



# Introduction to KNIME

The Konstanz Information Miner

Tobias Kötter

University of Konstanz, Germany

*Tobias.Koetter@uni-konstanz.de*



# Agenda

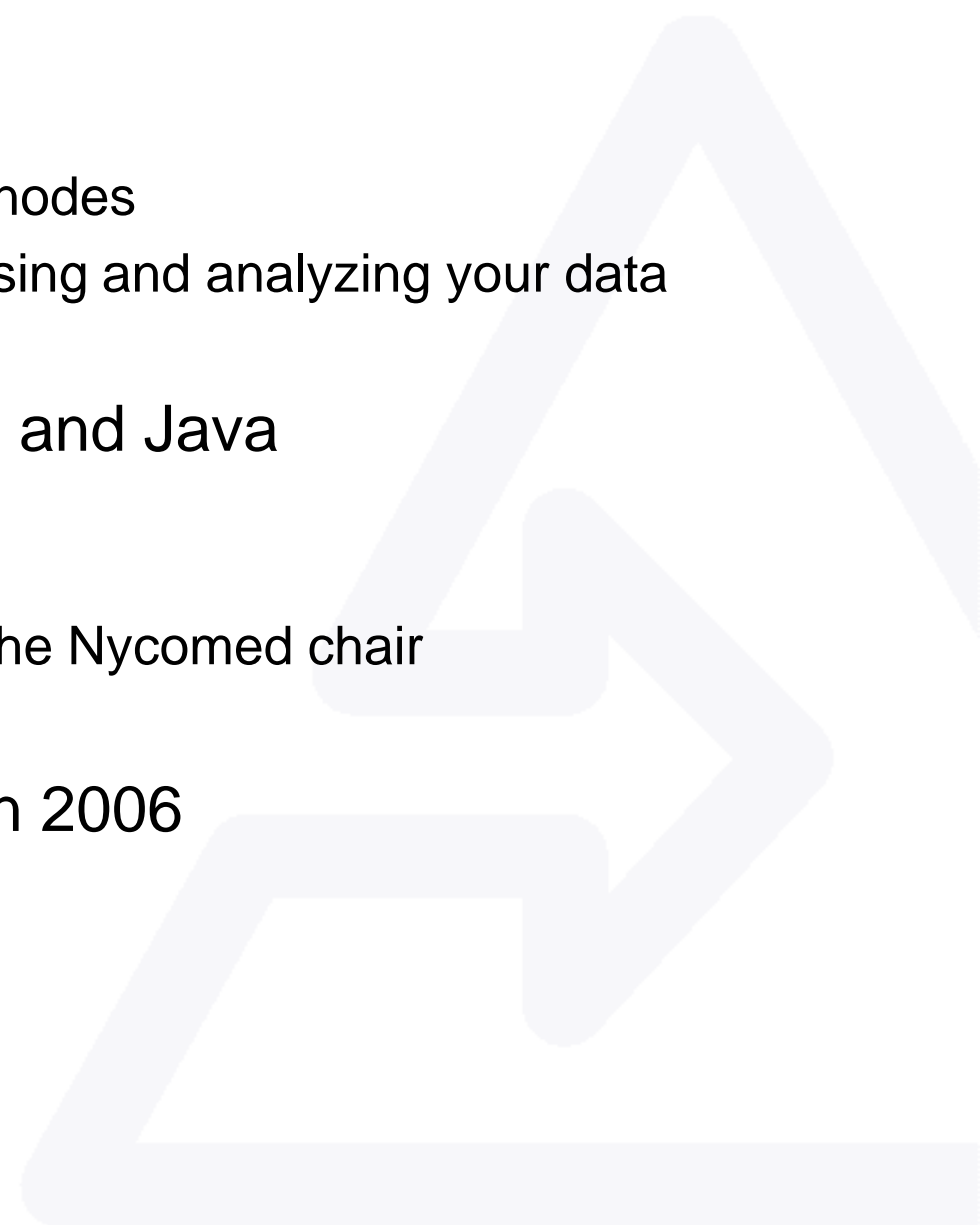
- Introduction
- Installation
- Usage
- Example Workflows
- Development
- Summary





# What is KNIME

- Data exploration platform
  - Data flows (data pipelining)
  - Operations are encapsulated in nodes
  - More than 100 nodes for processing and analyzing your data
- Based on the Eclipse platform and Java
  - Open Source
  - Development started in 2004
  - Developed by KNIME.com and the Nycomed chair
- Public release of version 1.0 in 2006
- More than 5500 active users





# KNIME

The screenshot displays the KNIME software interface with a workflow titled "NGs NCI" and "Meta-workflow: Node 35". The workflow includes nodes such as "File Reader", "Row Filter", "Domain Calculator", "Color Manager", "Prepare NG", "MetaNode 1:1", "Parallel Coordinates", "Scatter Plot", "MoSS", "Interactive Table", "Column Filter", and "Neighborhood Visualizer".

A "Scatter Plot (#49) - Scatterplot" window is open, showing a scatter plot of data points colored by class. The X-axis ranges from 100.0 to 940.0, and the Y-axis ranges from 27.0 to 391.0. The plot shows a dense cluster of points with some outliers.

A "File Reader" node description is visible, stating: "This node can be used to read data from an ASCII file or URL location. It can be configured to read in various formats. When you open the node's configuration dialog and provide a filename, it will try to guess the reader's settings by analyzing the beginning of the file. Check the results of these settings in the preview table. If the data shown is not correct or an error is reported, you can adjust the settings manually (see below). When the node is executed it reads in the entire file and caches it in a temporary file for faster access by the connected successor nodes. It also stores all possible values it came across for each column."

A "Dialog Options" section for the File Reader node is also visible, showing the "ASCII file location" field and a "Browse..." button.

An "Interactive Table (#87) - Table View (58 x 7)" window is open, displaying a table with columns: File, Hilite, Navigation, View, Output. The table contains data for various chemical structures, including a molecule with a benzene ring and a carboxylic acid group.

A "Neighborhood Visualizer (#82) - Interactive Neighborgrams" window is open, showing a grid of plots for different chemical structures. The plots display various metrics, including "Unity FP", "Atompar FP", and "VolSurf".

A "Neighborhood Visualizer (#82) - Interactive Neighborgrams" window is also open, showing a grid of plots for different chemical structures. The plots display various metrics, including "Unity FP", "Atompar FP", and "VolSurf".



# KNIME

KNIME loads and integrates data from diverse data sources:

- Different data bases
- Various file formats (CSV, XML, SDF, etc.)

## File Reader



Excel import

## Database Connector



Node 0: 1:8

## PMML Reader



Vendor independent predictive model

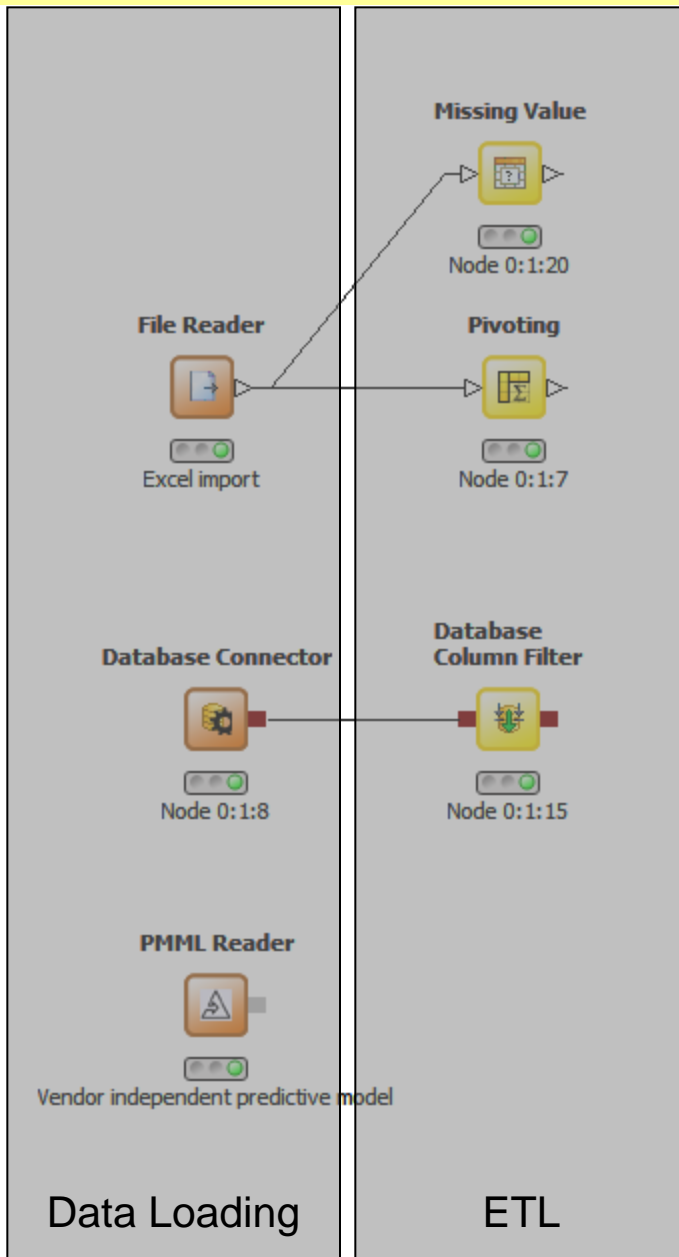
Data Loading



# KNIME

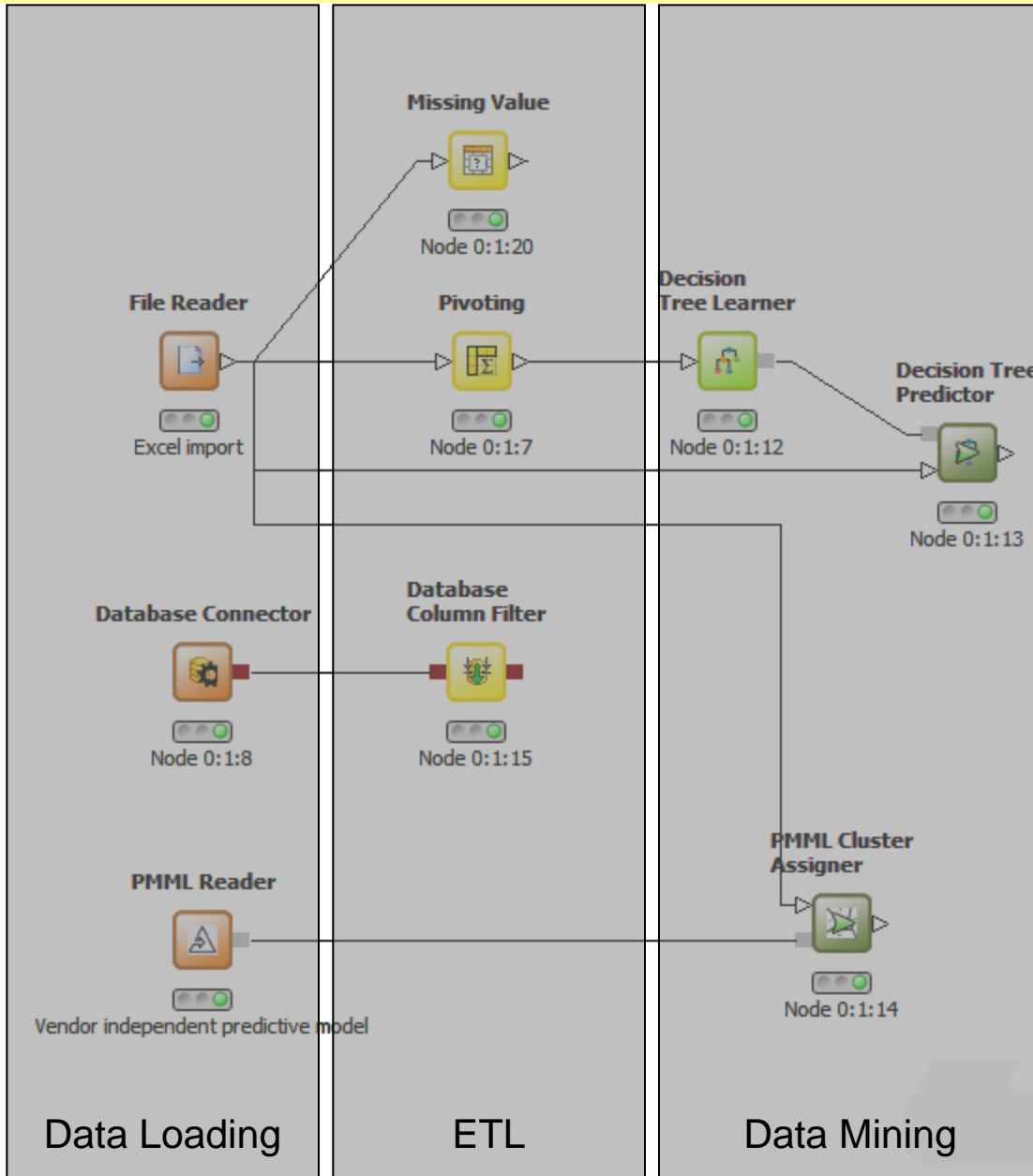
KNIME provides huge repository of modules for easy-to-use, modular

- Data preprocessing
- Data fusion
- Data transformation





# KNIME



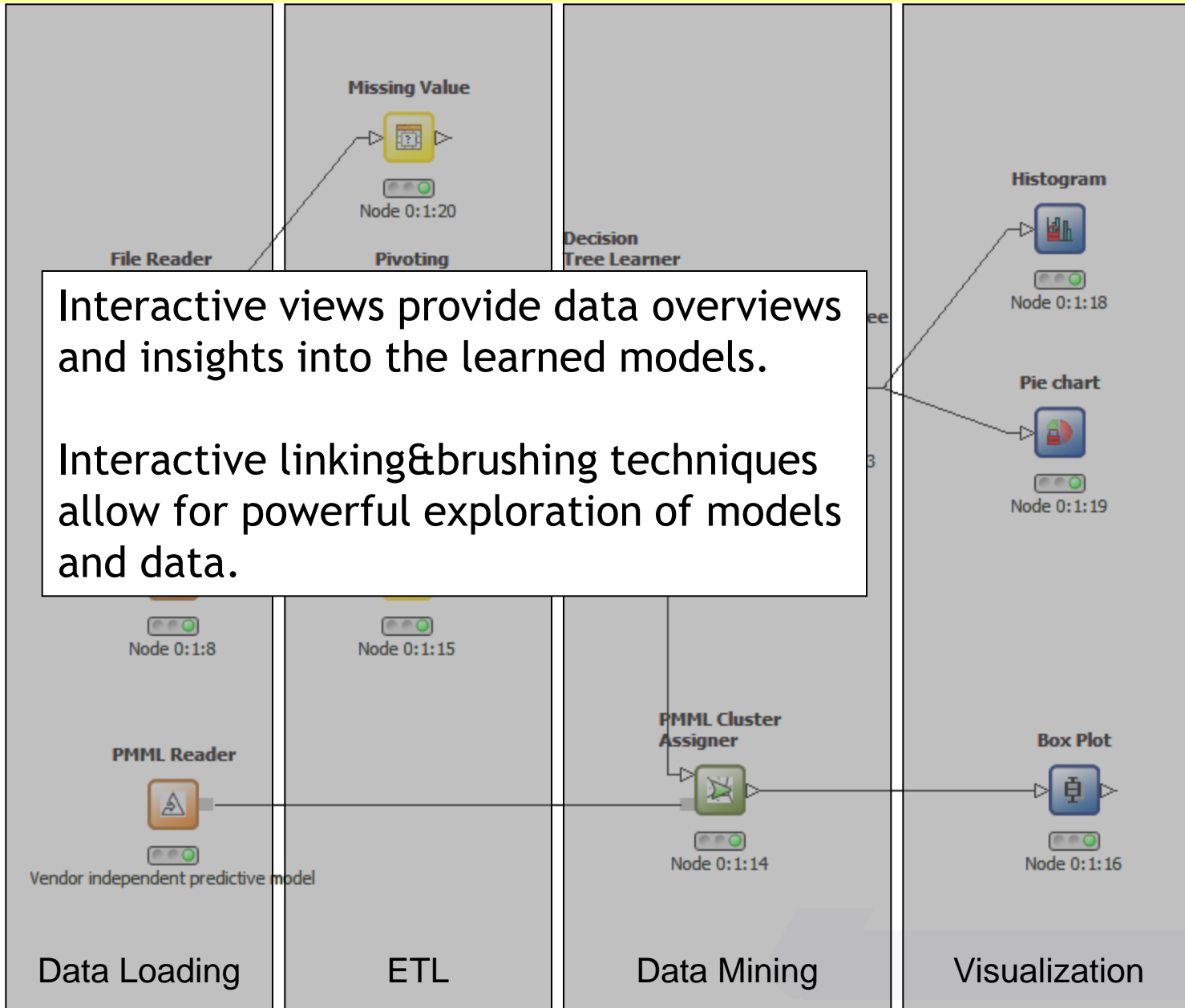
In addition to standard data mining techniques, KNIME adds cutting edge data analysis algorithms. (...thanks to its academic roots)



# KNIME

Interactive views provide data overviews and insights into the learned models.

Interactive linking&brushing techniques allow for powerful exploration of models and data.







