

# The use of KNIME to support research activity at Lhasa Limited

Data processing through to proof-of-concept implementations

Sam Webb

[samuel.webb@lhasalimited.org](mailto:samuel.webb@lhasalimited.org)



# Overview

---

- The Lhasa-KNIME timeline
- Internal KNIME node development
- Use cases
- Example of proof-of-concept developments
- Some use cases acquired from Lhasa employees
  - Opinions are my own



## Who is Lhasa?

- Not-for-profit organisation
- Develop software for the prediction of toxicity, metabolism, degradation and supporting databases
- Undertake data sharing initiatives
- 'Head office' in Leeds UK

<http://www.lhasalimited.org>





# Who am I?

- Working in the Research Group
  - Cheminformatics, data analysis & machine learning
  - Develop new libraries / tools to support my research activities
  - Share these developments with others at Lhasa
  - KNIME makes the sharing easier
    - But if we want heavy visualisation we may chose to prioritise into our internal cheminformatics platform and only provide some functionality in KNIME
- 

# Timeline

- 2011

- A few users within the Research Group
- Love of meta nodes begins

**You can do  
that in  
KNIME!**

- 2012

- Internal KNIME indoctrination training begins
- KNIME node development starts
  - Integrate our chemical engine
- First proof-of-concept: black box model interpretation
- Second proof-of-concept : negative predictions
- Yay, loops!

**Now  
implemented in  
our software.**

- 2013

- Second push for KNIME training, bigger uptake
- Evaluated the KNIME server

**I'll help you,  
but only if we  
use KNIME**

# The Lhasa-KNIME timeline

- 2014
  - Hit the 125 internal KNIME nodes
  - Bayesian network proof-of-concept
  - KNIME now popular with scientists
    - Training given to new employees, get them young...
  - Try to avoid using loops
  - Frequent requests: how do I do this not using a java snippet?
  - Attempt at using Pipeline pilot

Getting quicker at making these! But now I know what I did wrong in the first ones

Streaming is a nice concept (but I don't want to pay for it)

Why doesn't it store the data in the output!



# Timeline

---

- 2015
  - Jenkins build environment
  - Test workflows
  - Ongoing node development
  - Hosting the next Cheminformatics Special Interest Group meeting
    - May 2015 in Leeds
    - <http://www.knime.org/cheminformatics-workshop-may-2015>



## What do the KNIME users look like now?

- A lot of people KNIME
- 2 developers making nodes
- ~ 20 users: Research Group, knowledge scientists, database scientists

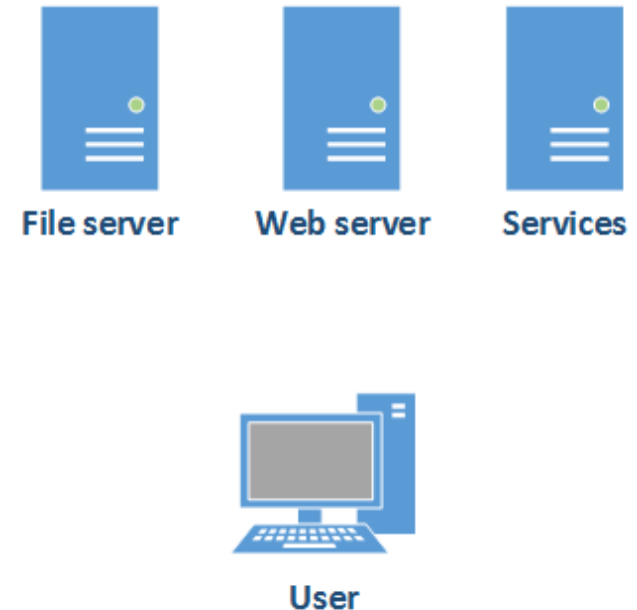




# KNIME INFRASTRUCTURE

# Our setup

- File server: sharing workflows
- Web server: running a Wiki
- Build machine: in the works
- Services:
  - Web services
  - Long jobs? ← under investigation
- User
  - Many different versions of KNIME
  - Many different version of plugins



# The Wiki

- Documentation of KNIME functionality, example workflows and procedures for common activities

## Usage

The KNIME node is very crude, it runs the command line configuration for MOPAC which causes MOPAC to load. The row is processed, MOPAC closes and then the next row loads a new instance of MOPAC.

