Workflow Mining:

Identification of frequent patterns in a large collection of KNIME workflows

Nils Weskamp, Research Scientist
Computational Chemistry
nils.weskamp@boehringer-ingelheim.com

Boehringer Ingelheim

Overview

Boehringer Ingelheim

- Motivation
  - History & current status
  - Why “Workflow Mining”?
- “Static Workflow Mining”
  - Extraction of workflow graphs
  - Frequent sub-graph mining
  - Results
- “Dynamic Workflow Mining”
  - KNIME Profiler Plugin
  - Analysis of profiling logs
- Conclusions and Outlook

Overview

Boehringer Ingelheim

- **Motivation**
  - History & current status
  - Why “Workflow Mining”?
- “Static Workflow Mining”
  - Extraction of workflow graphs
  - Frequent sub-graph mining
  - Results
- “Dynamic Workflow Mining”
  - KNIME Profiler Plugin
  - Analysis of profiling logs
- Conclusions and Outlook

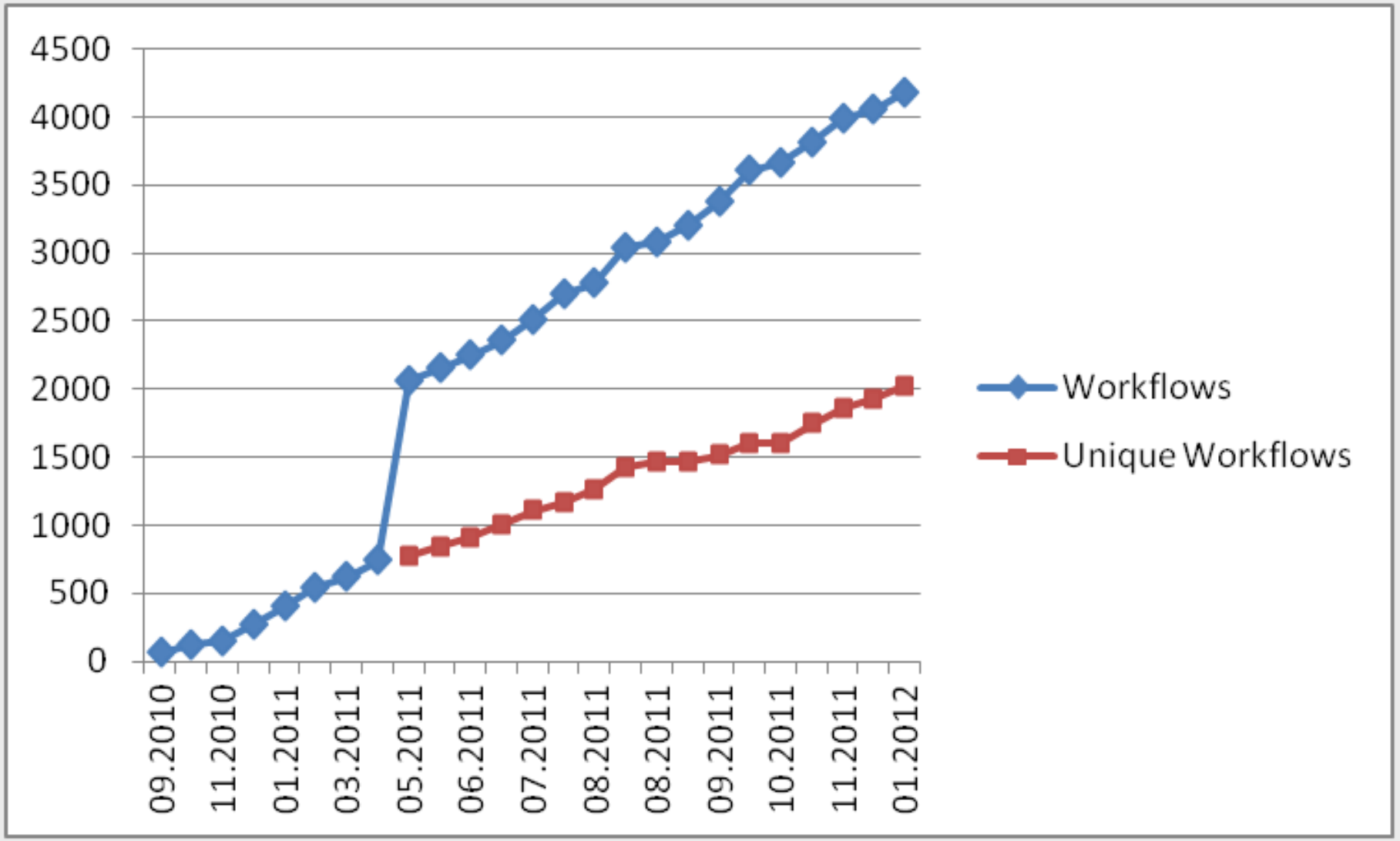
History

Boehringer Ingelheim

- 2009
  - First look at KNIME, decision to perform in-depth evaluation
  - Identification of critical functional gaps
  - Inhouse developer training
  - Start of collaboration with KNIME.com
- 2010
  - Implementation of missing nodes, integration of various external chemistry tools
  - Licensing of additional node collections from commercial vendors
  - Migration of deployed workflows (e.g. QSAR-models for ADME parameters)
- 2011
  - Migration of end-users from computational chemistry
  - Inhouse end-user trainings
  - Extension of user base to other groups (e.g. computational biology, IS, HTS)

Development of KNIME usage

Boehringer Ingelheim



Current status

Boehringer Ingelheim

- A significant group of end-users is applying KNIME on a regular / daily basis
  - Consistently more than 100 new, unique workflows are created per month
  - Significant exchange and sharing of workflows among users

- A lot of feedback from end-users
  - Bug reports
  - Feature requests
  - How do I …
  - …

- Collecting feedback is simple, but …
  - … how to prioritize feature requests ?
  - … what are best practices / common pitfalls / bottlenecks ?

Use data mining to support decisions!

Boehringer Ingelheim

- Difficult to assess significance of requests in a growing and diverse user community
  - Many different application scenarios
  - How are our users actually using KNIME?
  - Different people solve the same problem in a different way!
  - Which nodes are really important?

- Identify key users to cover diversity in user community
  - Still, a lot of gut feeling involved

