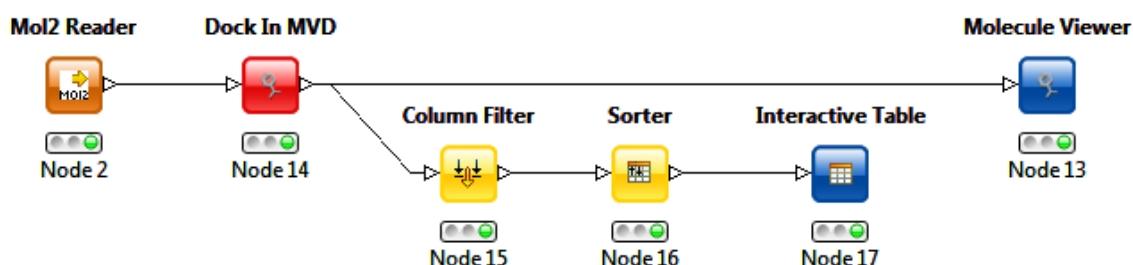


Molegro KNIME Extensions

Installation and usage guide

(for version 2.0.0)



Molexus

Information in this document is subject to change without notice and is provided "as is" with no warranty. Molexus makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Molexus shall not be liable for errors contained herein or for any direct, indirect, special, incidental, or consequential damages in connection with the use of this material.

Table of Contents

1	Installing Molegro Knime Extensions.....	3
1.1	Requirements.....	3
1.2	Installation.....	3
	Unable to access the Molexus website through KNIME?.....	4
2	Molegro KNIME Nodes.....	5
2.1	Global Preferences.....	5
2.2	Open In MDM.....	6
	Configuration.....	6
2.3	Apply MDM Model.....	7
	Configuration.....	7
	Example workflow.....	7
2.4	Molecule Viewer.....	9
	Configuration.....	9
	Example workflow.....	10
2.5	Dock in MVD.....	11
	Configuration.....	12
	Example workflow.....	13

1 Installing Molegro Knime Extensions

1.1 Requirements

KNIME version 3.6.2 or later (can be obtained at <http://www.knime.org/>).
Licensed and installed versions of relevant Molegro software:

- **Open In MDM** and **Apply MDM Model** requires MDM 6.5.0 or later
- **Molecule Viewer** requires at least MVD 6.5.0 or MMV 6.5.0 or later)
- **Dock In MVD** requires at least MVD 6.5.0

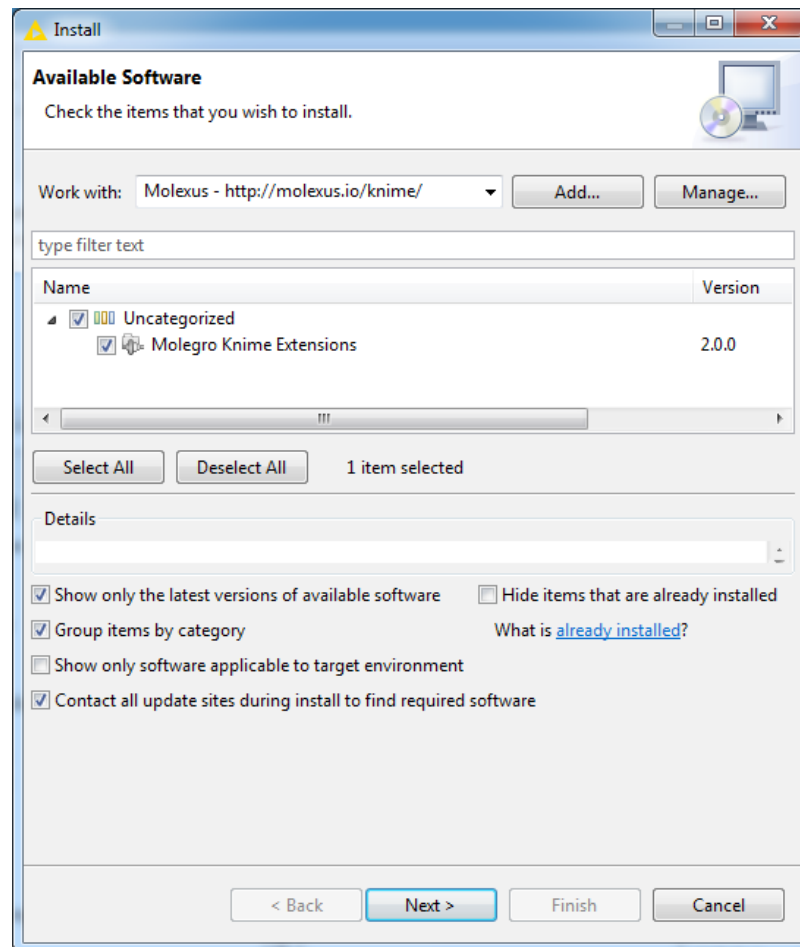
It is recommended to install the following packages for reading and writing MOL2/SDF/MOL files:

- KNIME Chemistry Add-Ons
- KNIME Base Chemistry Types & Nodes

1.2 Installation

Molegro KNIME Extensions use the KNIME Feature Installation System. This makes it easier to install and update the plugins.

To install the KNIME Extensions, open KNIME, choose **Help | Install New Software...**, click the **Add...** button, and add a new site with the name Molexus and the following location: <http://molexus.io/knime/>



Now, set the **Work with:** combobox to **Molexus**, check the **Molegro KNIME Extensions** branch, and click **Next>** to install.

Unable to access the Molexus website through KNIME?

Due to firewall restrictions it may not be possible to access the Molexus website through KNIME.

In this case, the plugin must be installed manually, by downloading the plugin and copying it to the 'dropins' directory in the KNIME installation folder.

The latest version of the Molegro KNIME Extension plugin can be downloaded here: <http://molexus.io/knime-extensions/>

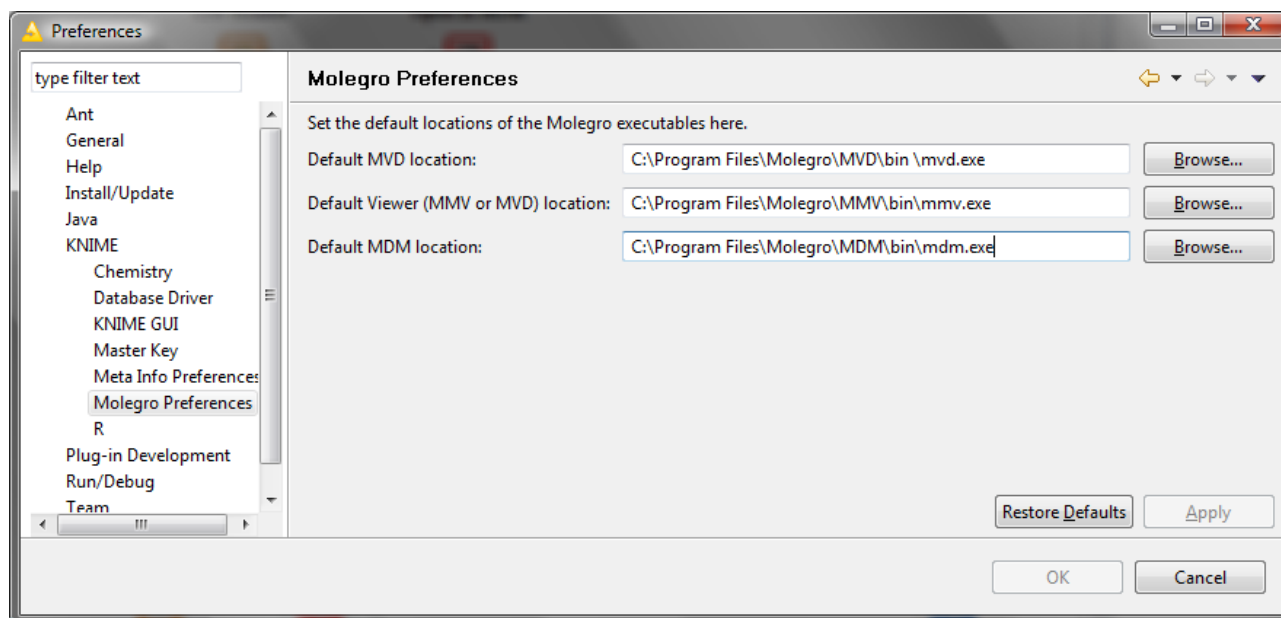
(Download the file named: com.molegro_2.x.x.jar, where '2.x.x' is the version number)

Notice that by manually installing the plugin, it is not possible to automatically check for new updates.