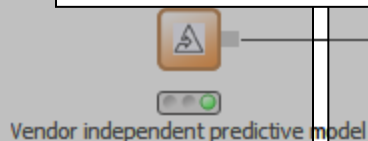
# KNIME

Due to its open API and “node-in-a-sandbox”-approach additional (also external) tools are easily integrated,

e.g.

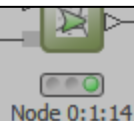
- Item set mining
- Access to the statistics tool R
- Complete integration of the machine learning library WEKA
- Application area specific integration, e.g. CDK (Chemical Development Kit)

KNIME is Eclipse-based: Integrating other Eclipse projects such as BIRT, DTP, etc. provides even more functionality

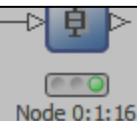


Data Loading

ETL



Data Mining



Visualization

R Snippet (Local)



Node 0:1:21

Logistic



Node 0:1:22

Weka Predictor

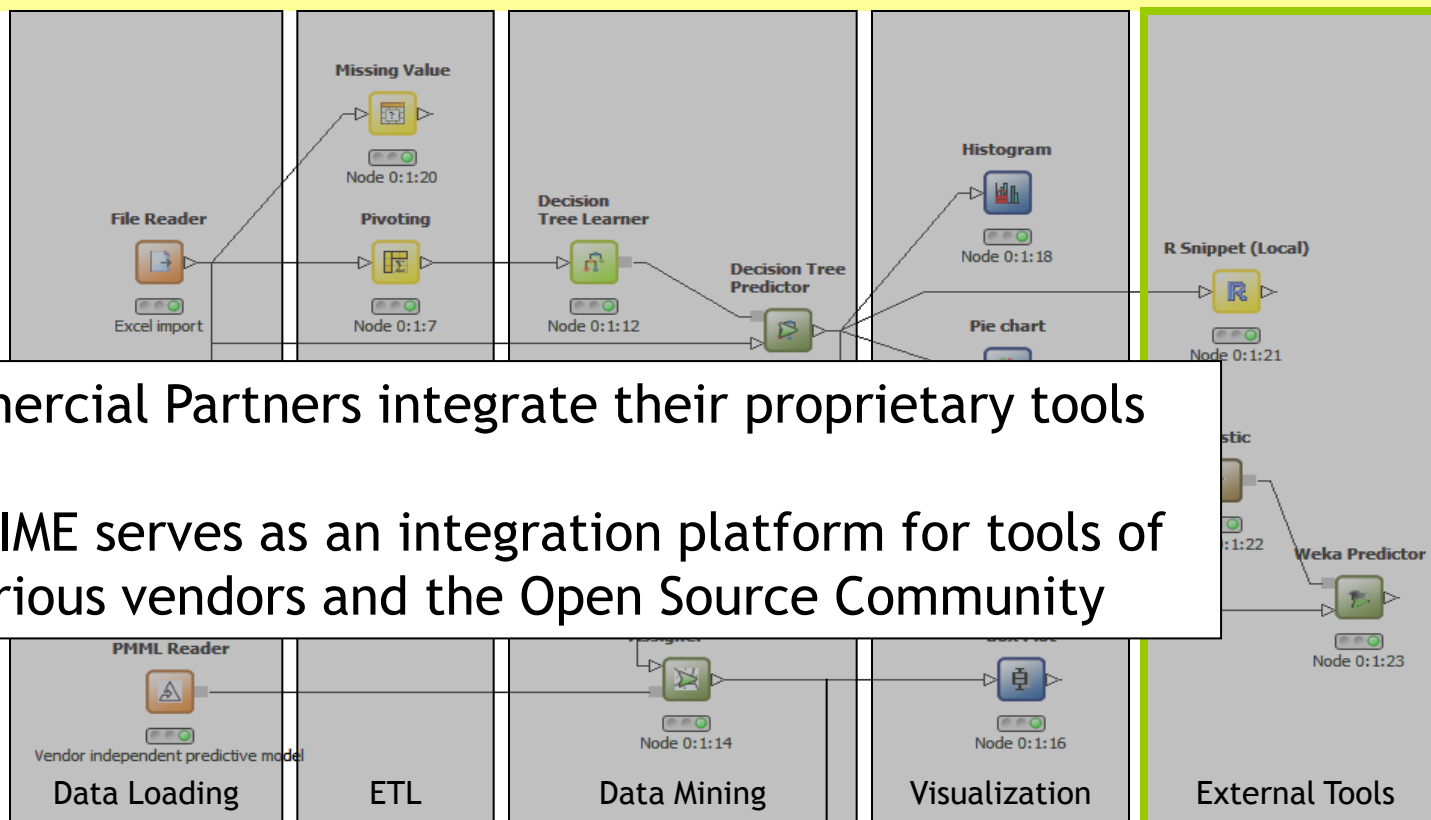


Node 0:1:23

External Tools

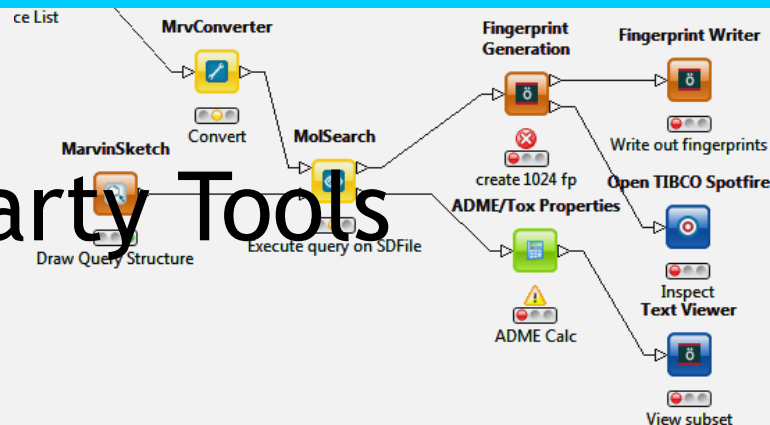


# KNIME



Commercial Partners integrate their proprietary tools  
⇒ KNIME serves as an integration platform for tools of various vendors and the Open Source Community

## 3rd Party Tools





# INSTALLATION





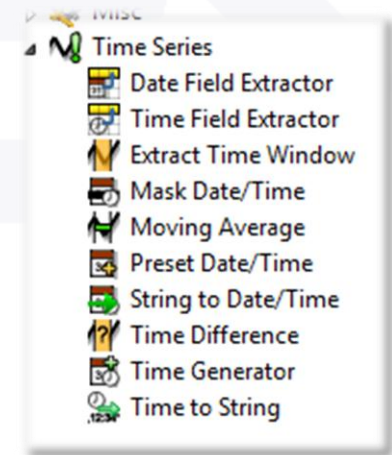
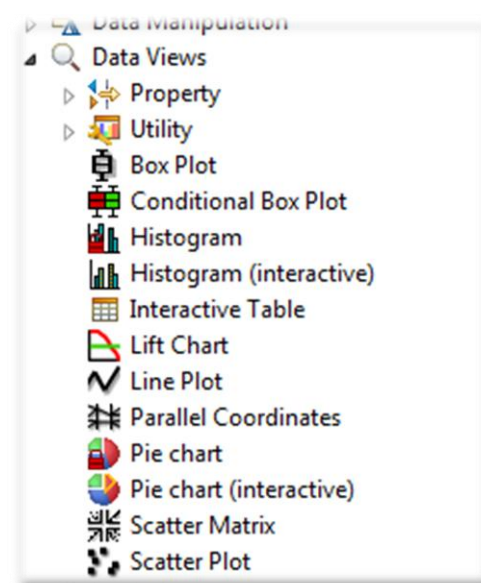
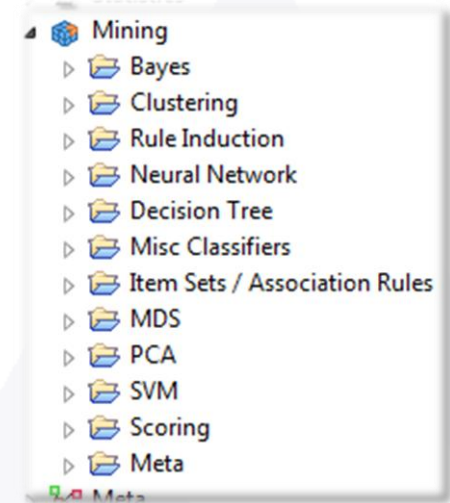
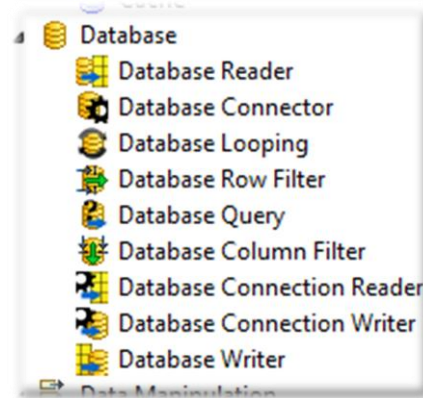
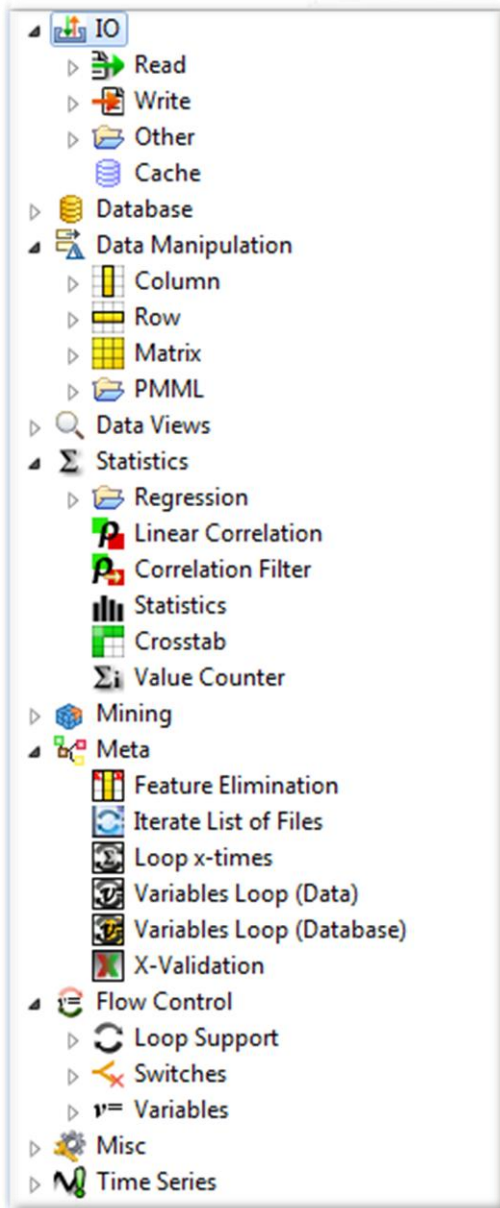
# KNIME Installation

- KNIME: <http://knime.org/download-desktop>
  - KNIME Desktop Version
  - KNIME SDK Developer Version
- Available for:
  - Windows 64/32bit
  - Linux 64/32bit
  - Mac OS X 64bit
- No installation needed simply unpack the downloaded archives





# KNIME Standard Nodes





# KNIME Extensions

- Chemistry types and features
- Distance Matrix
- Ensemble Learning
- External Tool Support
- HTML/PDF Writer
- Item Set Mining
- R Statistics Integration
- Report Designer
- Webservice Client
- Weka Data Mining Integration
- XLS Support
- XML Processing
- ...





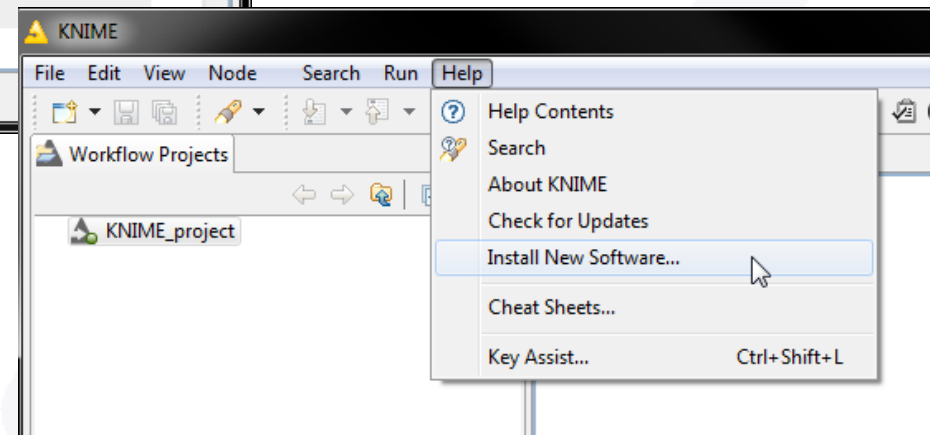
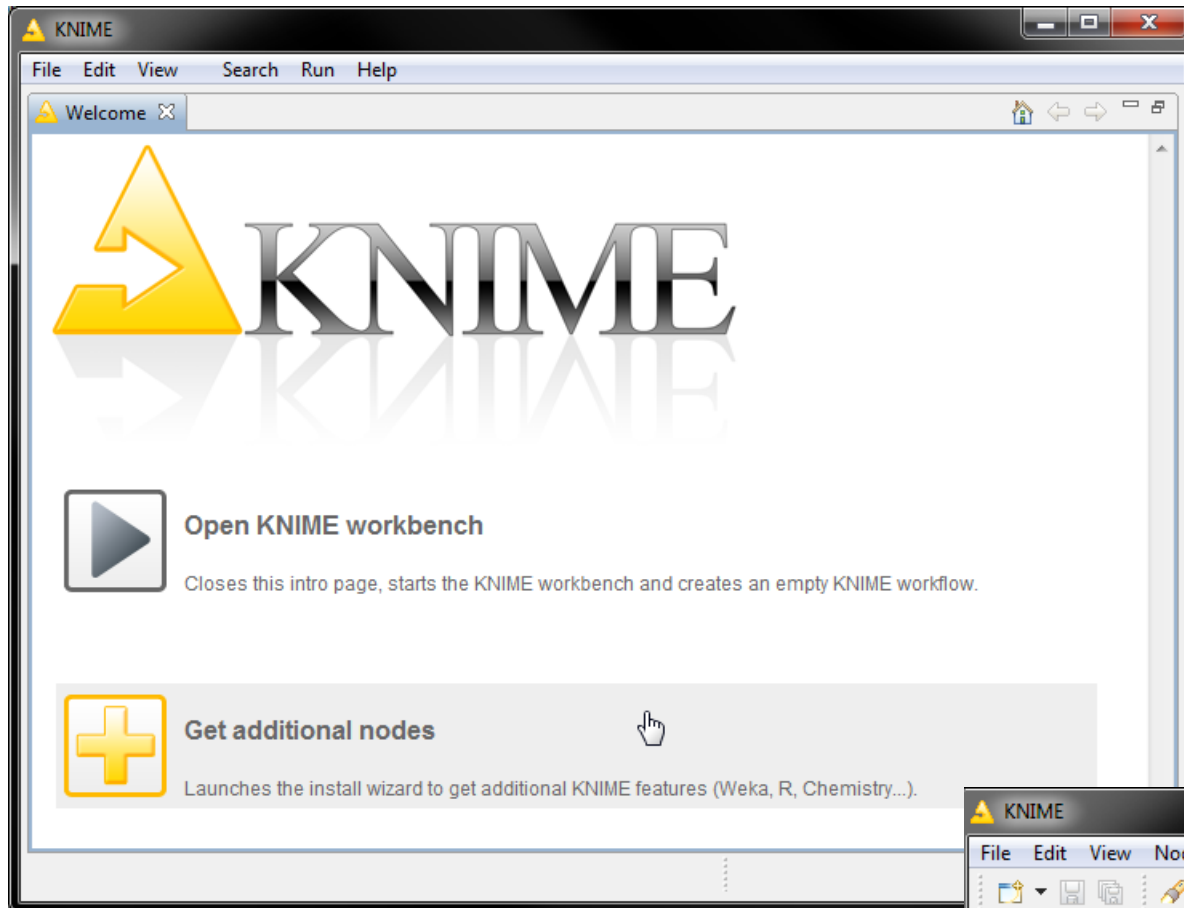
# Additional Extensions

- Experimental Extensions: <http://tech.knime.org/knime-labs>
  - Modular Data Generators
  - Network Mining
  - Perl Scripting
  - Text Processing
  - ...
- Community Contributions: <http://tech.knime.org/community>
  - Chemoinformatics
  - High Content Screening
  - Image Processing
  - Next Generation Sequencing
  - R/Groovy/Matlab/Python Scripting
  - STARK
  - ...