- Analyze existing workflow collection to get reliable usage statistics

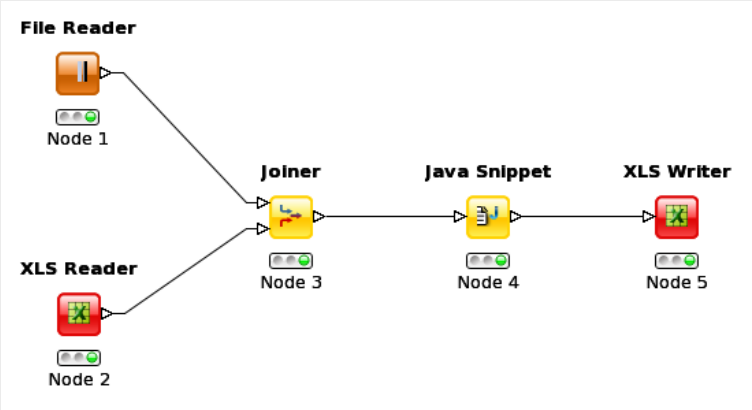
Overview

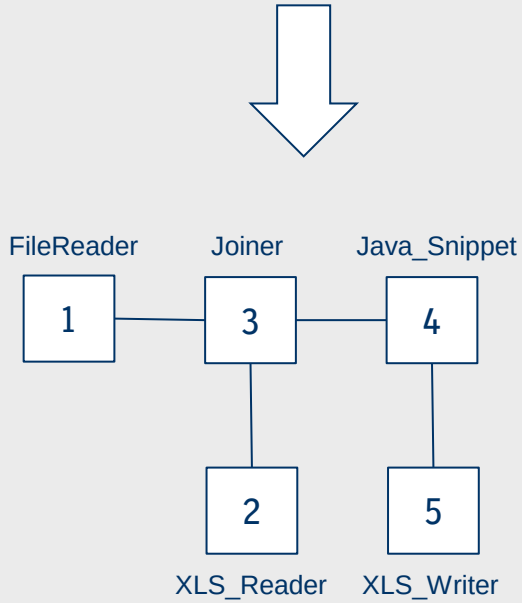
Boehringer Ingelheim

- Motivation
  - History & current status
  - Why “Workflow Mining”?
- **“Static Workflow Mining”**
  - Extraction of workflow graphs
  - Frequent sub-graph mining
  - Results
- “Dynamic Workflow Mining”
  - KNIME Profiler Plugin
  - Analysis of profiling logs
- Conclusions and Outlook

Extraction of useful descriptors from KNIME workflows

Boehringer Ingelheim





- KNIME workflows are stored in a generic XML format
- A simple parser extracts the **workflow graph**:
  - Nodes
  - Connections
- The analysis discards:
  - Node configuration / labels
  - Connection type / direction
  - Workflow layout
  - Workflow annotations
  - Personal information
  - …