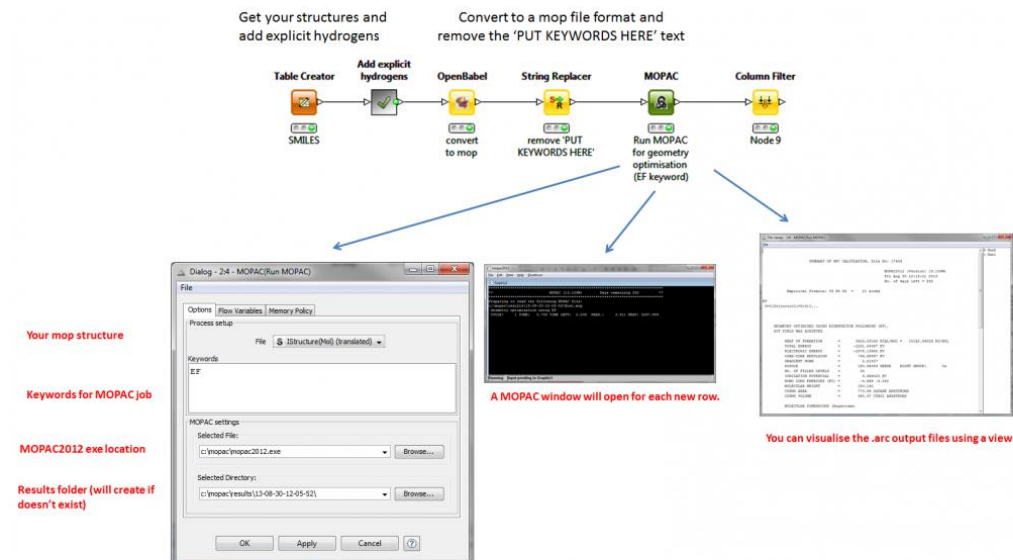
Therefore the node can only really be used for small numbers of structures. For larger datasets you will need to run it overnight.

The node has been configured to process a mop (MOPAC structure format) column. The keywords specified in the dialog will be added in and the complete mop will be run through MOPAC2012.exe. I have got the node to read in the .arc output file and store this as a string. A view allows you to easily browse the results. Currently the following values are automatically extracted from the output file and added to the table: energy, HOMO, LUMO, Alpha SOMO, Alpha LUMO, Beta SOMO and Beta LUMO.

**Make sure your structures have explicit hydrogens.**

## Workflow

It's best not to run this node when you need your machine. Currently an instance of MOPAC will load on your main monitor above all other windows when a new line is processed.