# Extension Installation

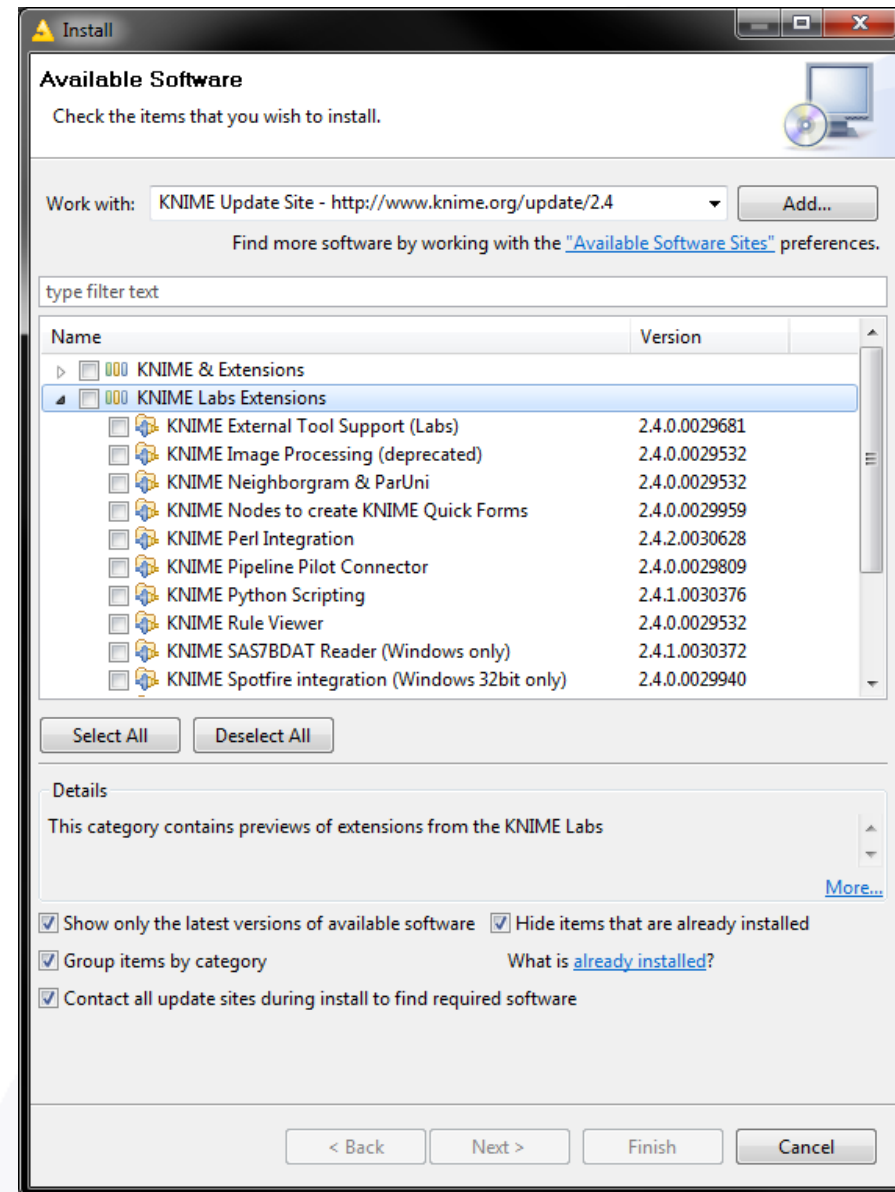






# Extension Installation

- Installation via Eclipse update manager
- KNIME and Labs extensions at
  - <http://www.knime.org/update/2.4>
- Community Contributions at
  - <http://tech.knime.org/update/community-contributions/release>
  - <http://tech.knime.org/update/community-contributions/nightly>





# USAGE



# The KNIME Workbench

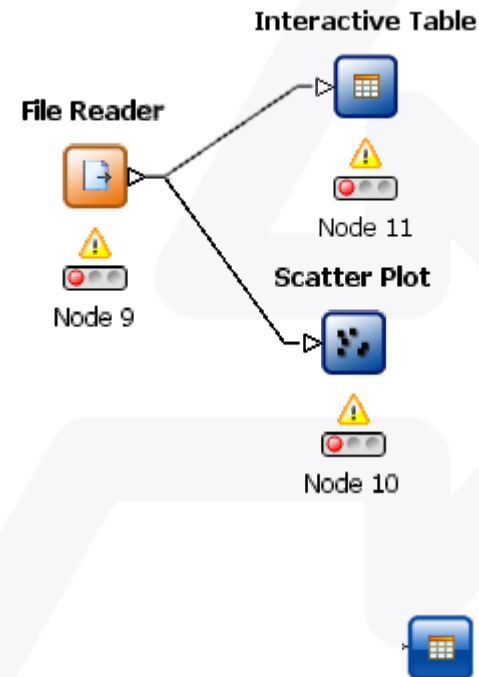
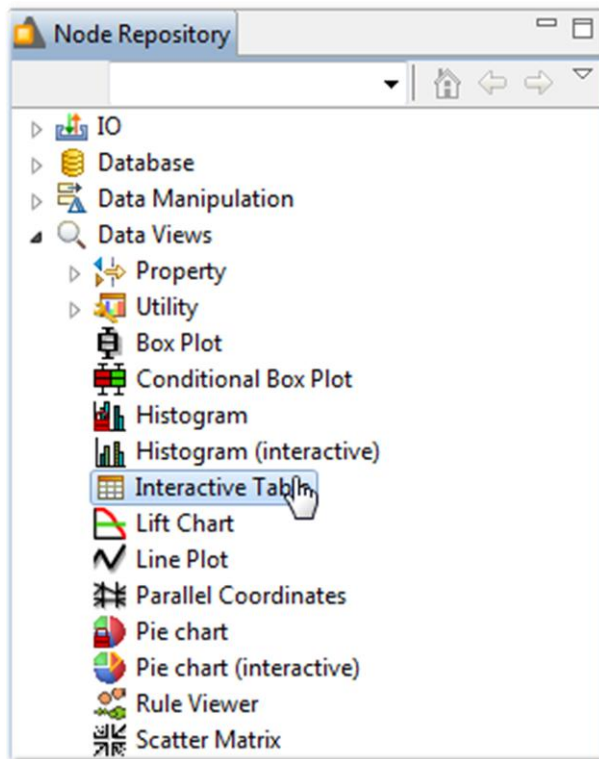
The screenshot shows the KNIME Workbench interface with several components highlighted by yellow boxes and labeled with text:

- Overview of workflows**: Points to the Workflow Projects pane on the left, which lists various workflow projects like '001001\_TimeSeries\_example'.
- Node repository**: Points to the Node Repository pane on the left, which lists various nodes categorized by type (IO, Database, Data Manipulation, etc.).
- Node description**: Points to the Node Description pane on the right, which shows the details for the 'Date Field Extractor' node, including its function and dialog options.
- Workflow area**: Points to the central workspace where a workflow is being built, showing nodes like 'File Reader', 'String to Date/Time', 'Column Filter', 'Date Field Extractor', 'GroupBy', 'Moving Average', 'Missing Value', 'Math Formula', 'Time to String', 'RowID', and 'Line Plot'.
- Console**: Points to the Console pane at the bottom, which displays debug messages from the KNIME engine.
- Outline**: Points to the Outline pane at the bottom right, which shows a hierarchical view of the workflow structure.






# Creating a Simple Flow

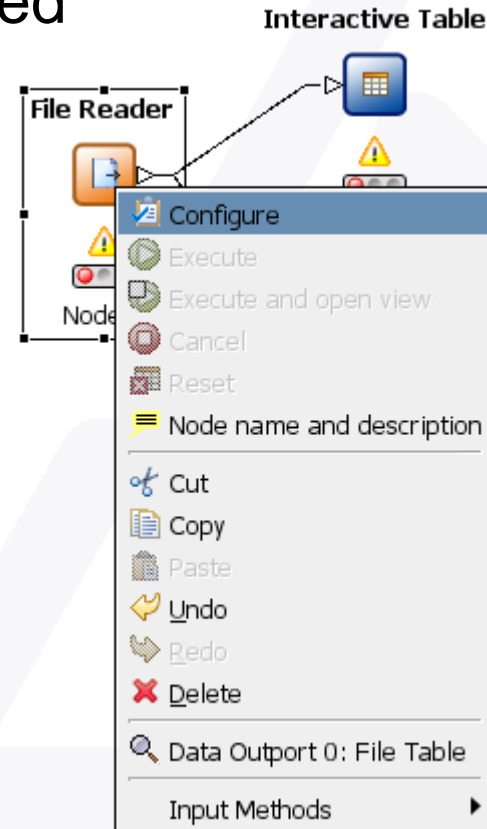
- Drag and drop nodes from the repository to the workflow
- Connect the nodes by “drawing” connections between output and input ports





# Configuring Nodes

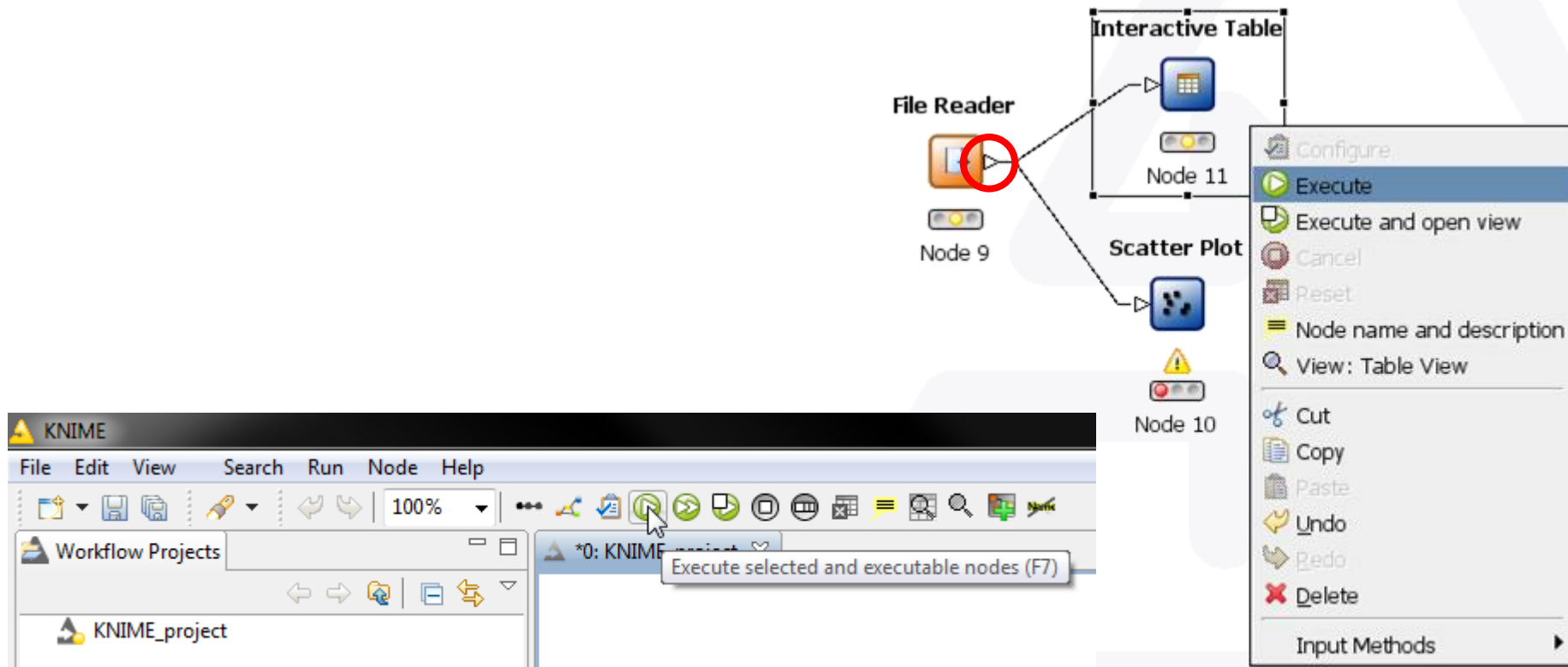
- Right-click on node to get context menu
- *Configure* opens a dialog where parameters can be set
- Not all nodes need to be configured manually
- Node states
  -  Node is not configured properly or not fully connected
  -  Node is ready to be executed
  -  Node is executed and has data at its output port(s)





# Executing Nodes

- Click *Execute* in the context menu or the toolbar button
  - only “yellow” nodes can be executed
- All predecesing nodes are executed automatically





# Data Table

- Contains meta information (spec)
  - data types
  - domains
  - # of rows / cols
- Large tables are buffered on disc
- Blob cell support for large data cells e.g. images

Bit vector data - 4:8 - Bitvector Generator

File

Table "default" - Rows: 150 | Spec - Columns: 5 | Properties

Row ID	D Double ...	S String Col	I Integer...	(...) Collecti...	[010] BitVectors
Row0	0.2	Iris-setosa	1	[0.2,1]	10
Row1	0.2	Iris-setosa	1	[0.2,1]	10
Row2	0.2	Iris-setosa	1	[0.2,1]	10
Row3	0.2	Iris-setosa	1	[0.2,1]	10
Row4	0.2	Iris-setosa	1	[0.2,1]	10
Row5	0.4	Iris-setosa	1	[0.4,1]	10
Row6	0.3	Iris-setosa	1	[0.3,1]	10
Row7	0.2	Iris-setosa	1	[0.2,1]	10
Row8	0.2	Iris-setosa	1	[0.2,1]	10
Row9	0.1	Iris-setosa	1	[0.1,1]	10
Row10	0.2	Iris-setosa	1	[0.2,1]	10
Row11	0.2	Iris-setosa	1	[0.2,1]	10
Row12	0.1	Iris-setosa	1	[0.1,1]	10
Row13	0.1	Iris-setosa	1	[0.1,1]	10
Row14	0.2	Iris-setosa	1	[0.2,1]	10
Row15	0.4	Iris-setosa	1	[0.4,1]	10
Row16	0.4	Iris-setosa	1	[0.4,1]	10
Row17	0.3	Iris-setosa	1	[0.3,1]	10
Row18	0.3	Iris-setosa	1	[0.3,1]	10
Row19	0.3	Iris-setosa	1	[0.3,1]	10
Row20	0.2	Iris-setosa	1	[0.2,1]	10



# Data Types

- Common data types

- Double Value

D Double ...  
0.2

- Int Value

I Integer ...  
1

- String Value

S String Col  
Iris-setosa

- Collections

- Sets
- Lists

(...) Collecti...  
[0.2,1]

- Bit vectors

[010] BitVectors  
10

- Additional data types

- Terms and Documents
- Image
- Network
- Chemical types
  - Molecules i.e. CDK, Smiles, SDF, ...
- Distance Matrix
- Custom data types





# Intermediate Results

- Quick view on the output tables at each executed node

**Partitioning**      **RProp MultiLayerPerceptron Learner**

Configure  
Execute  
Execute and open view  
Cancel  
Reset  
Node name and description  
Cut  
Copy  
Paste  
Undo  
Redo  
Delete  
Data Output 0: First partition (as defined in dialog)  
Data Output 1: Second partition (remaining rows)  
Input Methods

**Partitioning (#17), First partition (as defined in dia**

File

	DataTable	DataTableSpec	DataColumnProperties
15 Rows, 5 Cols	<b>D</b> sepal l...	<b>D</b> sepal ...	<b>D</b> petal le... <b>D</b> petal w... <b>S</b> class
Row17	5.4	3.9	1.3 0.4 Iris-setosa
Row20	5.1	3.8	1.5 0.3 Iris-setosa
Row21	5.4	3.4	1.7 0.2 Iris-setosa
Row30	4.7	3.2	1.6 0.2 Iris-setosa
Row41	5	3.5	1.3 0.3 Iris-setosa
Row49	5.3	3.7	1.5 0.2 Iris-setosa
Row52	6.4	3.2	4.5 1.5 Iris-versic...
Row64	6.1	2.9	4.7 1.4 Iris-versic...
Row67	5.6	3	4.5 1.5 Iris-versic...
Row93	5.8	2.6	4 1.2 Iris-versic...
Row102	5.8	2.7	5.1 1.9 Iris-virginica
Row108	7.3	2.9	6.3 1.8 Iris-virginica
Row118	7.7	3.8	6.7 2.2 Iris-virginica
Row119	7.7	2.6	6.9 2.3 Iris-virginica
Row147	6.3	2.5	5 1.9 Iris-virginica