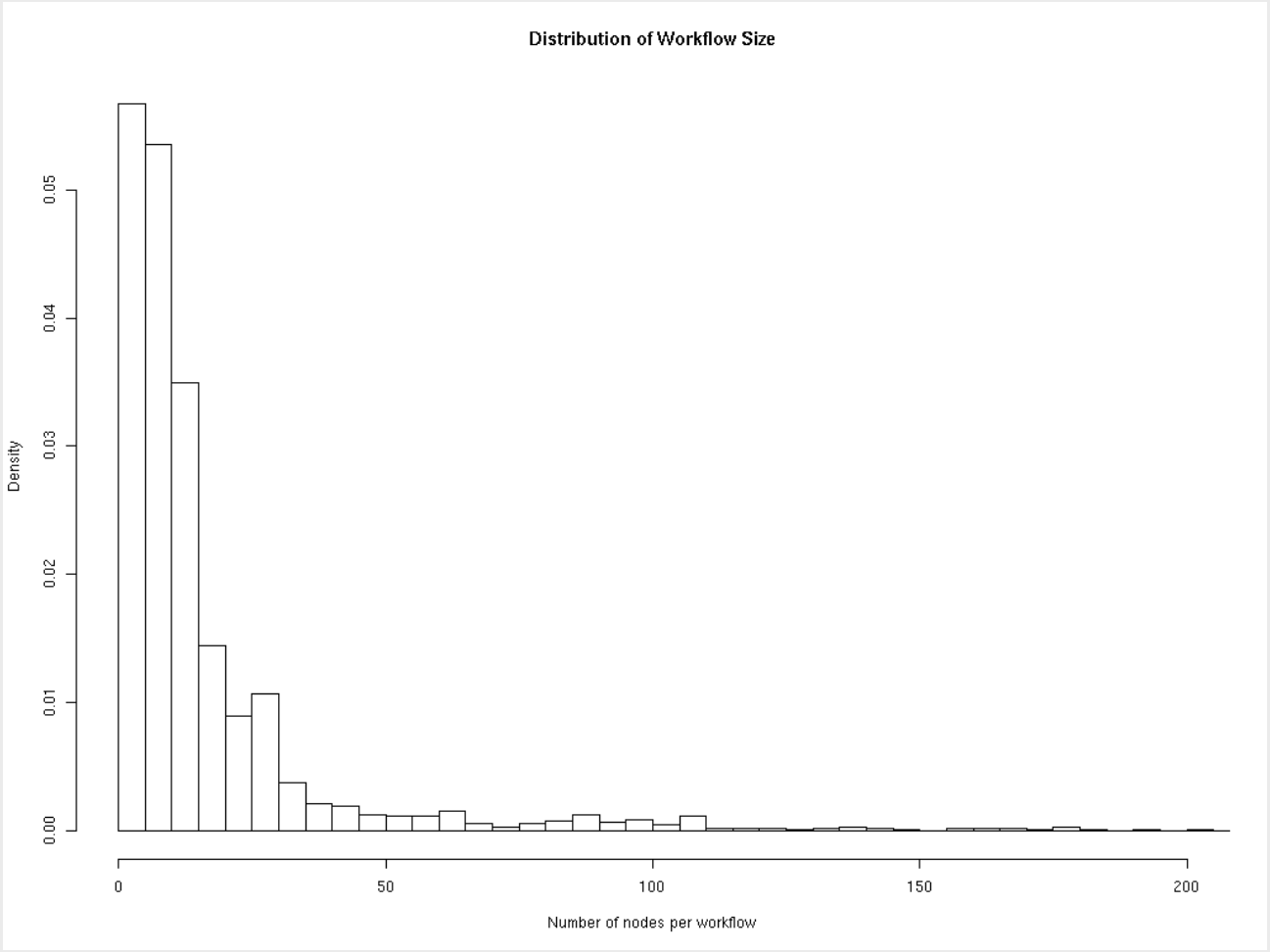
Frequent sub-graph mining

Boehringer Ingelheim

- What is frequent sub-graph mining?
  - Problem: Given a database of (labeled) graphs **G** = $\{G_1, ..., G_n\}$ return all (connected) sub-graphs **S** = $\{S_1, ..., S_k\}$ occurring at least **x** times in **G**.
  - Related to frequent item set mining & association rule mining
  - Many applications in the life-sciences
    - Chemical structures
    - Biological networks
    - …
  - Different algorithms and implementations available
    - MoSS node in KNIME for chemical data
    - …

Dataset

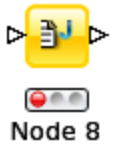
- Using all workflows from central installation (~4.000 as of 2012-01-10)
- 3.899 workflows were parseable and non-empty
- 64.357 nodes
- 62.841 connections



Results – Top 10 nodes

Boehringer Ingelheim

Java Snippet