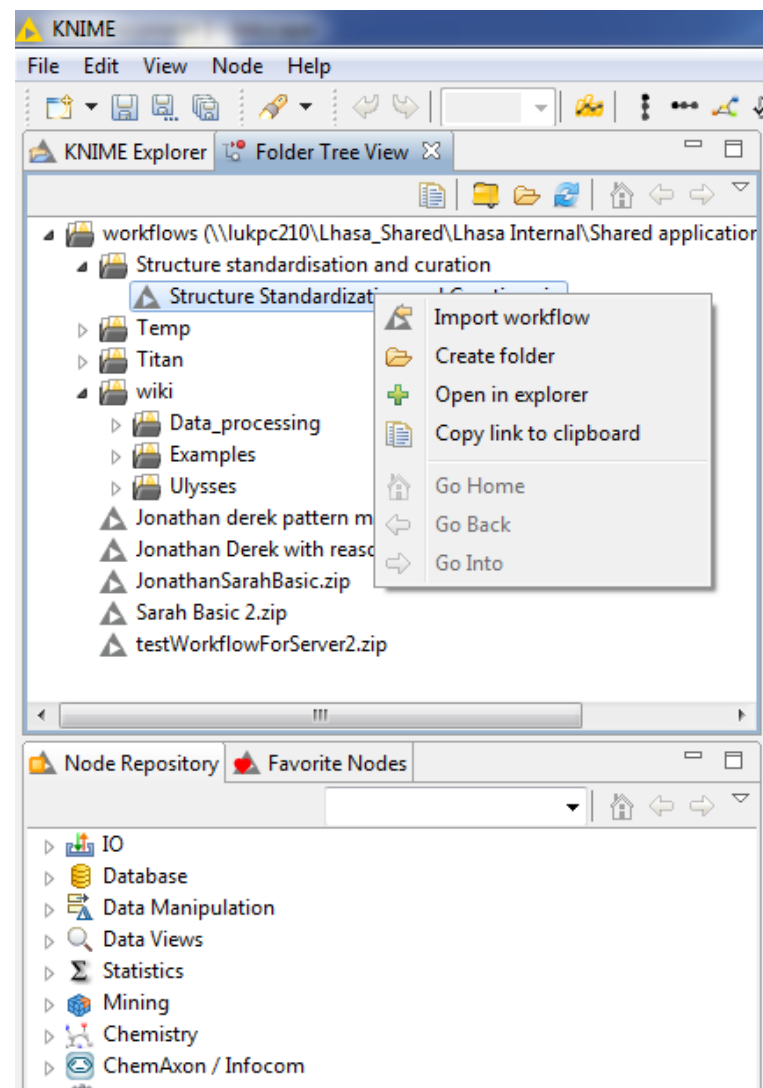


This workflow will provide you with a table with the energy, HOMO and LUMO of the structures given (if successfully calculated). The .arc output files can be viewed in the 'File viewer' view. On the right of the view you can choose the file (row index) - ID in mop file.

The .out, .arc and .mop files can be found in your chosen results directory.

# Workflow sharing

- Shared workflow area
- Import directly from the shared workflow area into your KNIME workspace
- In house addition/extension to KNIME

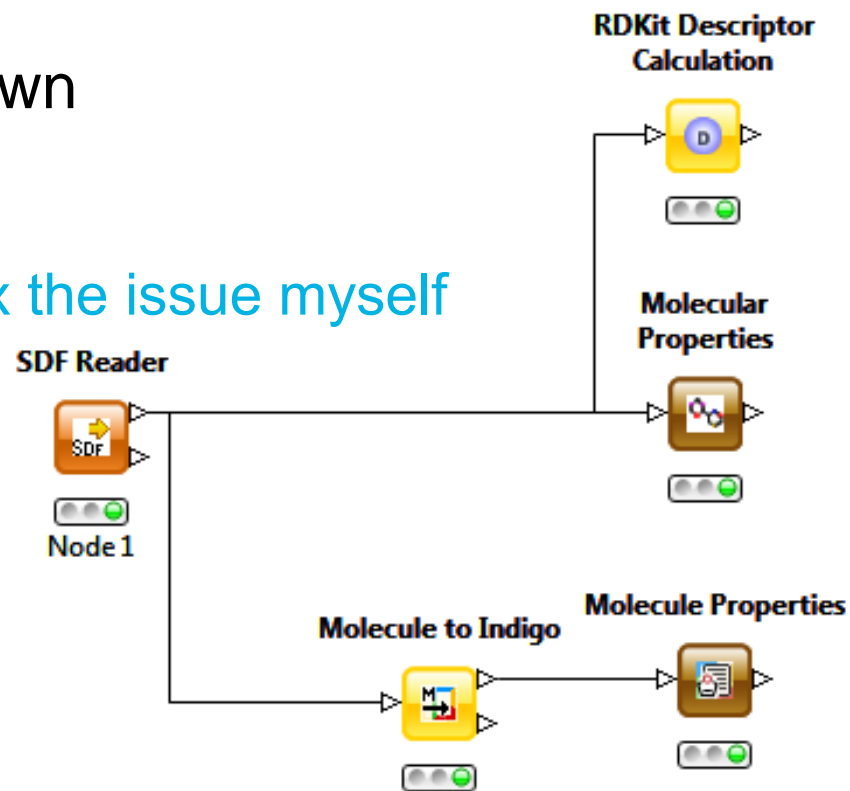




# USE CASES

# Lhasa use cases: descriptor calculation

- Reduce the number of programs needed to calculate descriptors
- Perform the data analysis where the descriptor calculation happens
- We have a few nodes of our own
- Avoid PaDEL
  - can't find the source code to fix the issue myself



## Lhasa use cases: structure curation

- KNIME has a number of chemical engines accessible: CDK, RDKit, Indigo, ChemAxon and our own
- We have developed structure curation workflows that automatically fix some issues and flag others for manual analysis
- Implemented our own nodes calling out to ChemAxon standardizer and Structure checker
  - <http://www.chemaxon.com/products/standardizer/>
  - <http://www.chemaxon.com/products/structure-checker/>