# Learners

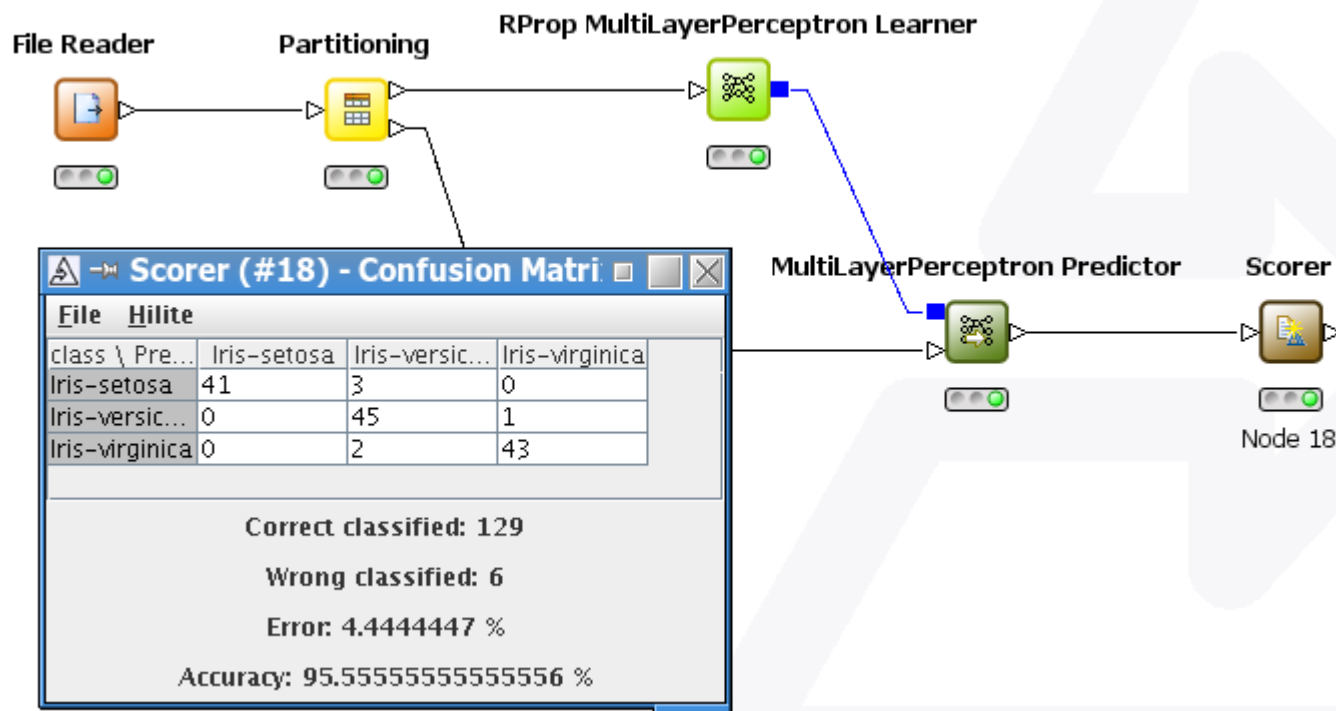
- Learners build “models” based on the input data
- Many different algorithms
  - Decision Trees
  - Neural Networks
  - Regression
  - All of Weka
  - ...
- Model output port





# Predictors

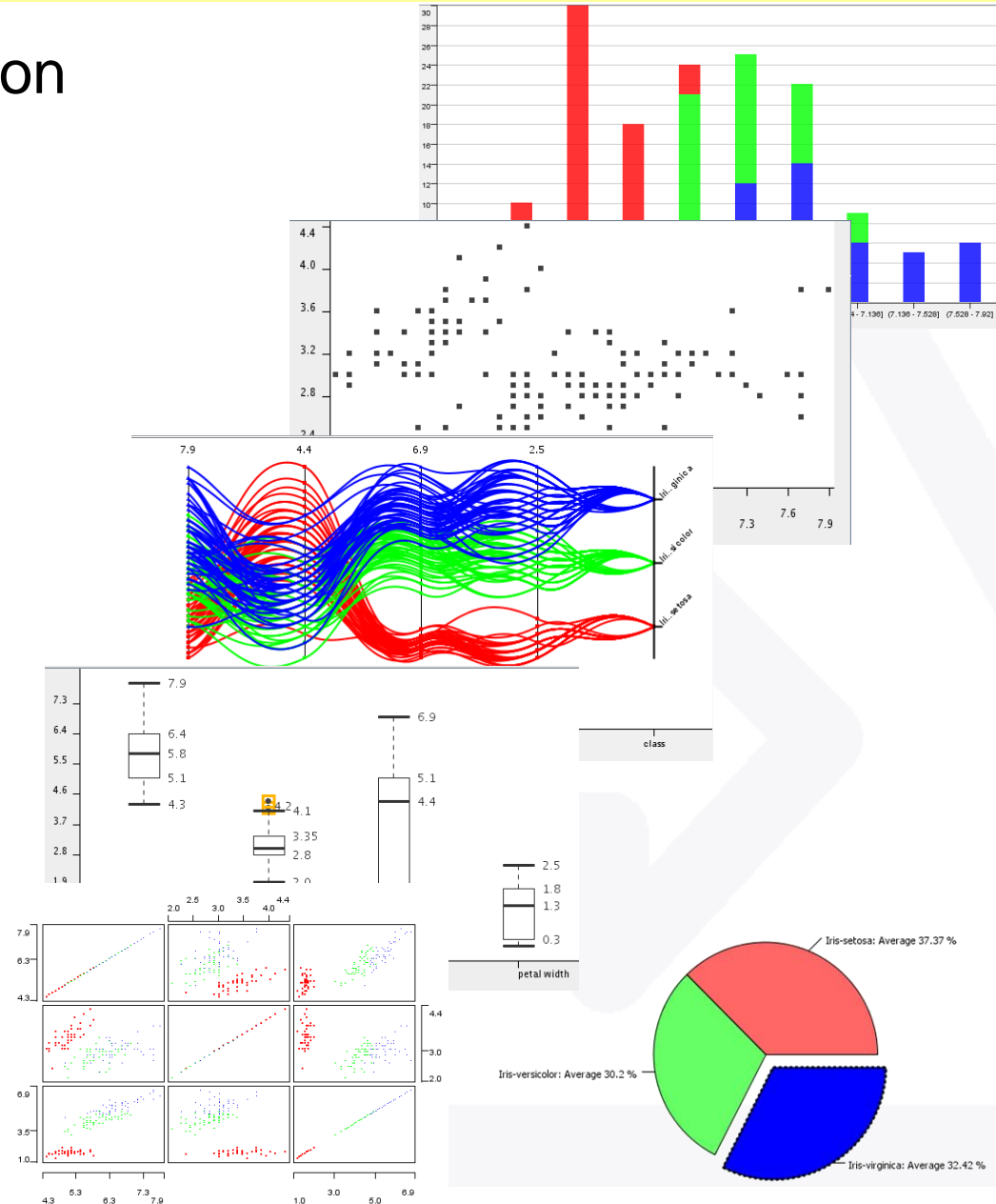
- Predictors predict class values
- Take a model from a learner and suitable data





# Views

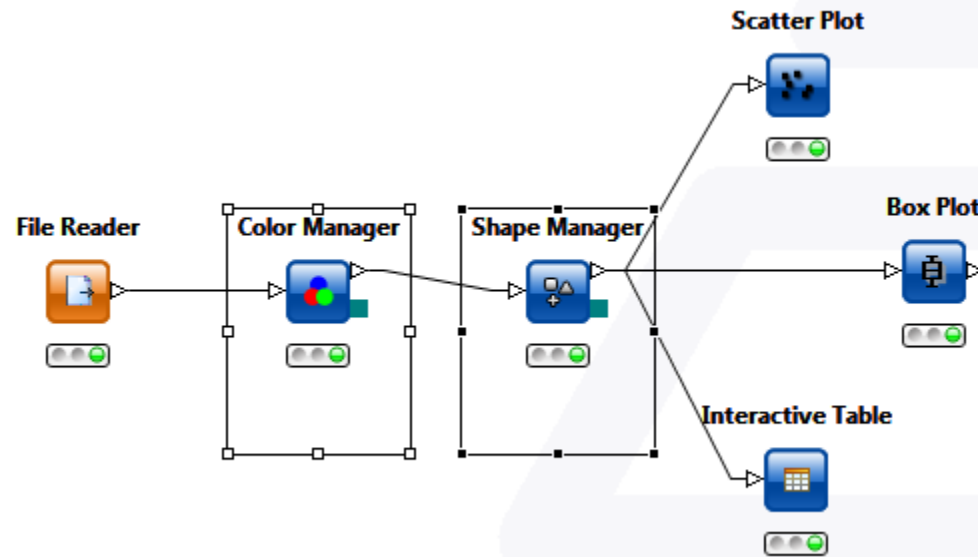
- Many nodes offers views on the input data
  - Scatter Plot, Scatter Matrix
  - Histograms
  - Interactive Table
  - ...
- Click *Open View* in the context menu
- Some nodes have more than one view





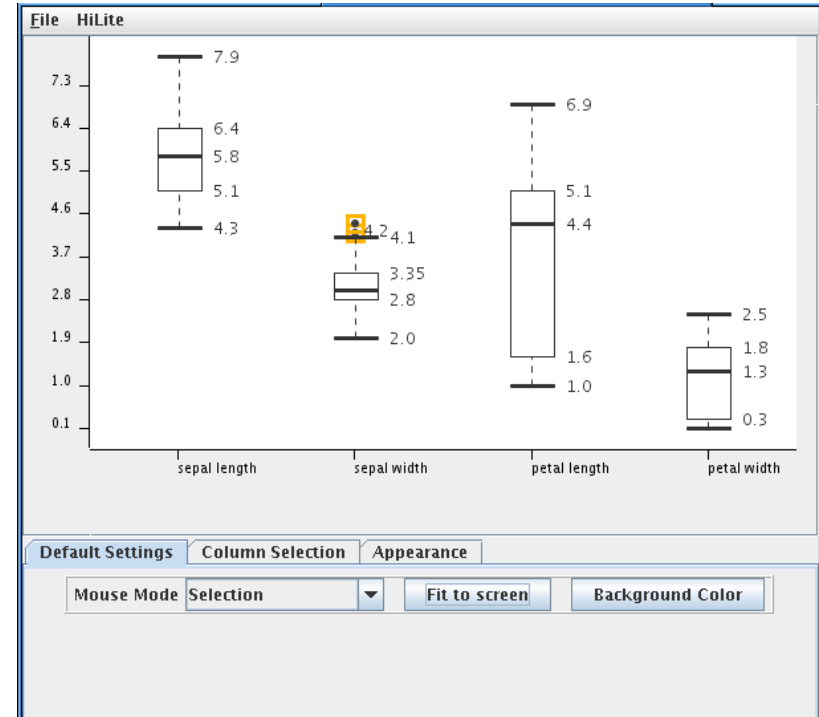
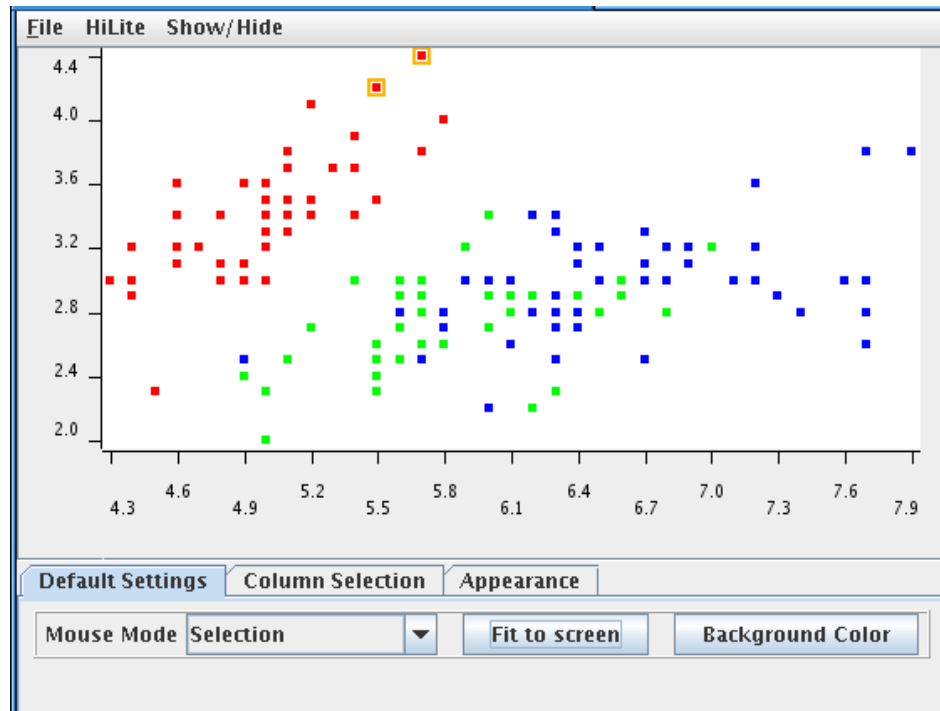
# Adding Color and Shape

- Data rows can be assigned colors and shapes
- Based on a certain column
- Special *Color Manager* and *Shape Manager* nodes
- Accessible in all following nodes





# Linked Brushing

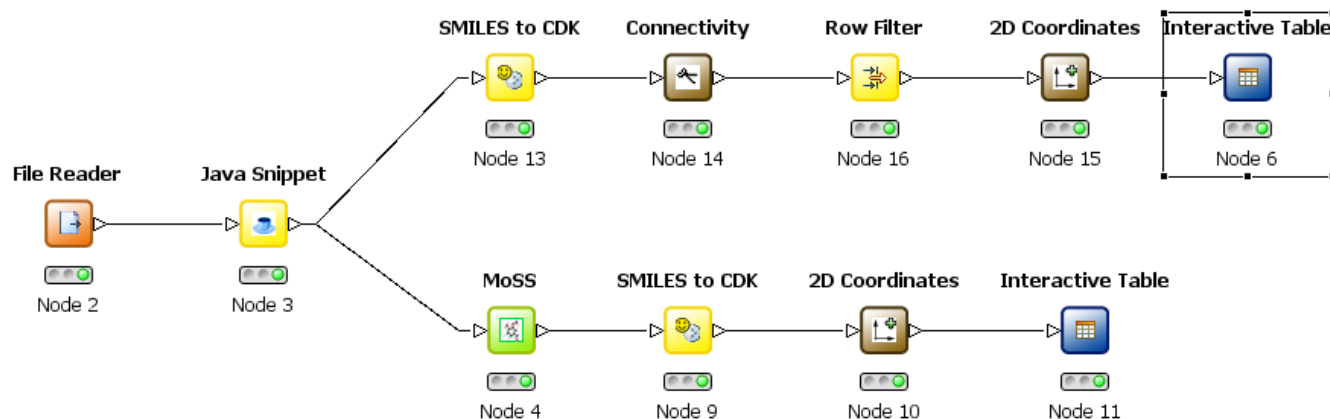


- Hilting can be changed from every view in the flow that supports it
- Views can show only hilited rows

File	HiLite	Navigation	View	Output
	Hilite Selected		sepal ...	D petal le...
	Unhilite Selected		4	1.5
	Clear Hilite		2	1.4
	<input checked="" type="checkbox"/> Show Hilited Only			0.2

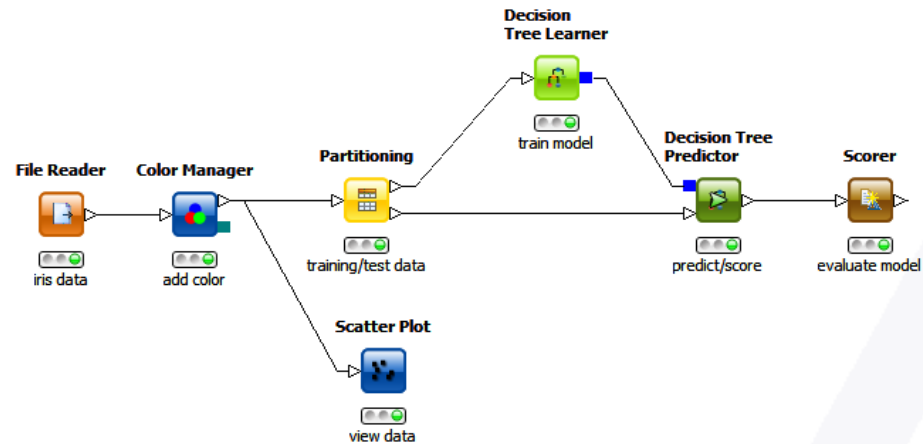


# 1:n Hilting



Interactive Table (#11) - Table View (7389 x 7) <2>						
File	Hilite	Navigation	View	Output		
Key	Fragment	Atom c...	Bond c...	Suppor...	Suppor...	Suppor...
12		8	7	1530	0	0.036
13		6	5	1309	0	0.031
14		10	9	2779	0	0.065
15		6	6	1916	0	0.045

Interactive Table (#6) - Table View (199:		
File	Hilite	Navigation
Key	Mol	S Act
652160		A
652161		A
652162		A



# EXAMPLE WORKFLOWS





# Example Workflow (step by step)

The screenshot displays the KNIME software interface. The main workspace shows a workflow project named "001001\_TimeSeries\_example" with a single node, "File Reader", which is configured to "Read US dollar rates". The left sidebar contains a "Node Repository" with various categories like IO, Database, Data Manipulation, etc. The right sidebar shows the "Node Description" for the "File Reader" node, explaining its functionality and providing "Dialog Options". The bottom panel includes a "Console" window showing debug messages and an "Outline" window.

**File Reader**

This node can be used to read data from an ASCII file or URL location. It can be configured to read various formats. When you open the node's configuration dialog and provide a filename, it tries to guess the reader's settings by analyzing the content of the file. Check the results of these settings in the preview table. If the data shown is not correct or an error is reported, you can adjust the settings manually (see below).

The file analysis runs in the background and can be cut short by clicking the "Quick scan", which shows if the analysis takes longer. In this case the file is not analyzed completely, but only the first fifty lines are taken into account. It could happen then, that the preview appears looking fine, but the execution of the File Reader fails, when it reads the lines it didn't analyze. Thus it is recommended you check the settings, when you cut an analysis short.

