can directly open a workflow in KNIME WebPortal from your Analytics Platform client. Simply click on the "**Open in WebPortal**" entry in the context menu of any workflow on Server.



This will open a new browser window with the selected workflow as starting point. You may need to authenticate first in case you do not have an active WebPortal session yet.

Show API definition

Workflows containing Quickform nodes such as *Integer Input*, *String, Input*, *JSON Input*, or *JSON Output* are exposed as RESTful webservices. Since KNIME Analytics Platform 3.5 and KNIME Server 4.6 an OpenAPI specification for such workflows is automatically generated (see the KNIME Server Administration Guide on how to create this information for existing workflows on your Server). You can view the specification by clicking on the "**Show API definition**" entry in the context menu of a workflow on Server.



This will open a new browser window with Swagger UI showing the specification of the selected workflow. You may need to authenticate first in case you do not have an active WebPortal session yet. Since OpenAPI 3.0 is a fairly new standard, Swagger UI does not support all features properly yet. For example, *File Upload* or *File Download* nodes are not shown yet. We will address these problems in the upcoming Server releases by updating the bundled Swagger UI.

User access permissions

You can assign access permissions to each Server item (workflows, workflow groups, Metanode Templates and data files) to control the access of other users to your workflows, groups and files.

The owner

Server stores the owner of each Server item, which is the user that created the item. When you upload a flow, file or Template, save a workflow job (an executed flow) or create a new workflow group you are assigned to the new item as owner. When a new server item is created, you can set the permissions how you want this item to be available to other users. Later on, only the owner can change permissions on an item.

User groups

When the KNIME Server administrator defines the users that have access to KNIME Server, the users are assigned to groups. Groups can be defined as needed - for example one group per department, or per research group, etc. Each user must be in at least one group, and could be in many groups.

There is a predefined special group called "admin". Users assigned to that group are considered Server Administrator.

Server administrator

A user that is assigned to the group "admin" is considered a Server Administrator. Administrators are not restricted by any access permissions. Administrators always have the right to perform any action usually controlled by user access rights. They can always change the owner of an item, change the permissions of an item, see all workflow jobs (while regular users only see their own jobs) and they can delete all jobs and items.

Workflow group permissions

Read	Allows the user to see the content of the workflow group. All workflows and subgroups are shown in the repository view.
Write	If granted, the user can create new items in this workflow group. One can create new sub-groups and can store new workflows in the group. Also deletion of the group is permitted.

Note: In order to access a workflow it is not necessary to have read-permissions in the workflow group the flow is contained in. Only the listing of contained flows is controlled by the read-right. Also, a flow can be deleted without the write-right in a group (if the corresponding right on the flow is granted).

Also, in order to add a flow to a certain group, you only need the permission to write to that particular group, not to any parent group.

Workflow user permissions

Execute	Allows the user to execute the flow, to create a workflow job from it. It does not include the right to download that job, or even store the job after it finishes (storing requires the right to download).

Write	If granted, the user can overwrite and delete the workflow.
Read	Allows the user to download the workflow (including all data stored in the flow) to its local desktop repository and inspect the flow freely.

Note: Executing or downloading a flow does not require the right to read in the group that contains the flow. In fact, there is currently no right controlling the visibility of a single flow (there is no hidden attribute).

Access to workflow jobs and scheduled jobs

There are no permissions to be set on a workflow job or a scheduled job. Only the owner - the user that created the job - can see the job in the repository view, and she is the only user that can delete it.

In order to store a workflow job as new workflow in the Server's repository, the user needs the right to download the original workflow (the flow, the job was created from). This behavior prevents an unauthorized user from downloading a workflow by executing it and downloading the resulting job.

"Owner", "Group", and "Other" rights

As the owner of a Server repository object (workflow, workflow group or file), you may grant workflow permissions to other users using the group level tools. Permissions for individual users other than the owner are not available.

Owner rights

The owner can assign permissions to herself to protect a flow from accidental deletion. She can change her own permissions at any time.

Group Rights

The owner of a server item can assign permissions to all users of a specific group. If an access right is granted to a group, all users that are in this group have this right.

"Other" rights

Permissions can be set to all users that are not the owner and that are not in one of the groups.

Note: Access rights are cumulative and cannot be withdrawn - for example, if you grant the right to execute a flow to "other" users and you define permissions for a certain group of users not including the execute right, these users of that group are still able to execute that flow, as they have obtained that right through the "other" permission settings.