Validation



Lhasa node

Standardizer



Our implementation of  
ChemAxon toolkit

StructureChecker



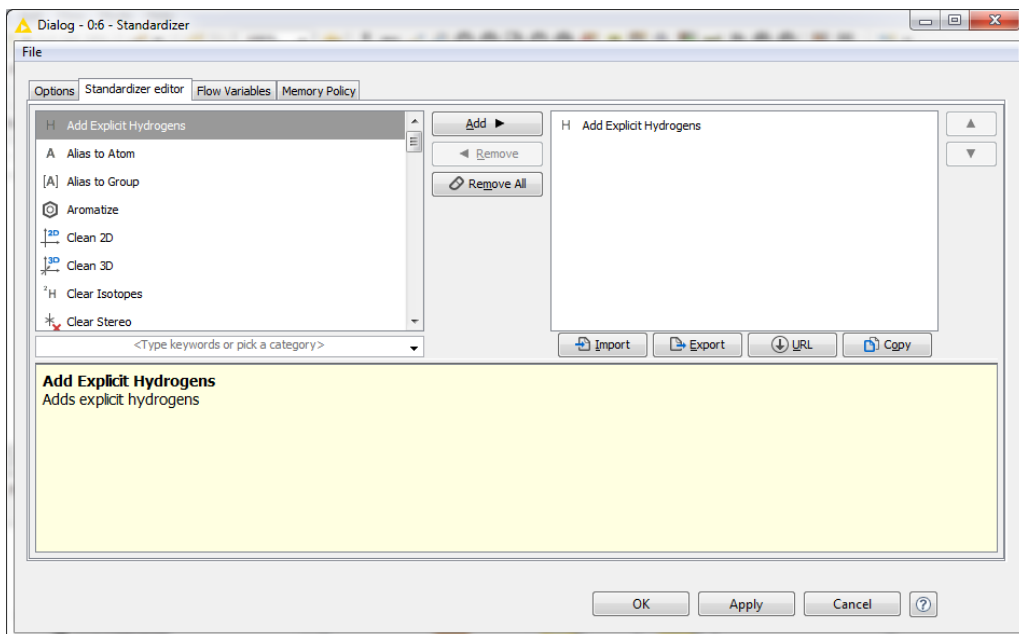
Our implementation of  
ChemAxon toolkit

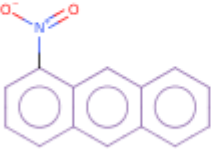
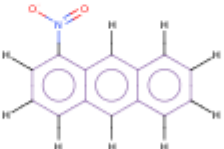
# Lhasa use cases: structure curation

**Standardiser editor  
available in KNIME.**

**Either configure or load in  
an XML configuration.**

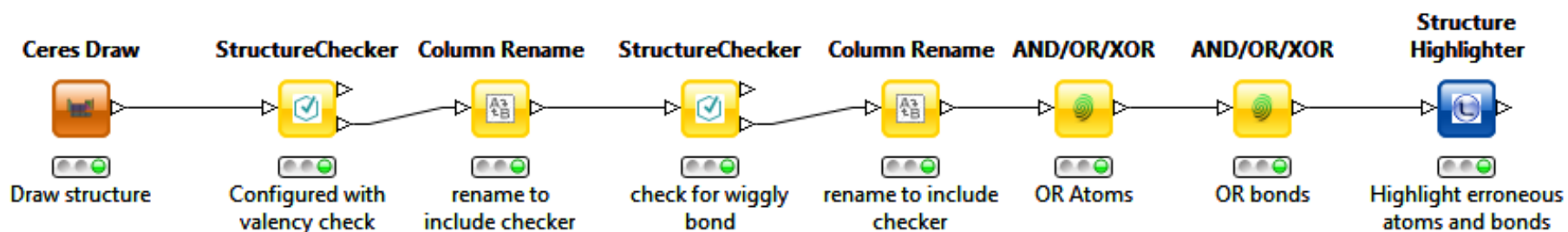
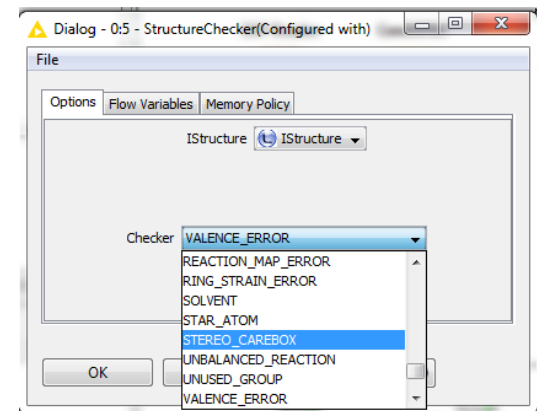
**Applies standardisation  
and lists completed  
tasks**