**Dialog Options**

KNIME Example Flow Server

Workflow Server:

Status: connected as 'guest'

001\_TimeSeries  
001001\_TimeSeries\_example  
002\_DataMining

**KNIME Console**

```
DEBUG KnimeResourceNavigator Workflow Moving Average 0:2:8 removed
DEBUG WorkflowEditor Workflow event triggered: WorkflowEvent [type=NODE_REMOVED;
DEBUG WorkflowRootEditPart WorkflowRoot: workflow changed, refreshing_children
DEBUG NodeContainerEditPart Missing Value 0:2:10 (IDLE)
DEBUG NodeContainerEditPart Math Formula 0:2:15 (IDLE)
```



# Example Workflow (step by step)

The screenshot displays the KNIME software interface. The main workspace shows a workflow project named "001001\_TimeSeries\_example" with a "File Reader" node. A context menu is open over the "File Reader" node, showing options like "Configure...", "Execute", "Cancel", "Reset", "Edit Node Name and Description...", "New Workflow Annotation", "Collapse into Meta Node", "Expand Meta Node", "Show Flow Variable Ports", "Cut", "Copy", "Paste", "Undo", "Redo", "Delete", and "File Table".

The left sidebar contains the "Workflow Projects" and "Node Repository" panels. The "Node Repository" shows a tree structure of nodes under the "IO" category, including "Read" (File Reader, ARFF Reader, CSV Reader, Line Reader, Table Reader, PMML Reader, XLS Reader, Model Reader) and "Write" (Cache, Database, Data Manipulation, Data Views, Statistics, Mining, Distance Matrix, Meta, Flow Control, Misc, KNIME Labs, Time Series, Reporting, Weka, XML).

The right sidebar shows the "Node Description" panel for the "File Reader" node. It contains the following text:

### File Reader

This node can be used to read data from an ASCII file or URL location. It can be configured to read various formats.

When you open the node's configuration dialog and provide a filename, it tries to guess the reader's settings by analyzing the content of the file. Check the results of these settings in the preview table. If the data shown is not correct or an error is reported, you can adjust the settings manually (see below).

The file analysis runs in the background and can be cut short by clicking the "Quick scan", which shows if the analysis takes longer. In this case the file is not analyzed completely, but only the first fifty lines are taken into account. It could happen then, that the preview appears looking fine, but the execution of the File Reader fails, when it reads the lines it didn't analyze. Thus it is recommended you check the settings, when you cut an analysis short.

### Dialog Options

Below the description, there is a "KNIME Example Flow Server" section with a "Workflow Server" field set to "publicserver.knime.org", a "Status" field set to "connected as 'guest'", and a "Disconnect" button. Below this, there is a list of workflow projects: "001\_TimeSeries", "001001\_TimeSeries\_example", and "002\_DataMining".

The bottom of the interface shows the "Console" and "Outline" panels. The "Console" panel displays the following log messages:

```
KNIME Console
DEBUG NodeContainer File Reader 0:2:1 has new state: CONFIGURED
DEBUG KnimeResourceNavigator state changed to CONFIGURED
DEBUG NodeContainer 001001_TimeSeries_example 0:2 has new state: CONFIGURED
DEBUG NodeContainer 001001_TimeSeries_example 0:2 has new state: CONFIGURED
DEBUG NodeContainer Workflow Manager 0 has new state: IDLE
```



# Example Workflow (step by step)

The screenshot displays the KNIME software interface. The main workspace shows a workflow project named "001001\_TimeSeries\_example" with a single node, "File Reader", which is configured to "Read US dollar rates". The left sidebar contains a "Node Repository" with various nodes categorized under "IO", "Read", "Write", "Other", "Cache", "Database", "Data Manipulation", "Data Views", "Statistics", "Mining", "Distance Matrix", "Meta", "Flow Control", "Misc", "KNIME Labs", "Time Series", "Reporting", "Weka", and "XML". The right sidebar shows the "Node Description" for the "File Reader" node, which explains its functionality and provides "Dialog Options". The bottom status bar shows the "Console" with debug messages indicating the workflow execution status.

**File Reader**

This node can be used to read data from an ASCII file or URL location. It can be configured to read various formats. When you open the node's configuration dialog and provide a filename, it tries to guess the reader's settings by analyzing the content of the file. Check the results of these settings in the preview table. If the data shown is not correct or an error is reported, you can adjust the settings manually (see below).

The file analysis runs in the background and can be cut short by clicking the "Quick scan", which shows if the analysis takes longer. In this case the file is not analyzed completely, but only the first fifty lines are taken into account. It could happen then, that the preview appears looking fine, but the execution of the File Reader fails, when it reads the lines it didn't analyze. Thus it is recommended you check the settings, when you cut an analysis short.

**Dialog Options**

KNIME Example Flow Server

Workflow Server: [publicserver.knime.org](http://publicserver.knime.org)

Status: connected as 'guest' [Disconnect]

001\_TimeSeries  
001001\_TimeSeries\_example  
002\_DataMining

KNIME Console

```
DEBUG NodeContainer 001001_TimeSeries_example 0:2 has new state: EXECUTED
DEBUG NodeContainer Workflow Manager 0 has new state: IDLE
DEBUG NodeContainer 001001_TimeSeries_example 0:2 has new state: EXECUTED
DEBUG OpenPortViewAction Open Port View File Reader (#1)
DEBUG OpenPortViewAction Open Port View File Reader (#1)
```



# Example Workflow (step by step)

The screenshot shows the KNIME software interface. On the left, the 'Workflow Projects' pane lists several projects, including '001001\_TimeSeries\_example'. The 'Node Repository' pane on the bottom left shows various nodes categorized by function, such as 'IO', 'Database', and 'Data Manipulation'. The 'File Reader' node is highlighted in the 'IO' category. The main workspace displays a workflow diagram with a 'File Reader' node labeled 'Read US dollar rates'. A pop-up window titled 'File Table - 2:1 - File Reader (Read US dollar rates)' shows the data loaded from the file 'forex.dat'. The data table has 4959 rows and 9 columns. The columns are labeled: 'S OBS', 'D AUSTRUS', 'D BRITPUS', 'D CONDLUS', 'D DTCHGUS', 'D FRNFRUS', 'D GERMDUS', 'D JAPYUS', and 'D SWISFUS'. The rows represent time series data for various countries, with dates ranging from 12/31/1979 to 2/19/1980.

Row ID	S OBS	D AUSTRUS	D BRITPUS	D CONDLUS	D DTCHGUS	D FRNFRUS	D GERMDUS	D JAPYUS	D SWISFUS
Row0	12/31/1979	0.904	0.451	1.168	1.902	4.018	1.726	240.3	1.595
Row1	1/01/1980	?	?	?	?	?	?	?	?
Row2	1/02/1980	0.903	0.446	1.166	1.892	4.015	1.714	238.45	1.578
Row3	1/03/1980	0.901	0.446	1.17	1.89	4.01	1.711	238.35	1.572
Row4	1/04/1980	0.902	0.447	1.169	1.891	4.025	1.713	234.8	1.576
Row5	1/07/1980	0.898	0.442	1.169	1.889	4.009	1.708	231.55	1.57
Row6	1/08/1980	0.899	0.442	1.17	1.892	4.012	1.713	234.75	1.577
Row7	1/09/1980	0.898	0.441	1.169	1.892	4.012	1.712	234.95	1.574
Row8	1/10/1980	0.899	0.443	1.165	1.895	4.023	1.717	235.8	1.579
Row9	1/11/1980	0.9	0.443	1.165	1.899	4.032	1.721	236.05	1.583
Row10	1/14/1980	0.9	0.439	1.166	1.898	4.032	1.719	236.2	1.583
Row11	1/15/1980	0.902	0.439	1.166	1.906	4.048	1.728	239.65	1.595
Row12	1/16/1980	0.9	0.44	1.164	1.898	4.032	1.722	238.15	1.585
Row13	1/17/1980	0.901	0.439	1.161	1.902	4.038	1.724	239.9	1.593
Row14	1/18/1980	0.901	0.438	1.162	1.9	4.036	1.724	239.65	1.593
Row15	1/21/1980	0.901	0.438	1.161	1.908	4.053	1.729	241	1.604
Row16	1/22/1980	0.902	0.44	1.159	1.912	4.059	1.734	239.7	1.609
Row17	1/23/1980	0.9	0.439	1.16	1.91	4.05	1.73	237.25	1.606
Row18	1/24/1980	0.901	0.439	1.157	1.912	4.058	1.732	239.35	1.608
Row19	1/25/1980	0.902	0.442	1.161	1.918	4.064	1.736	239.6	1.616
Row20	1/28/1980	0.903	0.445	1.164	1.918	4.067	1.739	240.22	1.619
Row21	1/29/1980	0.904	0.445	1.164	1.918	4.066	1.737	239.35	1.617
Row22	1/30/1980	0.903	0.442	1.162	1.917	4.066	1.737	239.5	1.622
Row23	1/31/1980	0.904	0.441	1.158	1.927	4.084	1.744	239.33	1.638
Row24	2/01/1980	0.905	0.44	1.156	1.926	4.084	1.744	240.45	1.634
Row25	2/04/1980	0.904	0.437	1.159	1.926	4.083	1.742	240.29	1.631
Row26	2/05/1980	0.903	0.437	1.162	1.924	4.082	1.742	240.1	1.626
Row27	2/06/1980	0.902	0.433	1.16	1.914	4.059	1.733	240.1	1.606
Row28	2/07/1980	0.902	0.434	1.159	1.918	4.07	1.738	240.09	1.616
Row29	2/08/1980	0.903	0.435	1.159	1.919	4.075	1.739	241.65	1.619
Row30	2/11/1980	0.904	0.434	1.159	1.918	4.072	1.739	241.4	1.617
Row31	2/12/1980	?	?	?	?	?	?	?	?
Row32	2/13/1980	0.904	0.433	1.162	1.911	4.064	1.735	242.1	1.613
Row33	2/14/1980	0.904	0.432	1.16	1.912	4.068	1.736	243.39	1.617
Row34	2/15/1980	0.899	0.436	1.162	1.918	4.077	1.742	243.2	1.629
Row35	2/18/1980	?	?	?	?	?	?	?	?
Row36	2/19/1980	0.906	0.44	1.16	1.925	4.098	1.748	245.36	1.638



# Example Workflow (step by step)

The screenshot displays the KNIME software interface. The main workspace shows a workflow with two nodes: 'File Reader' (labeled 'Read US dollar rates') and 'String to Date/Time' (labeled 'change string to date column'). A dialog box titled 'Dialog - 2:2 - String to Date/Time (change string to dat...' is open, showing the 'Options' tab. The dialog includes a 'Select column' dropdown set to '\$ OBS', a checked 'Replace selected column' checkbox, a 'New column name' field containing 'OBS\_time', and a 'Date format' dropdown set to 'MM/dd/yyyy'. There is also an 'Abort execution' section with a checked checkbox and a value of 100. The right sidebar contains a 'Node Description' panel for the 'String to Date/Time' node, explaining its function and providing 'Dialog Options' such as 'Select column', 'Replace selected column', 'New column name', and 'Date format'. The bottom of the interface features a 'Console' window with debug logs and an 'Outline' window.

**String to Date/Time**

Parses the strings in the selected String column and converts them into date/time cells by using the entered format pattern.

**Dialog Options**

**Select column**  
Select the input column containing the timestamp string.

**Replace selected column**  
Check if the string column is not needed anymore and should be replaced by the parsed and converted date/time values.

**New column name**  
If the string column should not be replaced enter here the name for the new column containing the parsed and converted date/times.

**Date format**  
A format string as required by the

KNIME Console

```
DEBUG KnimeResourceNavigator Workflow Extract Time Window 0:2:5 removed
DEBUG WorkflowEditor Workflow event triggered: WorkflowEvent [type=NODE_REMOVED;
DEBUG WorkflowRootEditPart WorkflowRoot: workflow changed, refreshing children
DEBUG NodeContainerEditPart String to Date/Time 0:2:2 (CONFIGURED)
DEBUG OpenFileDialogAction Opening node dialog...
```



# Example Workflow (step by step)

The screenshot displays the KNIME software interface. The main workspace shows a workflow with two nodes: 'File Reader' and 'String to Date/Time'. The 'File Reader' node is labeled 'Read US dollar rates' and the 'String to Date/Time' node is labeled 'change string to date column'. The 'String to Date/Time' node's description is visible on the right, stating: 'Parses the strings in the selected String column and converts them into date/time cells by using the entered format pattern.'

A data table titled 'Parsed time - 2:2 - String to Date/Time (change string to date column)' is shown in the foreground. The table has 9 columns: 'Row ID', 'OBS', 'D\_AUSTRUS', 'D\_BRITPUS', 'D\_CONDLUS', 'D\_DTCHGUS', 'D\_FRNFRUS', and 'D'. The rows show dates from 11.Feb.1980 to 21.Mar.1980 and corresponding numerical values.

The KNIME Console at the bottom shows the following log entries:

```
DEBUG KnimeResourceNavigator stat
DEBUG NodeContainer 001001_TimeS
DEBUG NodeContainer 001001_TimeS
DEBUG NodeContainer Workflow Manager 0 has new state: idle
DEBUG OpenPortViewAction Open Port View String to Date/Time (#1)
```





# Example Workflow (step by step)

The screenshot displays the KNIME software interface. On the left, the 'Workflow Projects' pane shows a list of projects, including '001001\_TimeSeries\_example'. Below it, the 'Node Repository' pane lists various nodes categorized by function, such as IO, Database, Data Manipulation, and Time Series. The main workspace shows a workflow with three nodes: 'File Reader' (labeled 'Read US dollar rates'), 'String to Date/Time' (labeled 'change string to date column'), and 'Extract Time Window' (labeled 'only years 85 - 95'). A 'Dialog - 2:5 - Extract Time Window (on...)' window is open in the foreground, showing the 'Options' tab. This dialog allows users to specify the columns containing timestamps, the starting point (Year: 1985, Month: 1, Day: 1), and the end point (Year: 1995, Month: 12, Day: 31). The 'Node Description' pane on the right provides detailed information about the 'Extract Time Window' node, including its purpose, dialog options, and input ports. The 'KNIME Console' at the bottom shows debug messages related to the workflow execution.

**Extract Time Window**

Extracts all rows where the time value of the selected column lies within a given time window from the input. The time window is specified by a start and end date (and time).

**Dialog Options**

**Columns containing date/time**  
Select the input column containing date/time stamps.

**Starting point**  
Start date/time of the time window.

**End point:**  
End date/time of the time window.

**Ports**

**Input Ports**

KNIME Example Flow Server

Workflow Server: publicserver.knime.org

Status: connected as 'guest' [Disconnect]

001\_TimeSeries  
001001\_TimeSeries\_example  
002\_DataMining

KNIME Console

```
DEBUG ConnectionContainerEditPart modelli
DEBUG WorkflowRootEditPart part: NodeConta
DEBUG ConnectionContainerEditPart refresh
DEBUG ConnectionContainerEditPart modelli
DEBUG OpenFileDialogAction Opening node di
```



# Example Workflow (step by step)

The screenshot displays the KNIME software interface. The main window shows a workflow with four nodes: **File Reader** (Read US dollar rates), **String to Date/Time** (change string to date column), **Extract Time Window** (only years 85 - 95), and **Column Filter** (only german rates). A dialog box titled "Dialog - 2:6 - Column Filter (only german rates)" is open, showing the "File" tab. The "Exclude" list contains various country codes (AUSTRUS, BRITPUS, CDNDLUS, DTCHGUS, FRNFRUS, JAPYNUS, SWISFUS). The "Include" list contains "GERMDUS". The "Enforce inclusion" radio button is selected. The "Node Description" panel on the right provides details about the "Column Filter" node, including its purpose and "Dialog Options". The "KNIME Console" at the bottom shows debug messages, and the "Outline" panel on the right shows the workflow structure.

**Column Filter**

This node allows columns to be filtered from the input table while only the remaining columns are passed to the output table. Within the dialog, columns can be moved between the Include and Exclude list.

**Dialog Options**

**Include**

This list contains the column names that are included in the output table.

**Enforce Inclusion**

Select this option to enforce the current inclusion list to stay the same even if the input table specification changes. If some of the included columns are not available anymore, a warning is displayed. (New columns will automatically be added to the exclusion list.)

**Select**

Use these buttons to move columns between the Include and Exclude list

KNIME Example Flow Server

Workflow Server: publicserver.knime.org

Status: connected as 'guest' [Disconnect]

001\_TimeSeries  
001001\_TimeSeries\_example  
002\_DataMining

KNIME Console

```
DEBUG ConnectionContainerEditPart modelling info: null
DEBUG WorkflowRootEditPart part: NodeContainerEditPart( Column Filter 0:2:6 (C
DEBUG ConnectionContainerEditPart refreshing visuals for: STD[0:2:5(1) -> 0:2
DEBUG ConnectionContainerEditPart modelling info: null
DEBUG OpenFileDialogAction Opening node dialog...
```





# Example Workflow (step by step)

The screenshot displays the KNIME software interface with a workflow titled "001001\_TimeSeries\_example" loaded. The workflow consists of five nodes connected in a sequence: "File Reader", "String to Date/Time", "Extract Time Window", "Column Filter", and "Line Plot". Below each node, a brief description is provided: "Read US dollar rates", "change string to date column", "only years 85 - 95", "only german rates", and "plot the rates". The "Run" button in the top toolbar is circled in red. The left sidebar shows the "Workflow Projects" and "Node Repository" panels. The bottom panel contains the "Console" and "Outline" views. The "Console" view shows debug messages related to the workflow execution, and the "Outline" view shows a small thumbnail of the workflow diagram.