2 Molegro KNIME Nodes

Molegro KNIME extensions exist for Molegro Data Modeller, Molegro Virtual Docker, and Molegro Molecular Viewer. The available nodes are described below.

2.1 Global Preferences



Every Molegro node for KNIME needs to know the location of the Molegro applications it uses. These locations can be specified for each node in its configuration dialog, but it is easier to specify the settings in the global preferences. These can be accessed by opening **File | Preferences**, and selecting the **KNIME | Molegro Preferences** page.

When the location of a binary executable is specified here, it will serve as the default location each time a new Molegro node is inserted into a workflow.

Notice, that it is still possible to override the locations in the node configuration dialog, and that changing the location preferences will not change the location for nodes that are already inserted into an workflow – only new nodes will be affected.

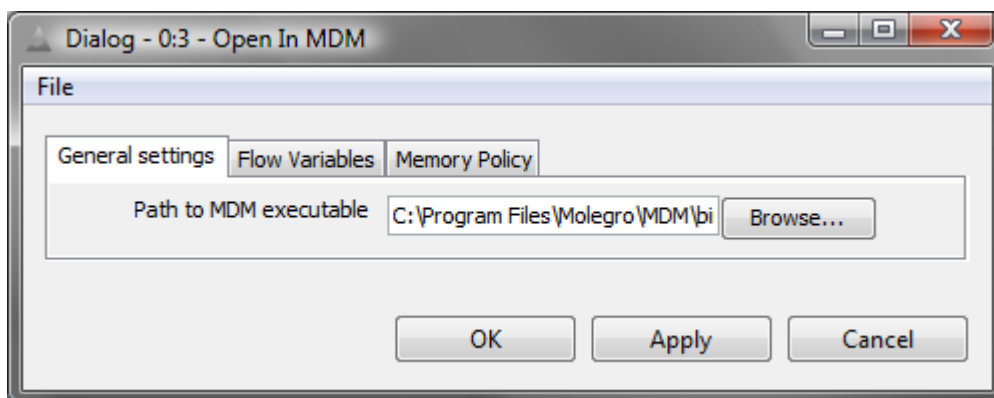
2.2 Open In MDM

Opens a KNIME input table in Molegro Data Modeller for visualization and model building. While KNIME has some support for browsing and visualizing data, MDM provides a much more interactive environment for exploring data. Molegro Data Modeller also makes it much easier to parameter tune regression and classification models and perform feature selection. Models created in MDM can be used together with the 'Apply MDM Model' node.

Notice: Data is passed to MDM through a temporary file. This file is automatically deleted, when MDM terminates. If you make changes to the data in MDM, you must save them to another location. The changes will not be visible to the KNIME environment. HiLiting is not supported.

After the node has been executed, it will change to the 'executed' state (indicated by green status light). If you want to execute the node again, it is necessary to reset the node to 'configured' state (yellow status light). This can be done using the context menu for the node.

Configuration



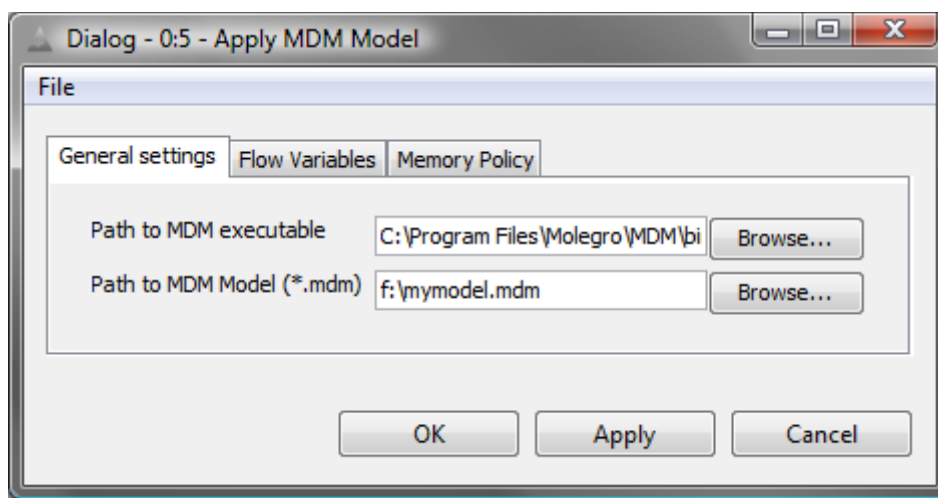
This node only requires that the path to the MDM executable is specified in the 'Configure' dialog.