Default permissions

When uploading or creating an item on Server its permissions are set to "inherit permissions from parent". This means that the item will automatically have the permissions of the closest parent item which does not have this default setting. If this workflow or file is moved to a new location, it will automatically inherit the permissions of the new parent. For example, if a workflow is uploaded to a group with the execution permission disabled, you will not be able to execute the workflow. If you then copy the workflow to a group that has the execute permission the workflow will become executable since it will inherit this attribute from its new parent.

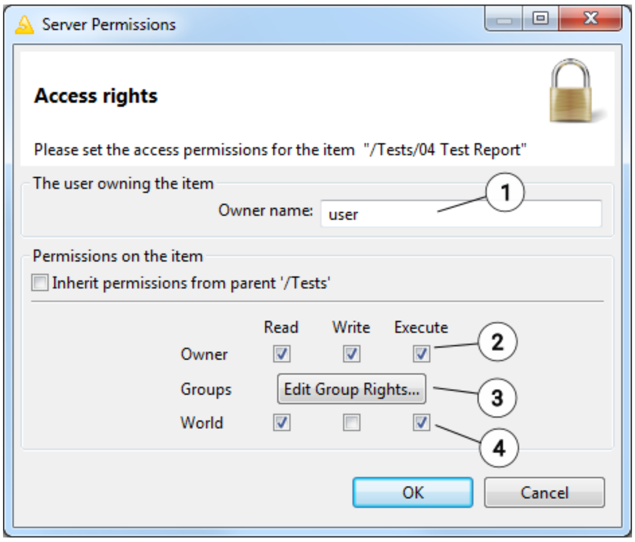
This feature is useful if you have "development" and "production" folders and you want items to become visible to end users when they are moved into the "production" folder, without the need to recursively change permissions on all items.

It is important to note that with these inherited permissions, everybody can access and modify your uploaded workflow if you are storing it in a "public" folder where everybody has write permissions. You can at any time change the permissions on an item through the "**Access Permissions**" dialog (available from the context menu of the item).

Setting/Inspecting access permissions

When a new Server item is created, you can assign it access rights through the "**Server Permissions**" dialog. If multiple new groups are created with one step, the specified permissions are applied to all groups created.

At any time, the owner can change the permissions from the context menu of the Server item, and all other users can inspect the permissions through this dialog:



Server Permissions

Access rights

Please set the access permissions for the item "/Tests/04 Test Report"

The user owning the item

Owner name: user

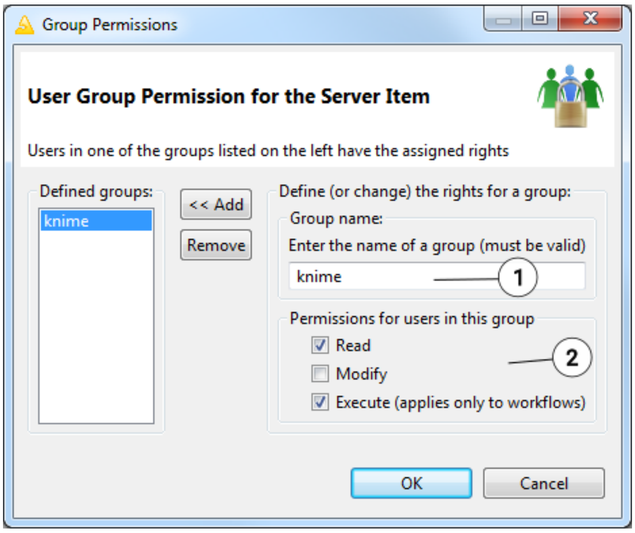
Permissions on the item

☐ Inherit permissions from parent '/Tests'

	Read	Write	Execute
Owner	☑	☑	☑
Groups	Edit Group Rights...		
World	☑	☐	☑

1. Only administrators can change the owner of the Server item. Note: Enter the exact name. No spell checking is done. Note: Only the new owner (and administrators) can change permissions from then on.
2. Grant access rights to you here.
3. If you want to set group permissions, click here to open the "**Group Permissions**" dialog.
4. Each checkmark grants a right to "other" users that are not in one of the groups.