**Workflow Projects:**

- 001001\_TimeSeries\_example
- 011001\_simpleWorkflowVariable
- 011004\_varsAndTimeSeries
- 050003\_ChurnAnalysis
- 050004\_lastfm\_Recommendations
- KNIME\_project

**Node Repository:**

- IO
- Database
- Data Manipulation
- Data Views
- Statistics
- Mining
- Distance Matrix
- Meta
- Flow Control
- Misc
- KNIME Labs
- Time Series
  - Date Field Extractor
  - Time Field Extractor
  - Extract Time Window
  - Mask Date/Time
  - Moving Average
  - Preset Date/Time
  - String to Date/Time
  - Time Difference
  - Time Generator
  - Time to String
- Reporting
- Weka
- XML

**Workflow Diagram:**

```
graph LR; A[File Reader] --> B[String to Date/Time]; B --> C[Extract Time Window]; C --> D[Column Filter]; D --> E[Line Plot];
```

**KNIME Console:**

```
DEBUG ConnectionContainerEditPart modelling info: null
DEBUG WorkflowRootEditPart part: NodeContainerEditPart( Line Plot 0:2:7 (CONF
DEBUG ConnectionContainerEditPart refreshing visuals for: STD[0:2:6(1) -> 0:2
DEBUG ConnectionContainerEditPart modelling info: null
DEBUG NodeContainerEditPart Line Plot 0:2:7 (CONFIGURED)
```

**KNIME Example Flow Server:**

Workflow Server: [publicserver.knime.org](http://publicserver.knime.org)

Status: connected as 'guest' [Disconnect](#)

**Workflow List:**

- 001\_TimeSeries
- 001001\_TimeSeries\_example
- 002\_DataMining



# Example Workflow (step by step)

The screenshot displays the KNIME software interface with a workflow titled "001001\_TimeSeries\_example" loaded. The workflow consists of five nodes: "File Reader" (Read US dollar rates), "String to Date/Time" (change string to date column), "Extract Time Window" (only years 85 - 95), "Column Filter" (only german rates), and "Line Plot" (plot the rates). The "Line Plot" node is currently selected, and its description is shown in the "Node Description" panel on the right. The description explains that the node plots numeric columns as lines, mapping all values to a single y coordinate, which may distort the visualization if the difference in values is large. It also notes that only columns with a valid domain are available in this view and that the predecessor node must be executed or the domain must be set with the "DomainCalculator" node. The "HiLiting:" section describes how data points can be selected by dragging a rectangle or clicking on points, and how they can be highlighted by right-clicking or using the "hilitte" menu. The "Tooltin:" section is also visible. Below the workflow, the "Console" panel shows the execution log, indicating that the workflow has been executed successfully. The "Outline" panel on the bottom right shows a hierarchical view of the workflow nodes.

**Line Plot**

Plots the numeric columns of the input table as lines. All values are mapped to a single y coordinate. This may distort the visualization if the difference of the values in the columns is large.

**Only columns with a valid domain are available in this view. Make sure that the predecessor node is executed or set the domain with the DomainCalculator node!**

**HiLiting:**

The data points may be selected by either dragging a rectangle with the mouse over the points or by clicking on points (if the dots are not displayed you won't see the selection or hilitte since only data points can be selected or hilitte). Hold control pressed for multiple selections. The selected datapoints may be hilitte by either right-click to get the context menu or via the hilitte menu in the menu bar.

**Tooltin:**

KNIME Example Flow Server

Workflow Server: [publicserver.knime.org](http://publicserver.knime.org)

Status: connected as 'guest' [Disconnect](#)

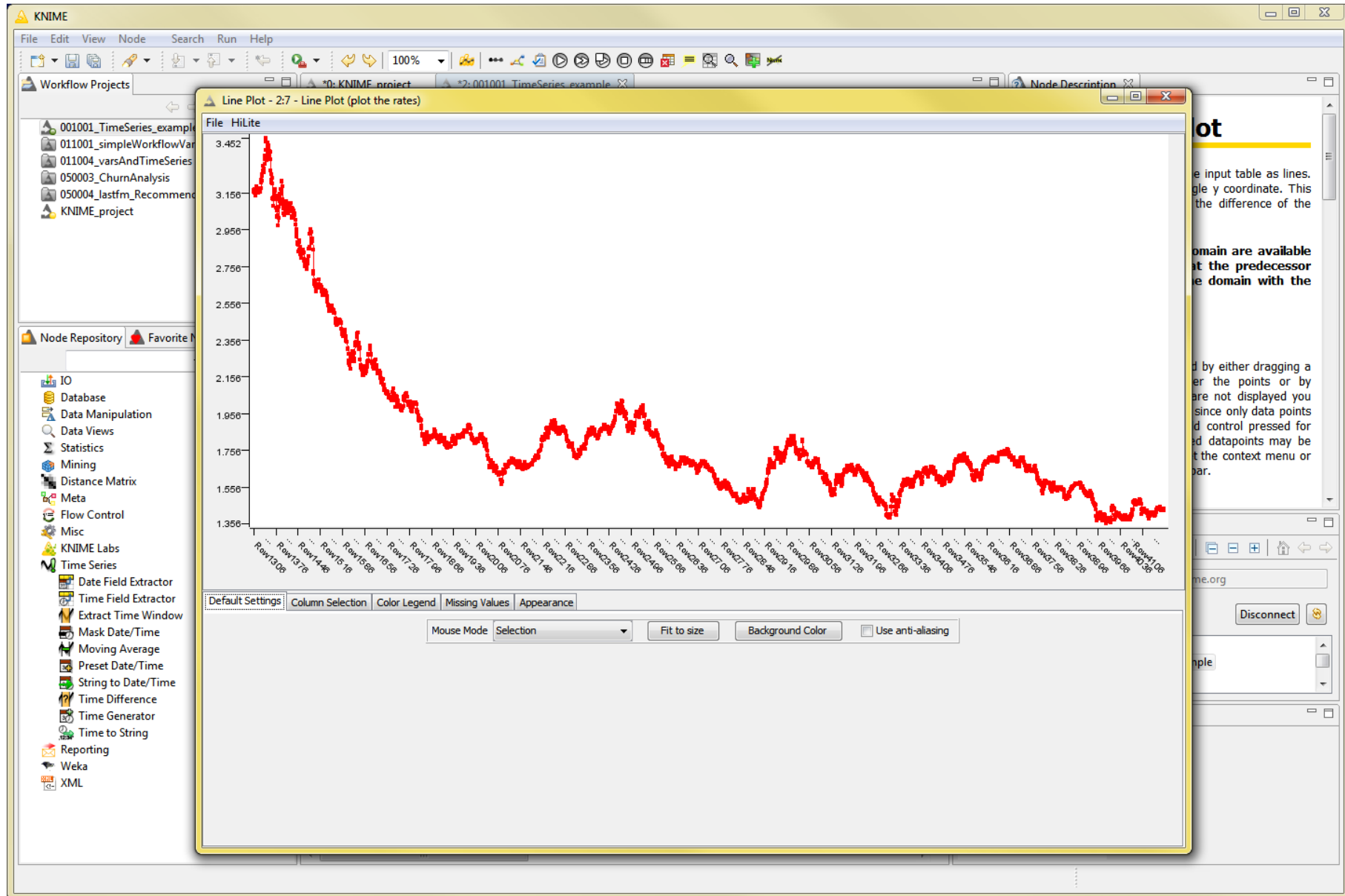
001\_TimeSeries  
001001\_TimeSeries\_example  
002\_DataMining

**KNIME Console**

```
DEBUG NodeContainer 001001_TimeSeries_example 0:2 has new state: EXECUTED
DEBUG NodeContainer 001001_TimeSeries_example 0:2 has new state: EXECUTED
DEBUG NodeContainer Workflow Manager 0 has new state: IDLE
DEBUG OpenViewAction Open Node View Line Plot (#0)
DEBUG DefaultVisualizationNodeModel Registering view at model (total count 1)
```



# Example Workflow (step by step)



ot

the input table as lines.  
gle y coordinate. This  
the difference of the

omain are available  
at the predecessor  
e domain with the

d by either dragging a  
er the points or by  
are not displayed you  
since only data points  
d control pressed for  
ed datapoints may be  
t the context menu or  
par.

me.org

Disconnect

ple



# Example Workflow (step by step)

The screenshot displays the KNIME software interface with a workflow titled "001001\_TimeSeries\_example" and a detailed view of the "Column Filter" node.

**Workflow Diagram:**

- File Reader:** Read US dollar rates
- String to Date/Time:** change string to date
- Extract Time Window:** only years 85 - 95
- Column Filter:** only german rates
- Line Plot:** plot the rates

A secondary "Column Filter" node is shown in a separate box, configured with "only swiss rates".

**Column Filter Node Description:**

### Column Filter

This node allows columns to be filtered from the input table while only the remaining columns are passed to the output table. Within the dialog, columns can be moved between the Include and Exclude list.

#### Dialog Options

**Include**  
This list contains the column names that are included in the output table.

**Enforce Inclusion**  
Select this option to enforce the current inclusion list to stay the same even if the input table specification changes. If some of the included columns are not available anymore, a warning is displayed. (New columns will automatically be added to the exclusion list.)

**KNIME Example Flow Server**  
Workflow Server: publicserver.knime.org  
Status: connected as 'guest' [Disconnect]

**KNIME Console**

```
DEBUG KnimeResourceNavigator ROOT's workflow has changed NODE_REMOVED
DEBUG KnimeResourceNavigator Workflow Date Field Extractor 0:2:3 removed
DEBUG WorkflowEditor Workflow event triggered: WorkflowEvent [type=NODE_REMOVED;
DEBUG WorkflowRootEditPart WorkflowRoot: workflow changed, refreshing children
DEBUG NodeContainerEditPart Column Filter 0:2:9 (CONFIGURED)
```

**Outline**



# Example Workflow (step by step)

The screenshot displays the KNIME software interface with a workflow titled "001001\_TimeSeries\_example". The workflow consists of the following nodes: File Reader (Read US dollar rates), String to Date/Time (change string to date column), Extract Time Window (only years 85 - 95), Column Filter (only german rates), and Line Plot (plot the rates). A second path branches from the String to Date/Time node, passing through a Column Filter (only swiss rates) and then a Date Field Extractor (Extract year and quarter) before joining the main flow.

The "Date Field Extractor" dialog is open, showing the following settings:

- Options: Flow Variables, Memory Policy
- Column to extract time fields from: OBS
- Year: ☒ Year, Column name: Year
- Quarter: ☒ Quarter, Column name: Quarter
- Month: ☐ Month, Value as: ☒ Text, Column name: Month
- Day of month: ☐ Day of month, Column name: Day of month
- Day of week: ☐ Day of week, Value as: ☒ Text, Column name: Day of week
- Day of year: ☐ Day of year, Column name: Day of year

The right sidebar shows the "Node Description" for the "Date Field Extractor" node, which explains its function: "Extracts the selected date fields from a date/time column and appends the values as integer columns. Available fields are year, quarter, month, day, day of week. If these values are not set in the date/time cell a missing value is added in the result table."

The bottom console shows the following log messages:

```
KNIME Console
DEBUG NodeContainerEditPart Column
DEBUG NodeContainerEditPart Date Fi
DEBUG WorkflowEditorTemplateDragSource
DEBUG WorkflowRootEditPart Workflo
DEBUG OpenFileDialogAction Opening
```



# Example Workflow (step by step)

The image displays the KNIME software interface with a workflow and a dialog box for configuring the GroupBy node.

**Workflow:**

- File Reader:** Read US dollar rates
- String to Date/Time:** change string to date column
- Extract Time Window:** only years 85 - 95
- Column Filter:** only german rates
- Line Plot:** plot the rates
- Column Filter:** only swiss rates
- Date Field Extractor:** Extract year and quarter
- GroupBy:** aggregate over quaters

**Dialog - 2:4 - GroupBy (aggregate over quaters):**

- Groups:** Available column(s) (red box), Column(s):, Search, Select all search hits, Select, add >>, add all >>, << remove, << remove all.
- Options:** Group column(s) (green box), Column(s):, Search, Select all search hits, Year, Quarter.

**Node Description: GroupBy**

Groups the rows of a table by the unique values in the selected columns. A row is created for each unique value group of the selected column(s). The remaining rows are aggregated by the defined method. The output table therefore contains one row for each existing value combination of the selected group column(s).

To change the aggregation method of more than one column select all columns to change, open the context



# Example Workflow (step by step)

The screenshot displays the KNIME software interface with a workflow and a dialog box for the 'GroupBy' node.

**Workflow:**

- File Reader:** Read US dollar rates
- String to Date/Time:** change string to date column
- Extract Time Window:** only years 85 - 95
- Column Filter:** only german rates
- Line Plot:** plot the rates
- Column Filter:** only swiss rates
- Date Field Extractor:** Extract year and quarter
- GroupBy:** aggregate over quaters

**Dialog - 2:4 - GroupBy (aggregate over quaters):**

Settings | Description | Flow Variables | Memory Policy

Groups | Options

Aggregation settings

Available columns: OBS, SWISFUS

Select:

- add >>
- add all >>
- << remove
- << remove all

To change multiple columns use right mouse click for context menu.

Column	Aggregation (click to change)	missing
SWISFUS	Mean	<input type="checkbox"/>
OBS	First	<input type="checkbox"/>

Advanced settings

Maximum unique values per group: 10,000

Column naming: Aggregation method (column name)

☐ Enable hilling ☐ Value delimiter: , ☐ Process in memory ☐ Sort in memory ☐ Retain row order

OK Apply Cancel





Group table - 2:4 - GroupBy (aggregate over quaters)

File

Table "default" - Rows: 277 Spec - Columns: 4 Properties Flow Variables

Row ID	Year	Quarter	D Mean(SWISFUS)	First(OBS)
Row23	1985	3	2.346	01.Jul.1985
Row24	1985	4	2.135	01.Oct.1985
Row25	1986	1	1.979	01.Jan.1986
Row26	1986	2	1.866	01.Apr.1986
Row27	1986	3	1.688	01.Jul.1986
Row28	1986	4	1.663	01.Oct.1986
Row29	1987	1	1.547	01.Jan.1987
Row30	1987	2	1.492	01.Apr.1987
Row31	1987	3	1.526	01.Jul.1987
Row32	1987	4	1.402	01.Oct.1987
Row33	1988	1	1.376	01.Jan.1988
Row34	1988	2	1.419	01.Apr.1988
Row35	1988	3	1.566	01.Jul.1988
Row36	1988	4	1.495	03.Oct.1988
Row37	1989	1	1.584	02.Jan.1989
Row38	1989	2	1.696	03.Apr.1989
Row39	1989	3	1.658	03.Jul.1989
Row40	1989	4	1.607	02.Oct.1989
Row41	1990	1	1.507	01.Jan.1990
Row42	1990	2	1.443	02.Apr.1990
Row43	1990	3	1.336	02.Jul.1990
Row44	1990	4	1.274	01.Oct.1990
Row45	1991	1	1.312	01.Jan.1991
Row46	1991	2	1.474	01.Apr.1991
Row47	1991	3	1.517	01.Jul.1991
Row48	1991	4	1.433	01.Oct.1991
Row49	1992	1	1.457	01.Jan.1992





# Example Workflow (step by step)

The screenshot displays the KNIME software interface with a workflow and a dialog box for the Moving Average node.

**Workflow:**

- File Reader:** Read US dollar rates
- String to Date/Time:** change string to date column
- Extract Time Window:** only years 85 - 95
- Column Filter:** only german rates
- Line Plot:** plot the rates
- Column Filter:** only swiss rates
- Date Field Extractor:** Extract year and quarter
- GroupBy:** aggregate over quarters
- Moving Average:** simple MA

**Dialog - 2:8 - Moving Average (simple MA):**

- Type of Moving Average:** Center simple
- Window Length:** 21
- ☐ Remove original columns
- Exclude:**
  - Column(s):
  - ☐ Select all search hits
  - Year
  - Quarter
- Select:**
  - add >>
  - add all >>
  - << remove
  - << remove all
- Include:**
  - Column(s):
  - ☐ Select all search hits
  - Mean(SWISFUS)

**Node Description: Moving Average**

This node calculates the moving average of a column. The moving average values are displayed in a new column appended at the end of the table or (if selected) replaces the original columns..

**Dialog Options**

Options

KNIME Console

```
DEBUG KnimeResourceNavigator state changed to
DEBUG NodeContainer 001001_TimeSeries_example
DEBUG NodeContainer 001001_TimeSeries_example
DEBUG NodeContainer Workflow Manager 0 has ne
DEBUG OpenFileDialogAction Opening node dial
```



# Example Workflow (step by step)

The screenshot displays the KNIME software interface with a workflow and a dialog box open.