9907

GroupBy

5275

Joiner

4534

Rename

4290

Row Filter

4144

Column Filter

3326

File Reader

1720

DB Open v1

1707

Sorter

1628

Java Edit Variable

1570

Results – Top 11-20 nodes

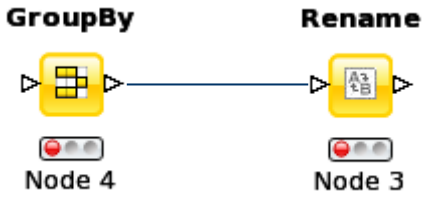
Boehringer Ingelheim

Table Reader  Node 4	1567
Concatenate  Node 9	1267
Table Writer  Node 5	1212
Metanode  Node 5	1136
DB PreparedQuery v1  Node 2	1051
Reference Row Filter  Node 1	893
Double To Int  Node 17	882
DB MethodsView v1  Node 8	790
JPython Script 1:1  Node 3	683
DB Query v1  Node 6	660

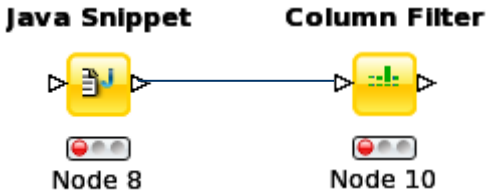
Results – Workflow patterns with 10% support (selection)

Boehringer Ingelheim

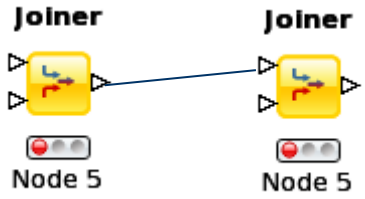
- Allow addition of multiple columns in Java Snippet ?
- Collaboration with KNIME.com



- Allow for more flexibility in column naming / typing ?