Add group permissions in this dialog:



Group Permissions

User Group Permission for the Server Item

Users in one of the groups listed on the left have the assigned rights

Defined groups: knime

Define (or change) the rights for a group:

Group name:

Enter the name of a group (must be valid)

knime

Permissions for users in this group

☑ Read

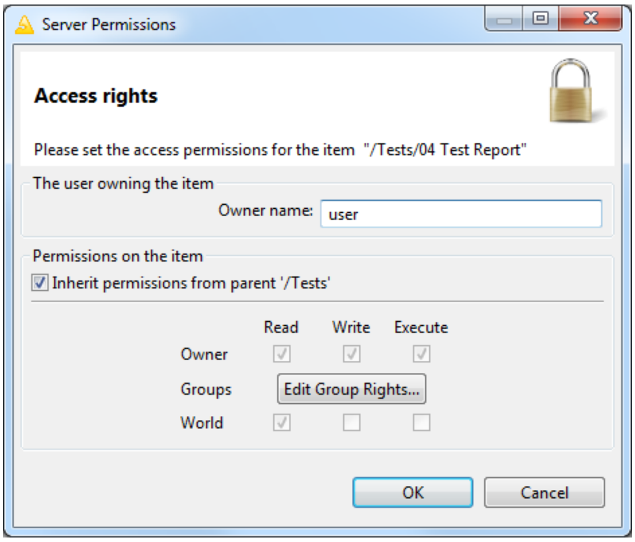
☐ Modify

☑ Execute (applies only to workflows)

1. Enter the name of the group you want to grant access rights. Note: Enter the exact name, no spell checking is done.
2. Each checkmark grants the corresponding right to all users of the group.

Only groups in the list on the left hand side have permissions granted. Select any group to change the permissions on. Add this group to apply those changes. Note: It is important to click the "**Add**" button after making changes to the permissions (in the right panel). If you click "**OK**" without prior "**Add**", the changes are not taken over but are discarded and lost.

Inheriting permissions from the parent



Server Permissions

Access rights

Please set the access permissions for the item "/Tests/04 Test Report"

The user owning the item

Owner name: user

Permissions on the item

☑ Inherit permissions from parent '/Tests'

	Read	Write	Execute
Owner	☑	☑	☑
Groups	Edit Group Rights...		
World	☑	□	□

If the option "Inherit permissions from parent" is set, the checkboxes for the individual rights are disabled and the access rights for the current item are taken over from the parent workflow group.

You can observe however, the effective permissions on the item, which are inherited. If you move the item, the effective permissions might change (as its parent may change). The same applies to group permissions.

Setting permissions recursively

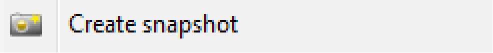
In the permissions dialog of workflow groups, there is the option (at the bottom of the dialog) to apply the permissions to the selected group and all elements contained in that group. If you select this option and click "**OK**", the permissions that apply are set on all groups and flows on which you have the right to change the permissions (i.e. the elements you are the owner of).

As an administrator you can also change the owner of an item. By default, changing the permissions recursively will keep the original owner - unless you select the corresponding option ("Also change the owner recursively").

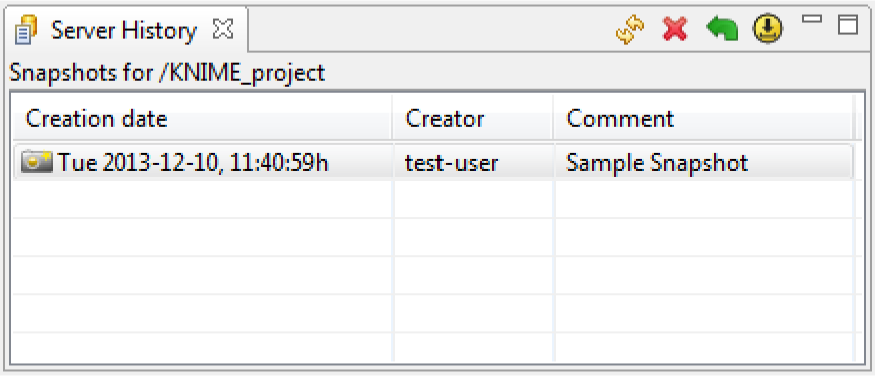
Versioning

It is possible to create a history of items on Server. To do that, you can create snapshots of workflows, data files and Metanode Templates. These are stored with a timestamp and a comment.

You can manually create a snapshot by right clicking an item and selecting "**Create snapshot**" from the context menu.