**Workflow:**

- File Reader:** Read US dollar rates
- String to Date/Time:** change string to date column
- Extract Time Window:** only years 85 - 95
- Column Filter:** only german rates
- Line Plot:** plot the rates
- Column Filter:** only swiss rates
- Date Field Extractor:** Extract year and quarter
- GroupBy:** aggregate over quaters
- Moving Average:** simple MA
- Math Formula:** diff betw. mean and MA
- Line Plot:** (Warning icon)

**Dialog - 2:15 - Math Formula (diff betw. mean and MA):**

**Math Expression** | Flow Variables | Memory Policy

**Column List:**

- Year
- Quarter
- Mean(SWISFUS)
- MA(Mean(SWISFUS))

**Mathematical Function:**

- ln(x)
- log(x)
- exp(x)
- abs(x)
- sqrt(x)
- rand()
- mod(x, y)
- if(x, y, z)
- round(x)
- round(x, y)
- ceil(x)

**Constants:**

- ROWCOUNT
- ROWINDEX
- pi
- e
- COL\_MIN(col\_name)
- COL\_MAX(col\_name)
- COL\_MEAN(col\_name)
- COL\_STDDEV(col\_name)
- COL\_VAR(col\_name)

**Expression:**

abs(\$MA(Mean(SWISFUS))-\$Mean(SWISFUS))

**Append Column:** trend

**Replace Column:** MA(Mean(SWISFUS))

**Buttons:** OK, Apply, Cancel

**Node Description:**

## Math Formula

This node evaluates a (free-form) mathematical expression based on the values in a row. The computed results can be either appended as new column or be used to replace an input column. Available variables are the values in the corresponding row of the table (left list in the dialog). Commonly used functions are shown in the list "Mathematical Functions". There are also some constants available, such as  $\pi$  (the ratio of the circumference of a circle to its diameter),  $e$  (the base of the natural logarithms), the total number of rows in the table, and some other column based constants (right list).

This node uses JEP, the [Java Math Expression Parser](#).

## Dialog Options

**KNIME Example Flow Server**

Workflow Server: publicserver.knime.org

Status: connected as 'guest' **Disconnect**

**001\_TimeSeries**

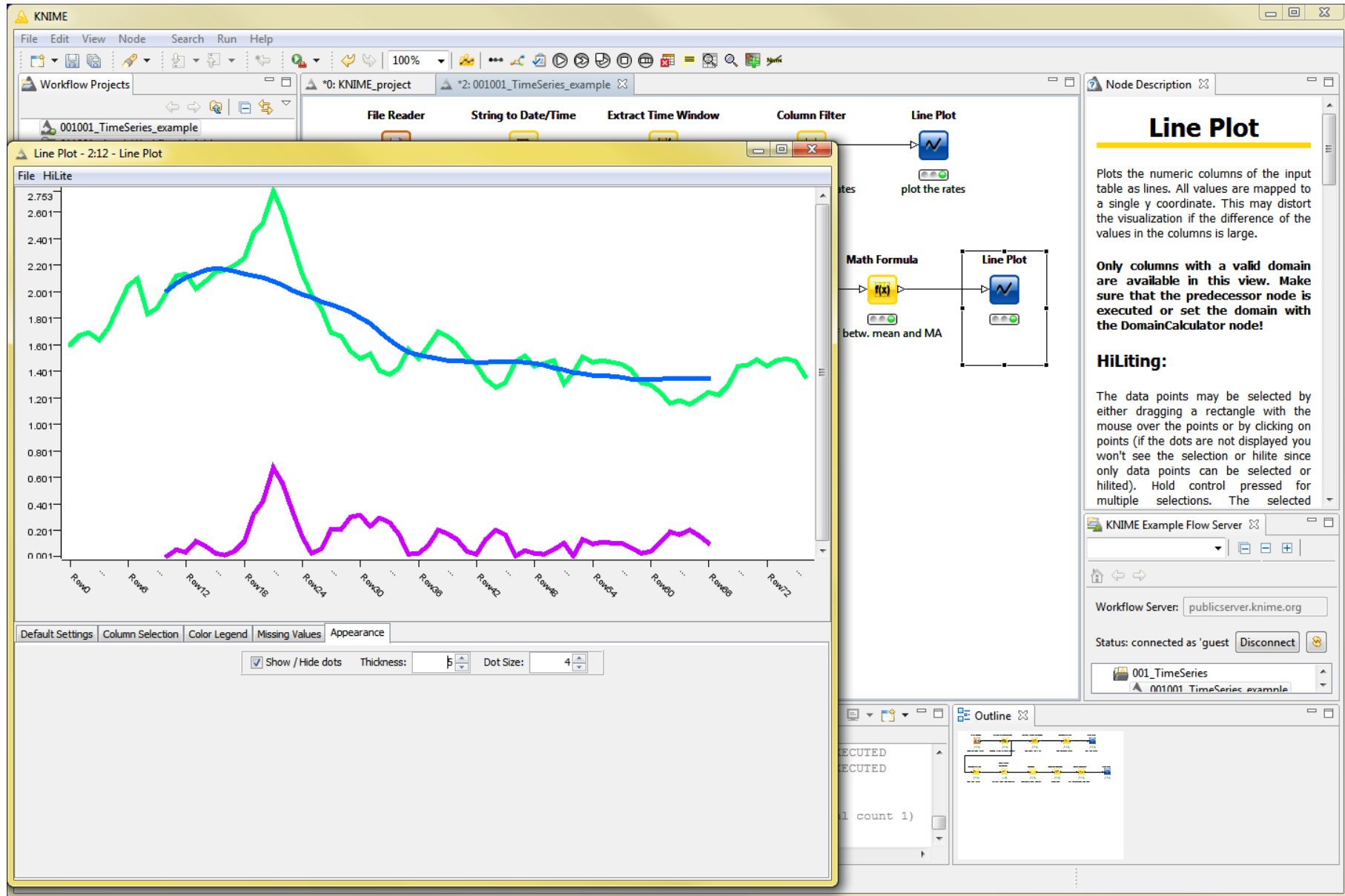
**001001\_TimeSeries\_example**

**Outline**

```
reshing visuals for: STD[0:2:15(1) -> 0:
lling info: null
reshing visuals for: STD[0:2:15(1) -> 0:
lling info: null
dialog...
```



# Example Workflow (step by step)





# Example Workflow (step by step)

The screenshot displays the KNIME software interface with a workflow and a dialog box for the 'Moving Average' node.

**Workflow:**

- File Reader:** Read US dollar rates
- String to Date/Time:** change string to date column
- Extract Time Window:** only years 85 - 95
- Column Filter:** only german rates
- Line Plot:** plot the rates
- Column Filter:** only swiss rates
- Date Field Extractor:** Extract year and quarter
- GroupBy:** aggregate over quarters
- Moving Average:** double exponential
- Math Formula:** diff betw. mean and MA
- Line Plot:** plot the rates

**Dialog - 2:8 - Moving Average (double exponential):**

- Options:** Double exponential (selected)
- Window Length:** 21
- Remove original columns:** ☐
- Exclude:** Year, Quarter
- Select:** add >>, add all >>, << remove, << remove all
- Include:** Mean(SWISFUS)

**Node Description: Moving Average**

This node calculates the moving average of a column. The moving average values are displayed in a new column appended at the end of the table or (if selected) replaces the original columns..

**Dialog Options**

**Options**

**Columns containing Double Values**

Select the input column containing double values on which to perform the moving average.

**Window Length**

The number of samples to include in the moving average window. It has to be an odd number if a

**KNIME Example Flow Server**

Workflow Server: publicserver.knime.org

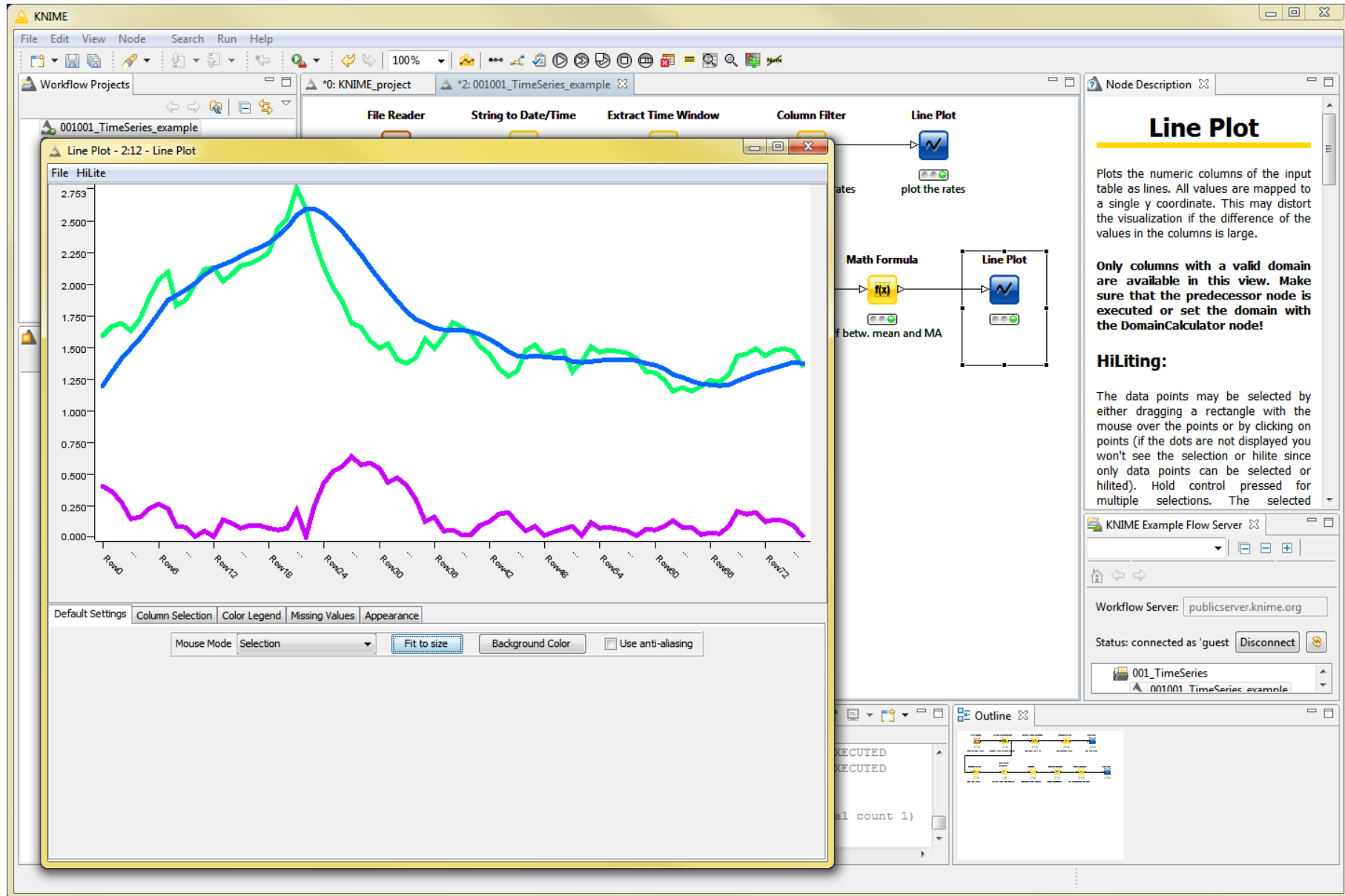
Status: connected as 'guest' Disconnect

**Outline**

FIGURED  
has new state: CONFIGURED  
has new state: CONFIGURED  
ate: IDLE  
ng node dialog after double cl



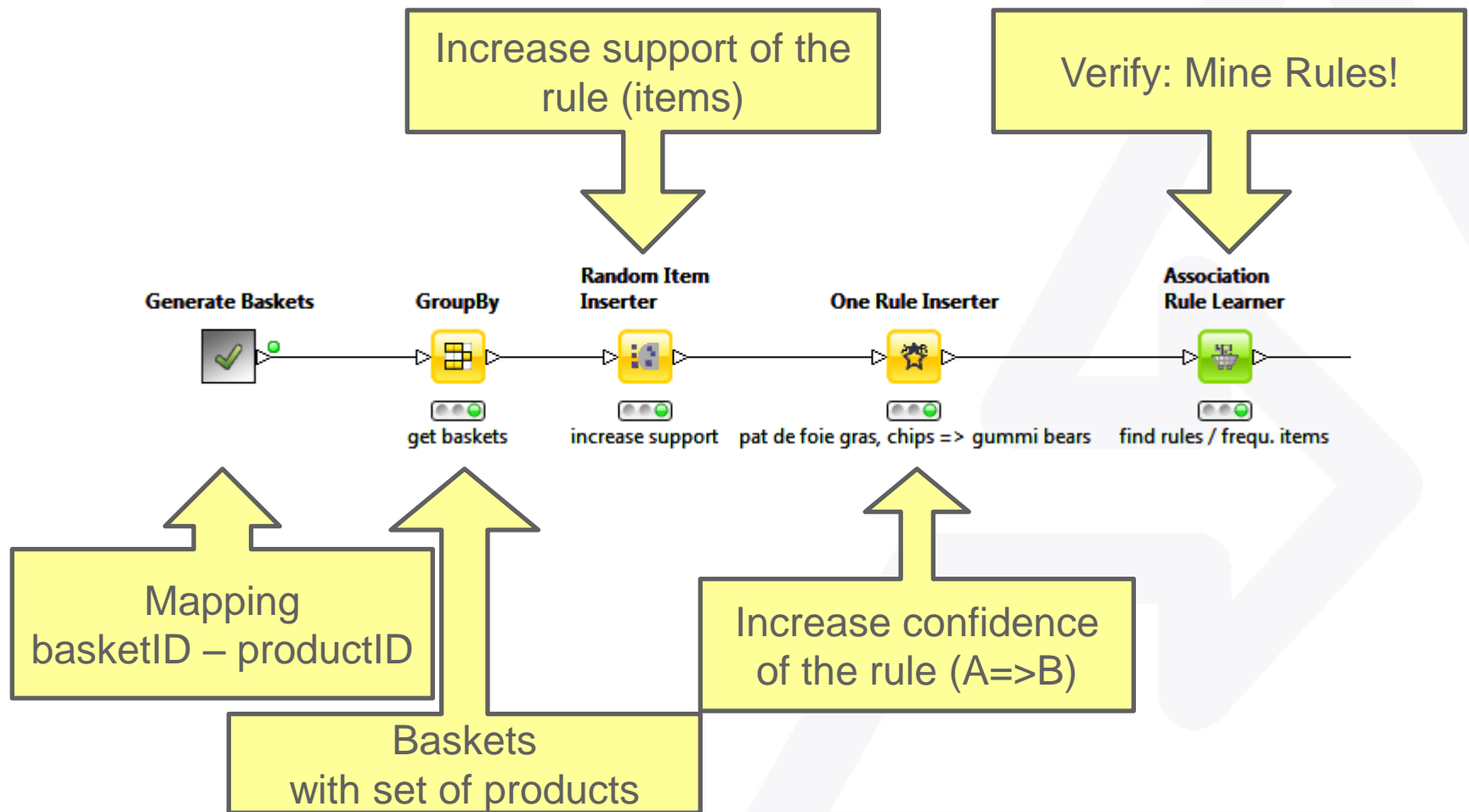
# Example Workflow (step by step)