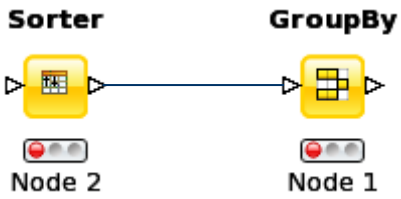
- Make users aware of “Replace columns” option

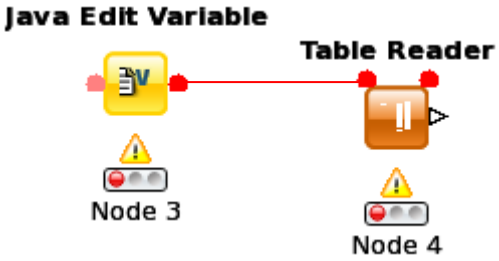


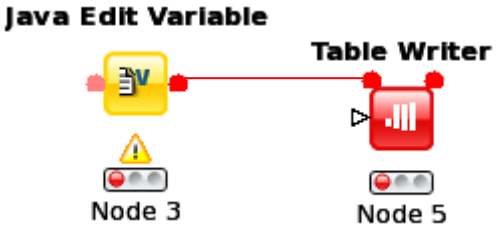
- Allow joining multiple input tables? (e.g. “Joiner (optional in)”)

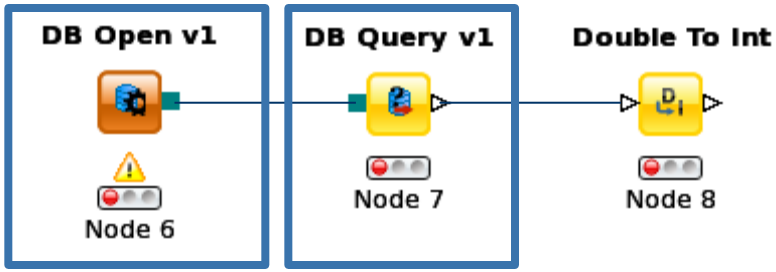
Results – Workflow patterns with 7.5% support (selection)

Boehringer Ingelheim









- Issues with sorting stability until v2.3.4 (Bug 2682)
- Develop a concept for simple and efficient data transfer among related workflows?
- Improve column typing in inhouse database integration

Overview

Boehringer Ingelheim

- Motivation
  - History & current status
  - Why “Workflow Mining”?
- “Static Workflow Mining”
  - Extraction of workflow graphs
  - Frequent sub-graph mining
  - Results
- **“Dynamic Workflow Mining”**
  - KNIME Profiler Plugin
  - Analysis of profiling logs
- Conclusions and Outlook

Static vs. dynamic workflow analysis

Boehringer Ingelheim

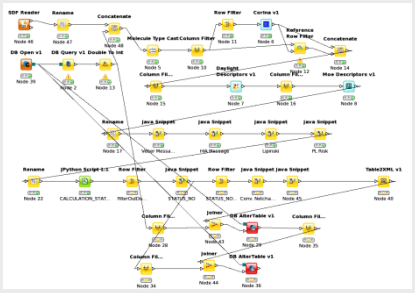
- Collection consists mainly of ad hoc-workflows
  - On average, only few executions per workflow
  - Non-interactive batch processes responsible for most workflow executions (20.000.000+ property predictions over the last 12 months)
  - Static analysis of all stored workflows might yield biased results

- No insights into bottlenecks and performance issues
  - Influence of different implementations for the same node
  - Improvements of the KNIME framework

- Limited ability to capture influence of “control flow” constructs
  - Include or exclude inactive branches in analysis?
  - How to handle loops?