After entering an optional comment, the snapshot is created and will be visible in the "Server History" view. You can enable the view by selecting it from the dialog after clicking on "**View**" → "**Other…**" and choosing the "**KNIME Views**" category.



Snapshots for /KNIME_project

Creation date	Creator	Comment
Tue 2013-12-10, 11:40:59h	test-user	Sample Snapshot

Additionally you can create a snapshot every time you overwrite an item on Server, by selecting the appropriate checkbox in the confirmation dialog.

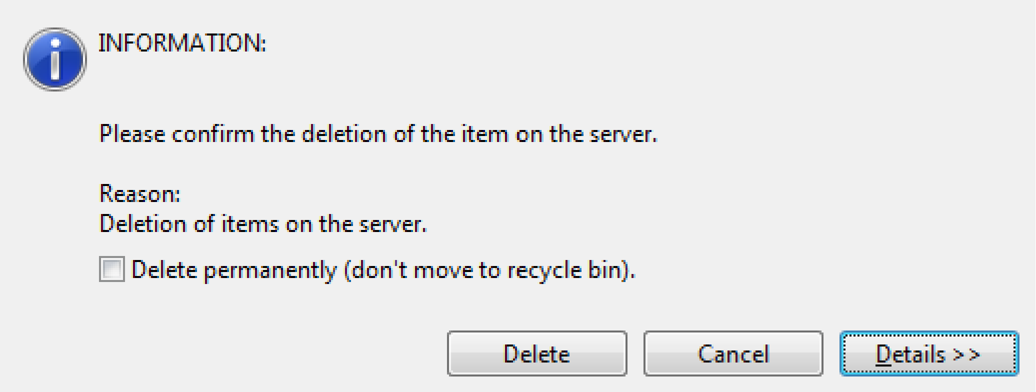
The "Server History" view allows you to view and interact with the snapshots you have created. You can restore a certain item to a previous state or download a snapshot into your local repository.

Important: You need write permissions on any item you want to create a snapshot of. It is also not possible to create a snapshot of a workflow group.

When overwriting any Server item with another item from Server the history of the overwritten file will also be substituted by the history of the overwriting item. Also note that the history is not available in the local workspace and thus will be lost on downloaded items.

Recycle bin

KNIME Server also offers a recycle bin feature.

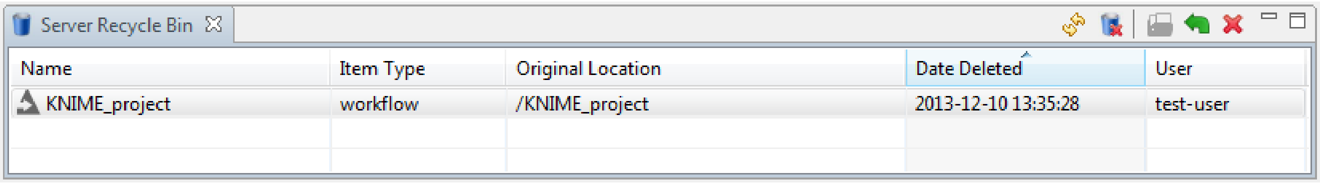

INFORMATION:

Please confirm the deletion of the item on the server.

Reason:
Deletion of items on the server.

□ Delete permanently (don't move to recycle bin).

Every time you delete an item on KNIME Server, the item is actually moved to the Recycle Bin, unless otherwise selected in the confirmation dialog. You can see the contents of the Recycle Bin in the "Server Recycle Bin" view. This view is available after selecting it from the dialog after clicking on "**View**" → "**Other…**" and choosing the "**KNIME Views**" category.