Row ID	IStructure	Standardized	Completed tasks
Row0			[Add Explicit Hydrogens]



# Lhasa use cases: structure curation



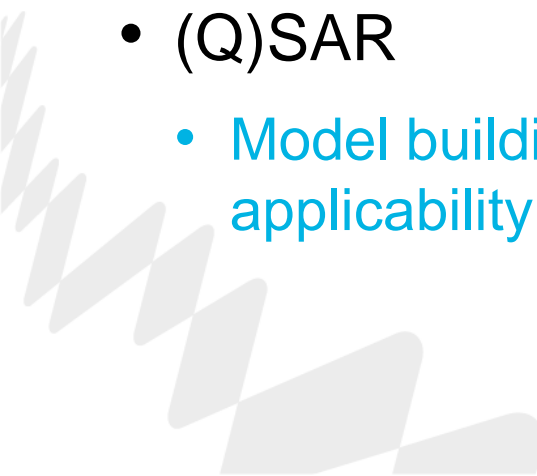
Out-Port - 0:8 - Structure Highlighter(Highlight erroneous)

Table "default" - Rows: 1 Spec - Columns: 12 Properties Flow Variables

Row ID	IStructure*	OR (Atoms - ...	OR (Bonds - ...	Error - ...	Description - v...	Atoms - vale...	Bonds - vale...	Error - ...	Description - wiggly
Row0		{10, 22}	{15}	VALENCE	2 valence errors fo...	{10, 22}	{}	WIGGLY	1 wiggly double bond fo...




# Lhasa use cases

- Data processing:
    - Combining datasets: find overlap, compare activities when overlap exists, join in data where no overlap exists...
    - Making overall calls: lots of results for a compound, combine into a single result based on defined rules
  - Monitoring:
    - Extracting data from a the database which has been altered identifying review work content
  - (Q)SAR
    - Model building, clustering, algorithm development, applicability domains, chemical space investigation....
- 



# Key benefits of KNIME

- All in one place (nearly)
  - Workflow = documentation
    - Only really works when data is present
    - Still need to annotate!
    - Workflows can be overly complex and tricky to understand
  - Quickly share new updates to internal code developments with users
    - Doesn't mean they will actually update though...
  - Actions are reversible! No overriding of data, just change a configuration of a node / replace the node!
- 

# Issues we've had

- People can develop super messy workflows
  - I hide mine behind meta nodes
- Memory can sometimes be an issue
- Reporting can be painful
- Looping in loops can be very slow
- 32 bit requirements on some 3<sup>rd</sup> party DLL's
- Getting people to stay up to date
- Remembering to look up what other words a node may go by
  - I have at least on one occasion made a node that already existed...