2.3 Apply MDM Model

Applies a MDM model to a KNIME input table.

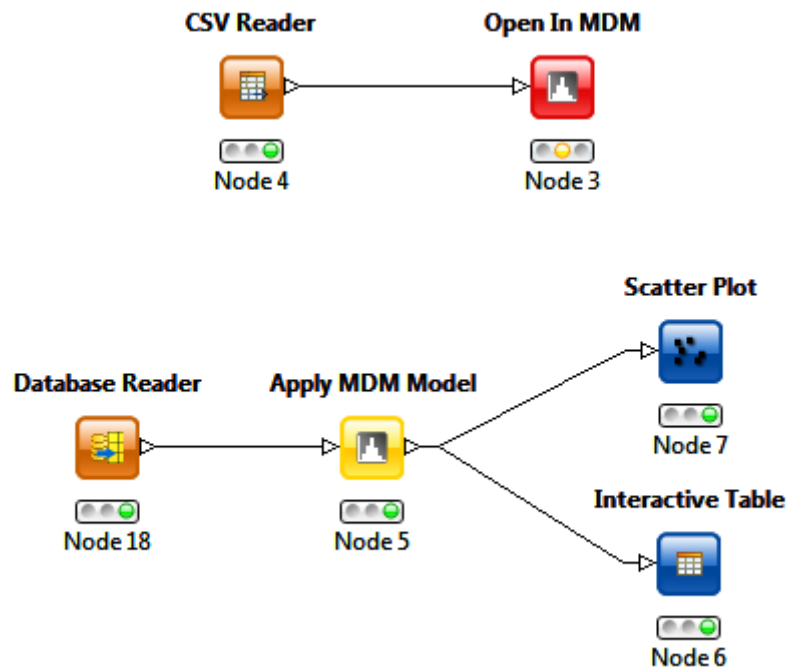
The model must be created in MDM beforehand and saved as a *.mdm file. A temporary copy of the table is saved to disc, before invoking the MDM executable.

Configuration



In the configuration dialog the path to the MDM executable and the path to the MDM model must be specified. The MDM model file must contain exactly one MDM model and no datasets.

Example workflow



In the illustration above the upper workflow has been created to read data from a CSV file and open it in MDM. In MDM a regression model has been trained on the data from the CSV file, and saved as a model in *.MDM format. The lower workflow shows how to apply the model to data from a database. Notice how the data is split into two connections: one to an **Interactive Table**, which allows for selections, and one to a **Scatter Plot** node.

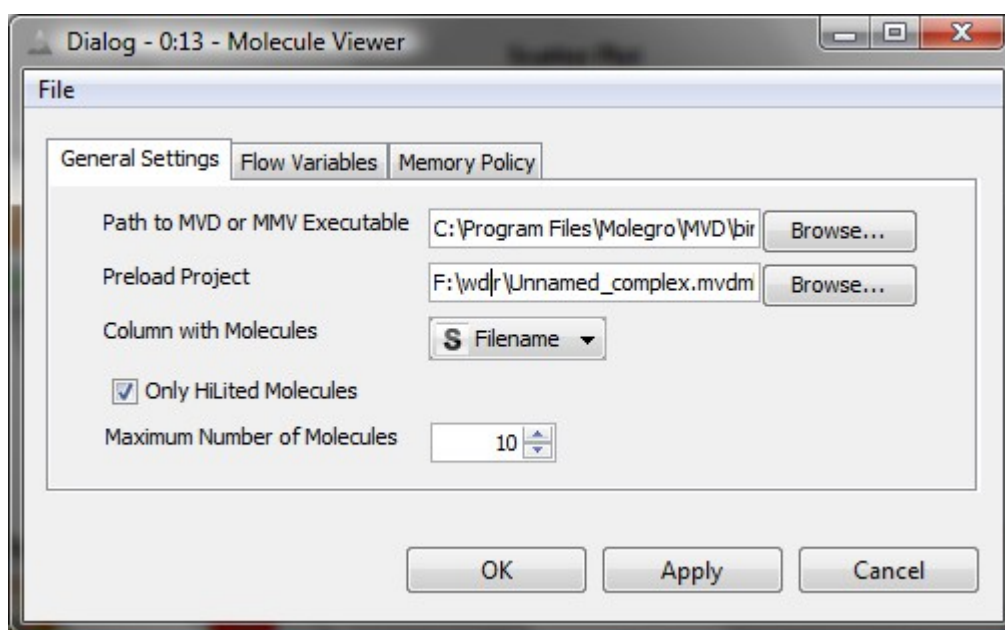
2.4 Molecule Viewer

Opens the molecules from a selected column in a KNIME input table in either Molegro Virtual Docker or Molegro Molecular Viewer.