KNIME Execution Profiler Plugin
© KNIME.com

Boehringer Ingelheim

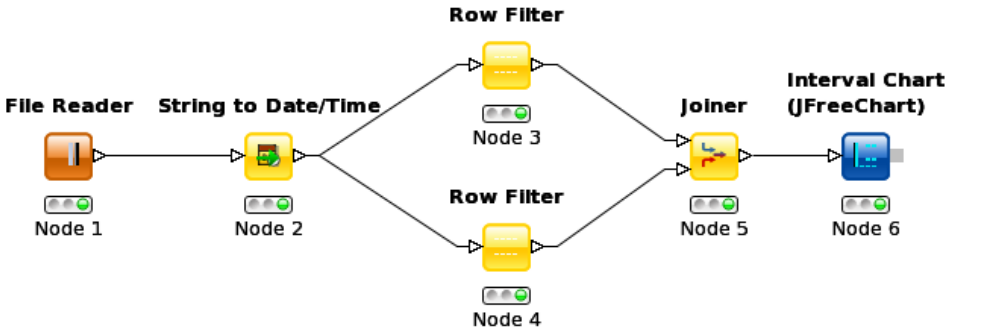


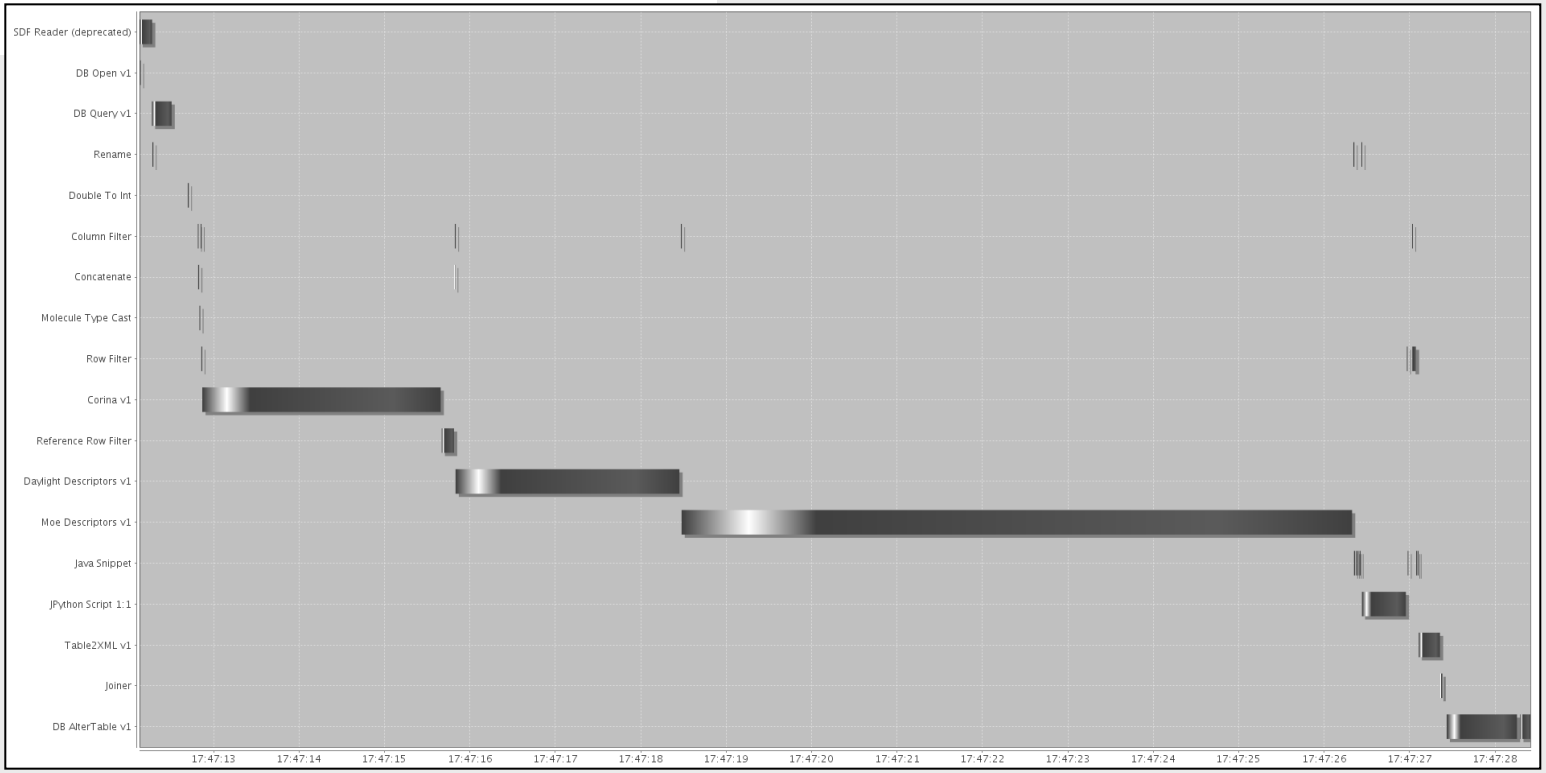
Timestamp	Workflow Path	Node Name	Node Index	Custom Name	State
2011-11-12;06:45:54.662		<Metanode>	0	Node 0	EXECUTED
2011-11-12;06:45:54.670		<Metanode>	0	Node 0	CONFIGURED
2011-11-12;06:45:55.237	PROP4	DB Open v1	39	Node 39	IDLE
2011-11-12;06:45:55.240	PROP4	DB Open v1	39	Node 39	CONFIGURED
2011-11-12;06:45:55.262	PROP4	SDF Reader (deprecated)	46	Node 46	IDLE
2011-11-12;06:45:55.317	PROP4	SDF Reader (deprecated)	46	Node 46	CONFIGURED
2011-11-12;06:45:55.372	PROP4	DB Query v1	2	Node 2	IDLE
2011-11-12;06:45:55.706	PROP4	DB Query v1	2	Node 2	CONFIGURED
2011-11-12;06:45:55.751	PROP4	Rename	47	Node 47	IDLE
2011-11-12;06:45:55.757	PROP4	Rename	47	Node 47	CONFIGURED