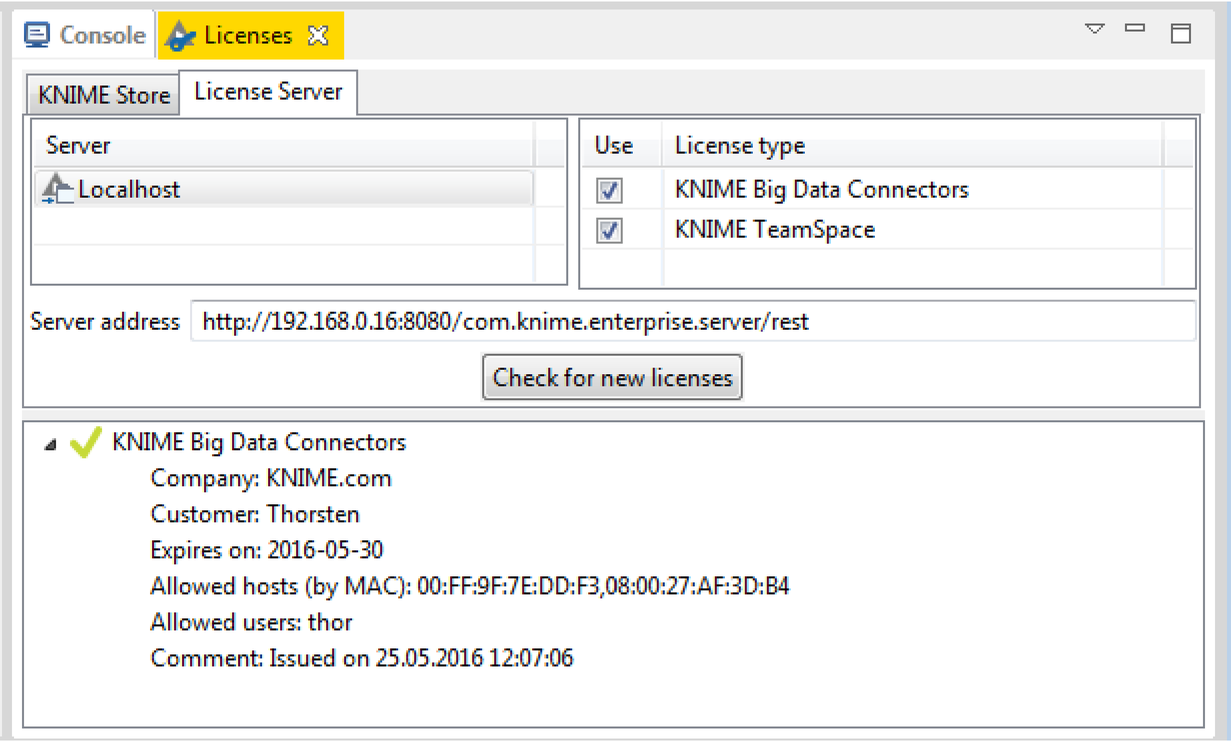
Server Recycle Bin

Name	Item Type	Original Location	Date Deleted	User
KNIME_project	workflow	/KNIME_project	2013-12-10 13:35:28	test-user

In this view you will always see all deleted items of the Server instance currently logged into (permissions granted). You can restore or permanently delete items from the Recycle Bin view and additionally show the original contents of deleted workflow groups.

Retrieving client licenses

If you are running KNIME Server 4.3 or later, Server is capable of distributing licenses for additional extensions, such as TeamSpace or BigData Connectors, to clients. Such licenses can be requested via the Licenses view. To open this view go to "**View**" → "**Other…**" and select the "**KNIME Views/Licenses**" view.



Server
Localhost

Use	License type
☑	KNIME Big Data Connectors
☑	KNIME TeamSpace

Server address: http://192.168.0.16:8080/com.knime.enterprise.server/rest

✓ KNIME Big Data Connectors
Company: KNIME.com
Customer: Thorsten
Expires on: 2016-05-30
Allowed hosts (by MAC): 00:FF:9F:7E:DD:F3,08:00:27:AF:3D:B4
Allowed users: thor
Comment: Issued on 25.05.2016 12:07:06

In the second tab of the view you should see a list of available license Servers, which is a subset of all configured Servers in KNIME Explorer. Depending on the number of Servers and their reachability it may take a few seconds before the list is populated. Only Servers that distribute licenses are shown. In case you do not have a mount point to the license Server, you can also enter its address in the text field and press Enter. Once a Server is selected (or an address is entered manually), the table on the right shows the available licenses. Select the licenses that you want to use and click on the "**Check for new licenses**" button. The retrieved licenses will then be shown in the tree view below.

Once a license Server is configured, a background job will check for new licenses every day. By default licenses retrieved from a license Server are valid for five days and stored locally in the workspace. This means you can use such licenses also when you do not have access to Server (until they expire).

The settings in this view are stored as part of the workspace preferences which means they can be transferred into other workspaces by ex- and importing the preferences.

Open for Innovation
KNIME

KNIME AG
Technoparkstrasse 1
8005 Zurich, Switzerland
www.knime.com
info@knime.com