The molecules must be present in the selected table column either as filenames with absolute paths (e.g. "C:\molecules\ligand.sdf") or in one of the chemical formats supported by KNIME: Mol2Cell, SdfCell, MolCell (or one of their 'Blob'-cell variants).

After the node has been executed, it will change to the 'executed' state (indicated by green status light). If you want to execute the node again (for instance if the viewer has been closed), it is necessary to reset the node to 'configured' state (yellow status light). This can be done using the context menu for the node.

Configuration



Path to MVD or MMV executable must point to relevant file on disk.

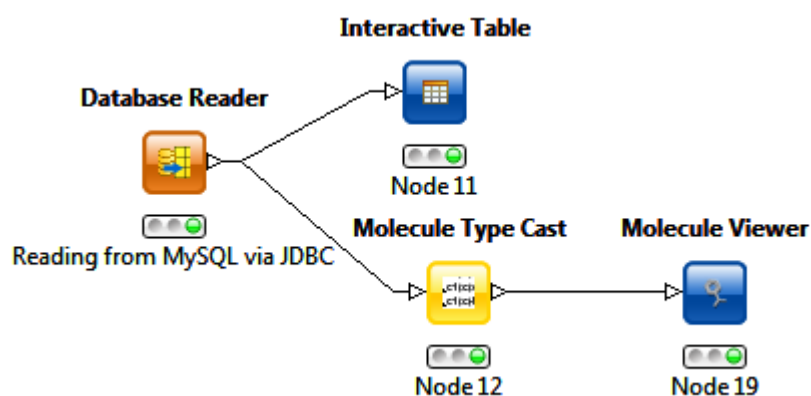
Preload Project makes it possible to specify a workspace project (a MVDML file) that is loaded before the molecules in the table. This can be used to e.g. preload a protein when inspecting poses from a docking run.

Column with Molecules determines which column in the input table, that contains molecules to use as docking input. The molecules must be present in the selected table column as either filenames with absolute paths (e.g. "C:\molecules\ligand.sdf") or in one of the chemical formats supported by KNIME: Mol2Cell, SdfCell, MolCell (or one of their 'Blob'-cell variants).

Only HiLited Molecules. If this is checked, only HiLited molecules will be sent to the viewer. This makes it easy to only view a limited subset of the molecules in a table. Notice, that if no molecules are HiLited, the option will be ignored (the maximum number of molecules will be sent to the viewer).

Maximum Number of Molecules. This determines the maximum amount of molecules passed to the viewer. The proper way to use the Molecule Viewer node is to view only a small amount of molecules: opening thousands of molecules will be slow and difficult to manage from the viewer. Instead limit the number of molecules by HiLiting the relevant ones in the table (see the option above).

Example workflow



The illustration above shows a typical workflow. Here molecules are read from a database, where they are stored as text strings in Mol2 format. Since the **Molecule Viewer** cannot determine the molecular format without knowing the filename extension, it is necessary to *cast* the strings into a type known by KNIME. Here, the **Molecule Type Cast** node (which can be found in the **Chemistry | Translators** category), has been configured to cast the strings from the database into Mol2 format. Notice, that this does not convert the molecule format or change the text content of the cells in the table – it is only a way to tell KNIME what type of data we are reading from the database.