2011-11-12;06:45:55.761	PROP4	Double To Int	13	Node 13	IDLE
2011-11-12;06:45:55.764	PROP4	Double To Int	13	Node 13	CONFIGURED
2011-11-12;06:45:55.769	PROP4	Concatenate	48	Node 48	IDLE
2011-11-12;06:45:55.774	PROP4	Concatenate	48	Node 48	CONFIGURED
2011-11-12;06:45:55.778	PROP4	Column Filter	28	Node 28	IDLE
2011-11-12;06:45:55.779	PROP4	Column Filter	28	Node 28	CONFIGURED
2011-11-12;06:45:55.790	PROP4	Molecule Type Cast	5	Node 5	IDLE
2011-11-12;06:45:55.793	PROP4	Molecule Type Cast	5	Node 5	CONFIGURED
2011-11-12;06:45:55.797	PROP4	Column Filter	10	Node 10	IDLE
2011-11-12;06:45:55.798	PROP4	Column Filter	10	Node 10	CONFIGURED
2011-11-12;06:45:55.807	PROP4	Row Filter	11	Node 11	IDLE
2011-11-12;06:45:55.807	PROP4	Row Filter	11	Node 11	CONFIGURED
2011-11-12;06:45:55.868	PROP4	Corina v1	6	Node 6	IDLE
2011-11-12;06:45:55.876	PROP4	Corina v1	6	Node 6	CONFIGURED