# Association Rules

- Using the Modular Data Generation from KNIME Labs





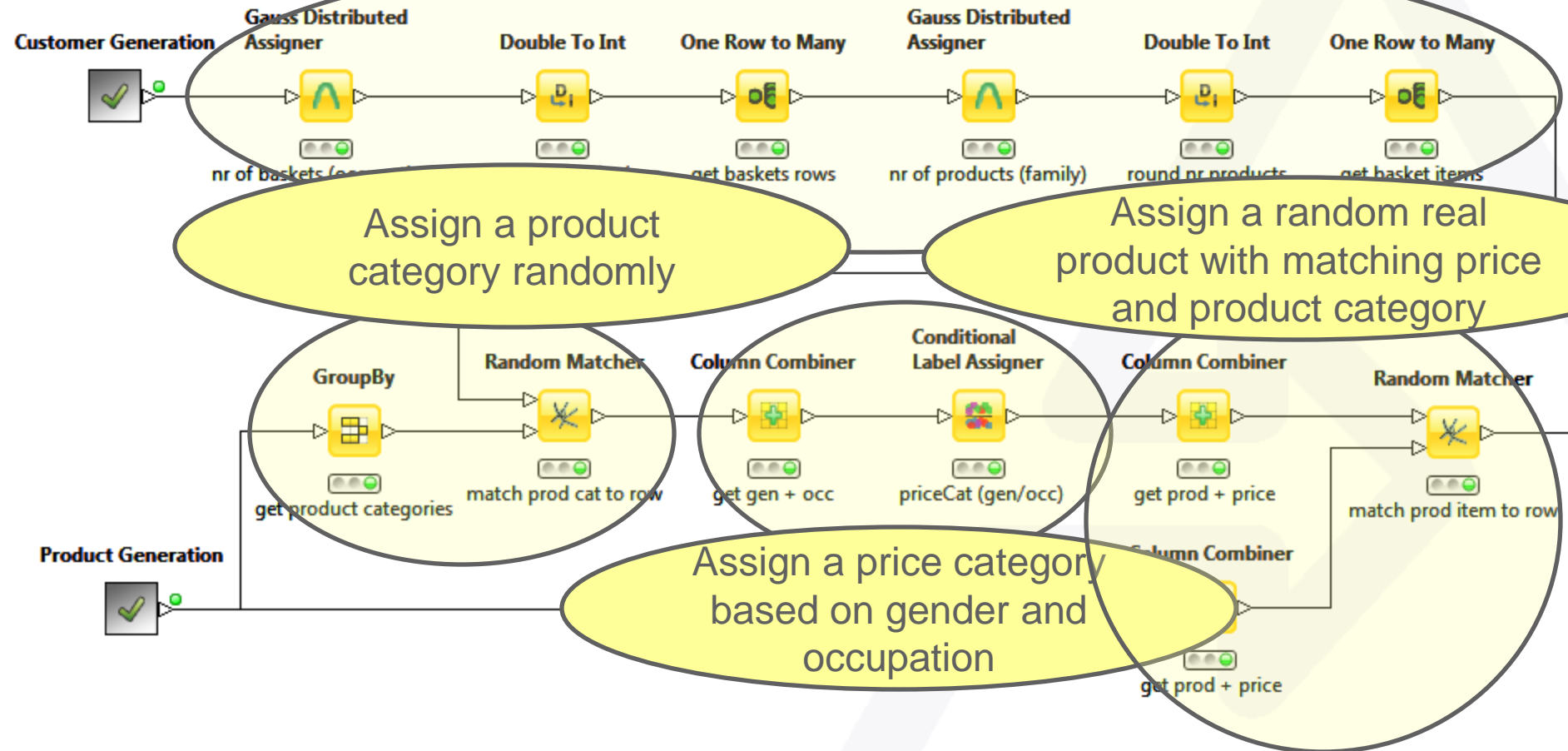
# Generate Customer Data

Generate one row per bought item

Assign a product category randomly

Assign a random real product with matching price and product category

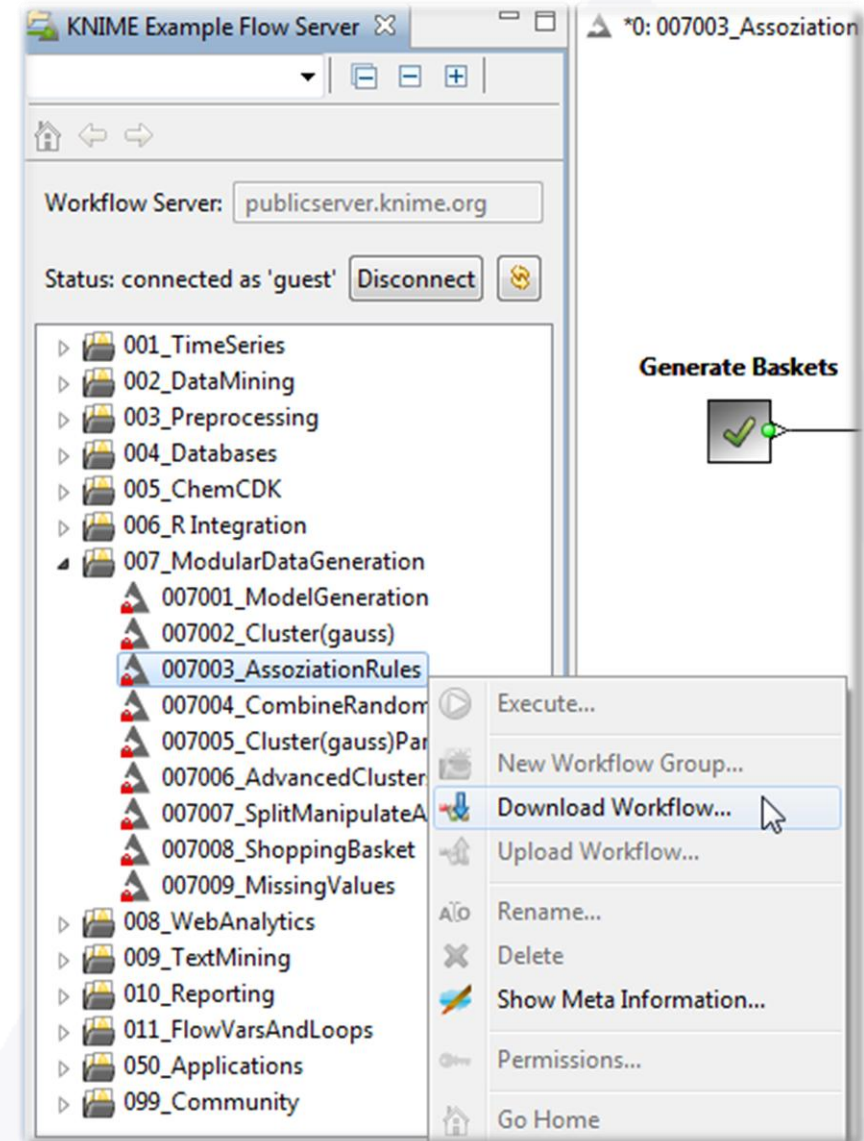
Assign a price category based on gender and occupation





# Download Example Workflows

- KNIME Example Flow Server
- Available via the view menu in KNIME
- Public workflow repository
- Demonstrated workflows:
  - 001001\_TimeSeries\_example
  - 007003\_AssoziationRules
  - 007008\_ShoppingBasket



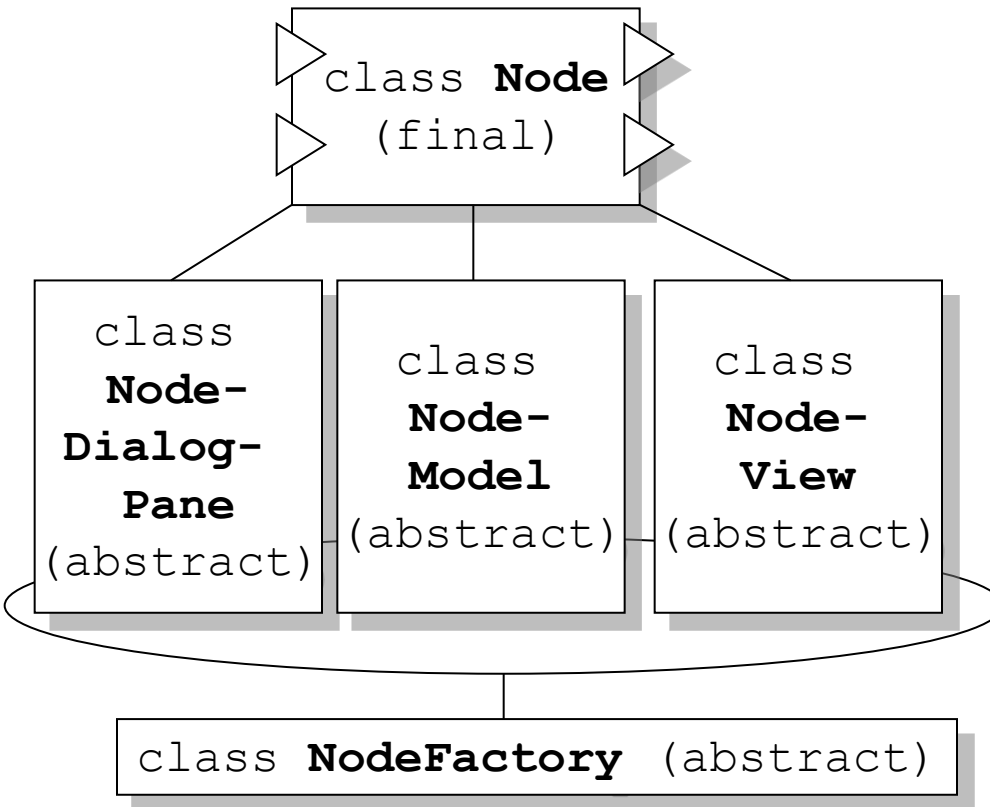




# DEVELOPMENT



# Node Architecture



- KNIME interacts only with a **Node**
- **Node** takes care of embedding the node in the infrastructure
- New nodes implement Model/View/Dialog



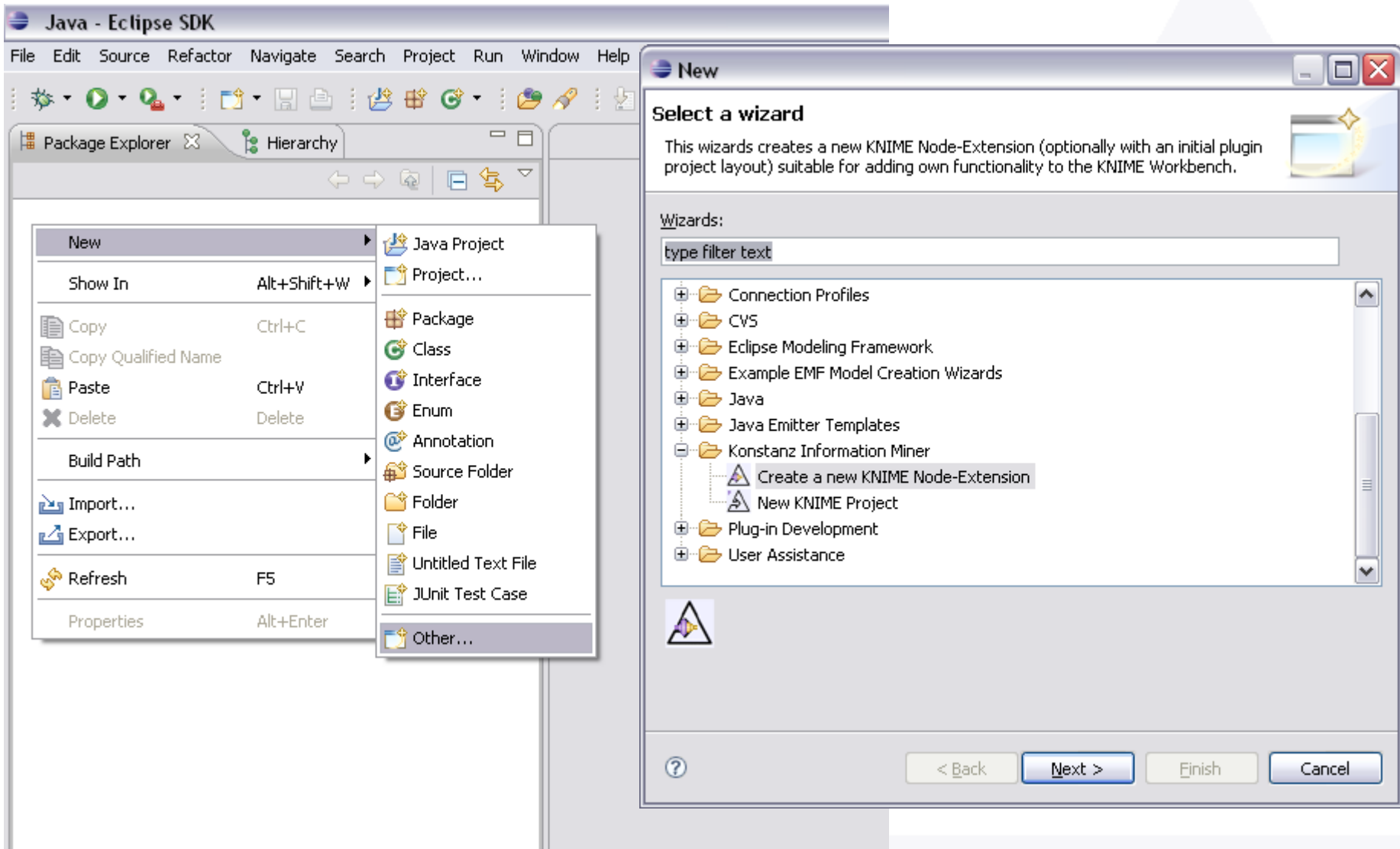
# Node Extension Wizard

- Included in the KNIME Developer Version
- Allows creation of plugin projects including functioning KNIME nodes (with sample code)
- Helpful to easily create all node classes
  - Generates all Java classes
  - Node is registered with the plugin project
  - Launch KNIME and enjoy the new node working!





# Node Extension Wizard





# Node Extension Wizard

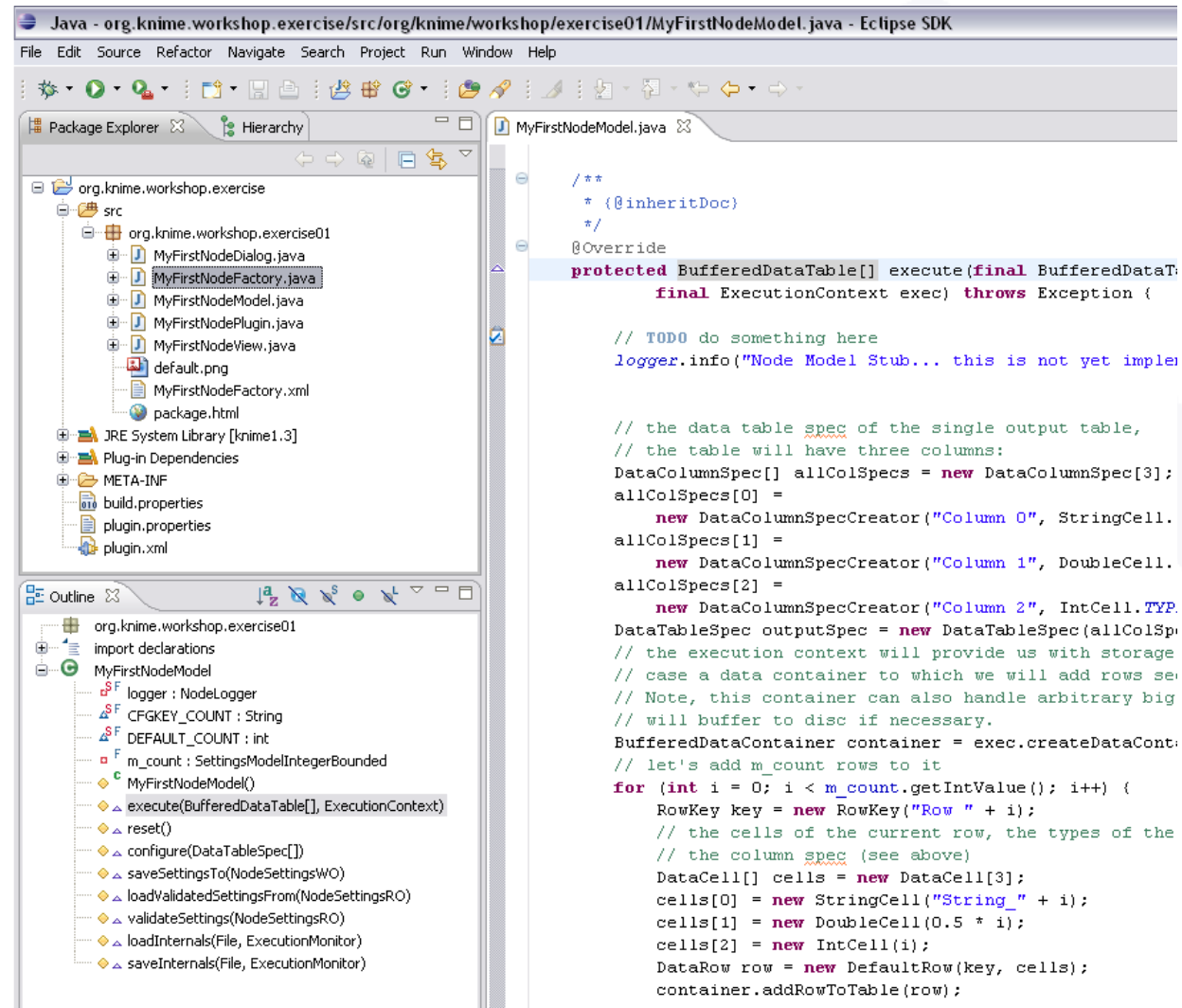
- Specify all settings to create a new KNIME node
  - In a completely new plugin project, or
  - Into an existing project
- Node type: Sink, Source, Learner, Predictor, Manipulator, Visualizer, Meta, or Other
- Include sample code or not

The screenshot shows the 'Create new KNIME Node-Extension' wizard. The title bar says 'Create new KNIME Node-Extension'. Below the title bar, it says 'This wizard creates a KNIME Node-Extension (optionally with an initial plugin project)'. The wizard has two radio buttons: 'New Project Name:' (selected) and 'Existing Project:'. The 'New Project Name' field contains 'org.knime.workshop.exercise'. Below this is a section titled 'KNIME extension settings'. It contains several fields: 'Node class name (e.g. 'MyLearner'):' with 'MyFirst', 'Package name (e.g. 'org.myname'):' with 'org.knime.workshop.exercise01' and a 'Browse' button, 'Node vendor (e.g. Your Name):' with 'KNIME GmbH', and 'Node description text:' with an empty text area. Below these is a 'Node type:' dropdown menu with 'Other' selected. To the right of the dropdown is a list of other node types: Predictor, Manipulator, Visualizer, Meta, and Other. At the bottom left of the settings section is a checkbox 'Include sample code in generated' which is checked. At the bottom of the dialog are four buttons: '?', '< Back', 'Next >', 'Finish', and 'Cancel'.



# Node Extension Wizard

- Contains all Java classes (including sample code)
- Node is registered in the **plugin.xml**
- **NodeDialog** and **NodeView** class are also created and registered to the **NodeFactory**





# Summary

- Nodes encapsulate operations
- Nodes provide intermediate results
- Self explaining workflows
- Handles large data sets
- Easy extensible