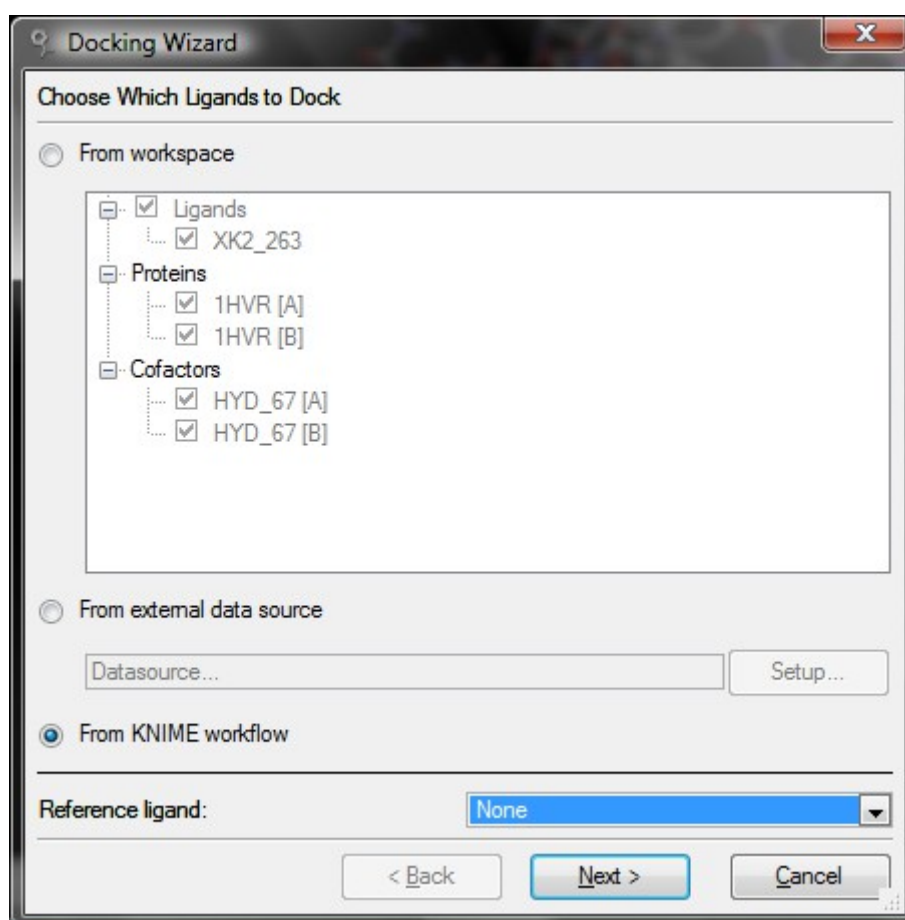
Notice how the **Interactive Table** is connected: this way we can choose to only view a selection of molecules from the database by HiLiting them in the table.

2.5 Dock in MVD

This node makes it possible to dock a number of ligands from an input table against a protein target.

In order to use this node, a protein must have been imported and prepared in Molegro Virtual Docker, and a script containing all settings must have been created in the Docking Wizard. Any script created by the Docking Wizard can be used – the KNIME node will scan the script for ligands being docked (the 'DOCK' line), and replace those with input ligands from the KNIME workflow.

It is also possible to create a script specifically for a KNIME workflow directly from the Docking Wizard: On the first page in the Docking Wizard, choose the **From KNIME workflow** option.