- Collected execution profiles for 10.000+ non-interactive batch runs
- Not used in interactive sessions, no monitoring of end-user behavior

Visualization – Gantt chart

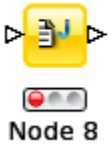
Boehringer Ingelheim





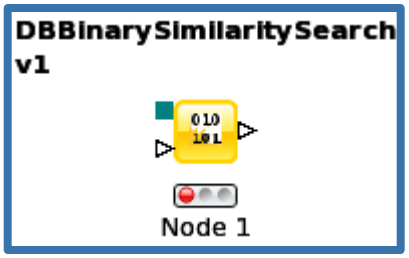
Results – Top 10 nodes (by number of executions)

Boehringer Ingelheim

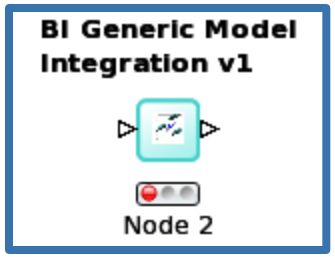
Java Snippet  Node 8	161.824	**Row Filter**  Node 11	37.590
Column Filter  Node 10	65.282	**GroupBy**  Node 4	25.774
Rename  Node 3	41.138	**Reference Row Filter**  Node 1	24.773
Joiner  Node 5	40.672	**Double To Int**  Node 8	21.351
JPython Script 1:1  Node 3	38.530	**Concatenate**  Node 9	19.952

Results – Top 10 nodes (by consolidated runtime)

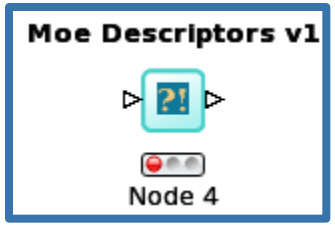
Boehringer Ingelheim



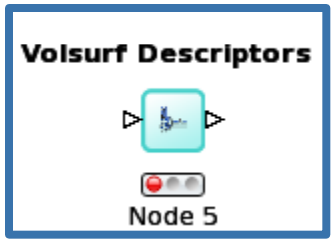
522.013 sec



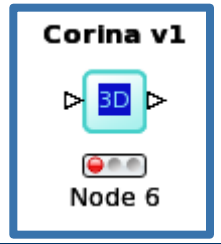
419.561 sec



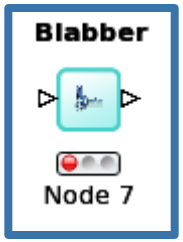
291.633 sec



212.338 sec



138.287 sec

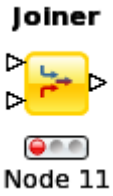


75.838 sec

66.679 sec

58.606 sec

55.274 sec



53.019 sec

Overview

Boehringer Ingelheim

- Motivation
  - History & current status
  - Why “Workflow Mining”?
- “Static Workflow Mining”
  - Extraction of workflow graphs
  - Frequent sub-graph mining
  - Results
- “Dynamic Workflow Mining”
  - KNIME Profiler Plugin
  - Analysis of profiling logs
- **Conclusions and Outlook**

Conclusions and Outlook

Boehringer Ingelheim

- "Static Workflow Mining" serves as a tool to identify typical patterns in KNIME workflows
  - Supports prioritization of bug reports / feature requests
  - Identification of best practices and common pitfalls
  - Plans to provide useful Metanodes based on frequent patterns

- "Dynamic Workflow Mining" monitors workflow behavior under real-world conditions
  - Identification of bottlenecks and critical steps in workflows
  - Performance optimizations for batch-processes planned
  - Additional option to perform real-time monitoring of system health

- Large workflow collection an asset, but also a burden
  - Have to ensure compatibility of new node versions

Acknowledgements

Johannes Koppe (IS BP R&DM)
Andreas Teckentrup (IS BP R&DM)
Bernd Wiswedel (KNIME.com)

All our KNIME users for their feedback, their suggestions und their patience!