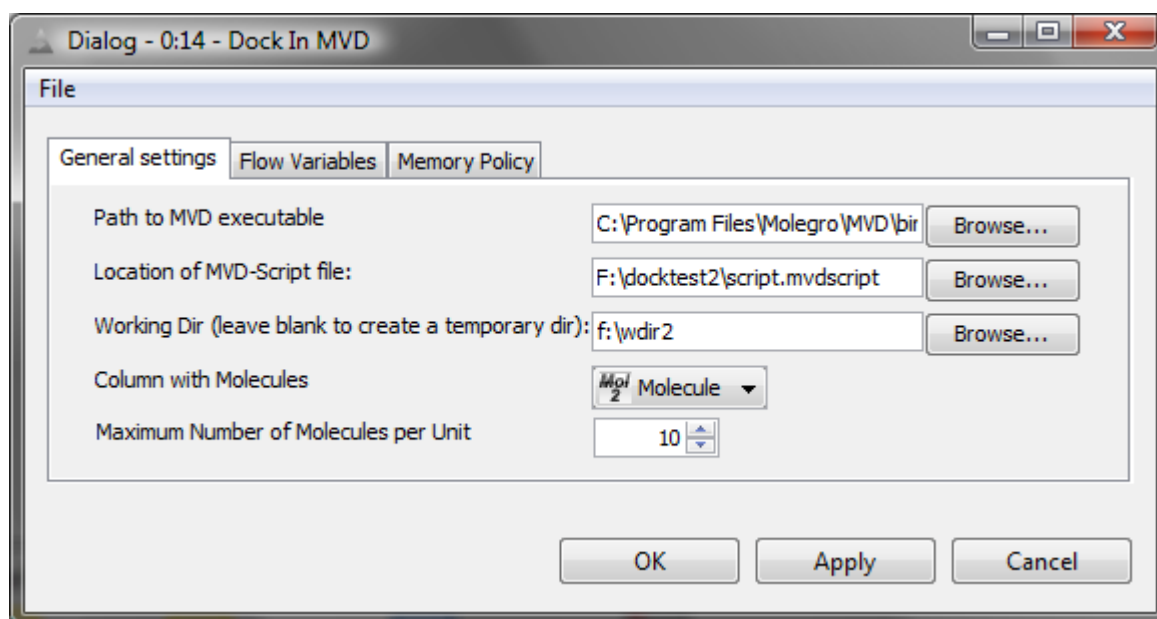


After the node has been executed, it will change to the 'executed' state (indicated by green status light). If you want to execute the node again (for instance if the viewer has been closed), it is necessary to reset the node to

'configured' state (yellow status light). This can be done using the context menu for the node.

If the job is cancelled by the user (this can be done using the context menu for the node), the work is not lost: it is possible to execute the node again, which will display a dialog with options for either resuming the job, or sending the results previously obtained to the output port for further processing.

Configuration



In the configuration dialog the path to the MVD executable and the MVD-Script must be specified.

All files in the same folder as the MVD-Script are copied into the working directory when the node is executed – this is done to ensure that the required dependencies (e.g. the protein target) for the node are available when MVD is executed.

The **Working Dir** is used to store all input files to MVD (a script for each unit, the molecules from the input table, and the protein target) and all output files (docking results files with energies and poses).

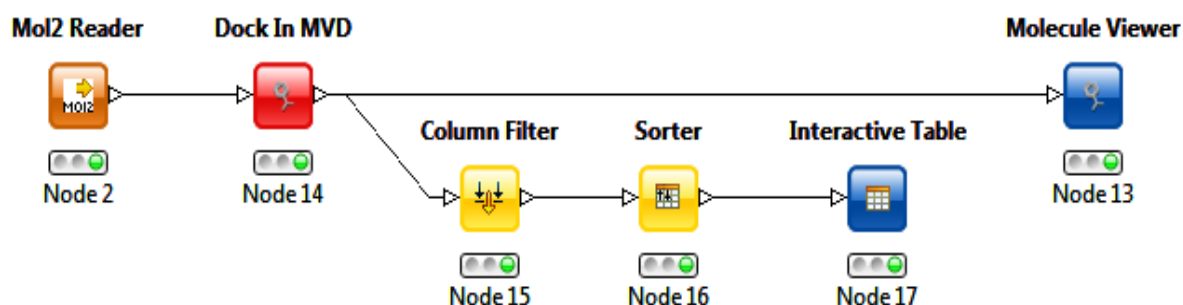
Column with Molecules determines which column in the input table that contains the molecules to use as docking input. The molecules must be present in the selected table column as either filenames with absolute paths (e.g. "C:\molecules\ligand.sdf") or in one of the chemical formats supported by KNIME: Mol2Cell, SdfCell, MolCell (or one of their 'Blob'-cell variants).

Maximum Number of Molecules per Unit. When MVD docks a number of molecules it splits the job into *units*. Each unit is processed by a new instance of MVD. The molecules in a given unit will reuse the same energy grid, making

it faster to process multiple molecules – on the other hand, if you want to pause and resume the job later on, you can only resume from completed units.

Example workflow

A typical KNIME workflow starts with a KNIME source node, which reads molecules from a file or a database. These input molecules are then passed into the **Dock In MVD** node. This usually results in a large number of poses in the output table. To be able to navigate this data, it is a good idea to apply a **Column Filter** to filter out only the relevant columns, such as the docking energy. Finally, to access e.g. the highest ranked poses, a **Sorter** node can be inserted before passing the filtered and sorted data into an interactive table.



When browsing the interactive table, the most interesting molecules can be investigated further in the **Molecule Viewer** by HiLiting them before running the viewer. Notice that it is a good idea to preload the protein in the **Molecule Viewer** (see the configuration description for the Molegro Molecular Viewer for more information about this).