**Column Appender**



  
**KNIME node**

**DumbJoiner**



  
**My node**



# NODE DEVELOPMENT



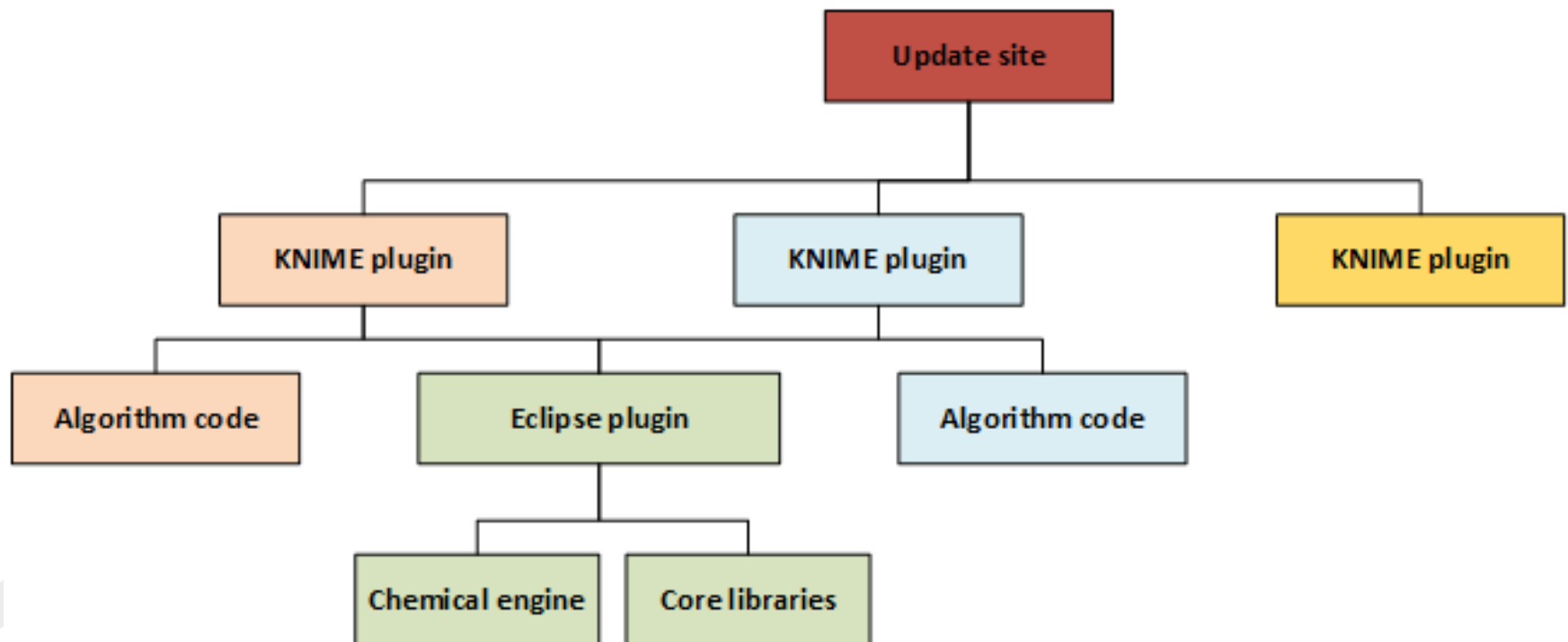
## Node development

---

- SVN for versioning of our KNIME nodes
- Old build machine for building the update site – plan to switch to a Jenkins build process
  - Thanks to Vernalis for help with this!
- Everyone uses a different version of KNIME and plugins
  - Our version of “have you turned it off and on again is”: “have you updated?”

# Workflow and node development

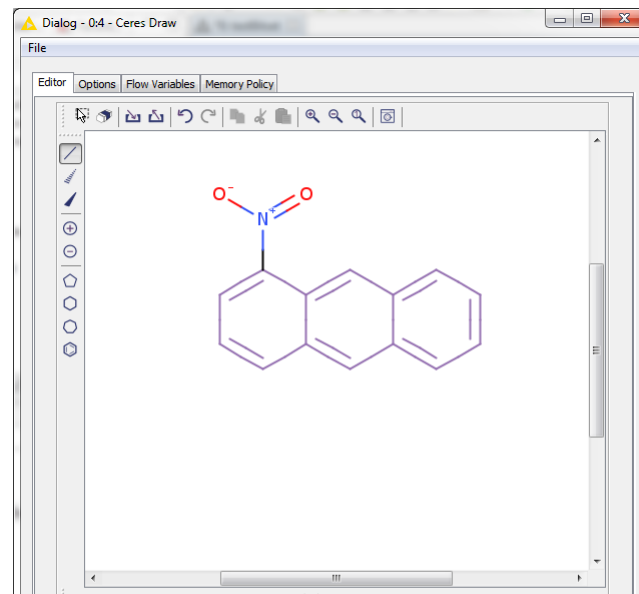
- We now have over 100 nodes
- We also have some Eclipse RCP additions (views)



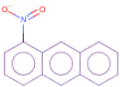
# Chemical engine integration

- We have our own Java based chemical engine and have added a some of its functionality to KNIME
  - Structure conversion
  - Inchi generation
  - Structure drawing
  - Structure validation
  - Fingerprint generation
  - Property calculation
  - Similarity calculation
  - Tautomer generation
  - Fragmentation

**Overlaps with  
RDKit, Indigo  
and CDK**



A screenshot of a KNIME table view titled "Out-Port name - 0:5 - Properties engine integration". The table has 12 columns representing various chemical properties. The first column is "Row ID" with the value "Row0". The second column is "Structure", which contains a small chemical structure of 1-nitrofluorene. The remaining 10 columns are numerical values for different properties.

Row ID	Structure	nAtoms	nBonds	nHeavy...	nRings	nRingA...	nNonRi...	nExplici...	nCarbon	nHetero	nNonRi...
Row0		17	19	17	3	14	3	0	14	3	0





# ChemAxon integration

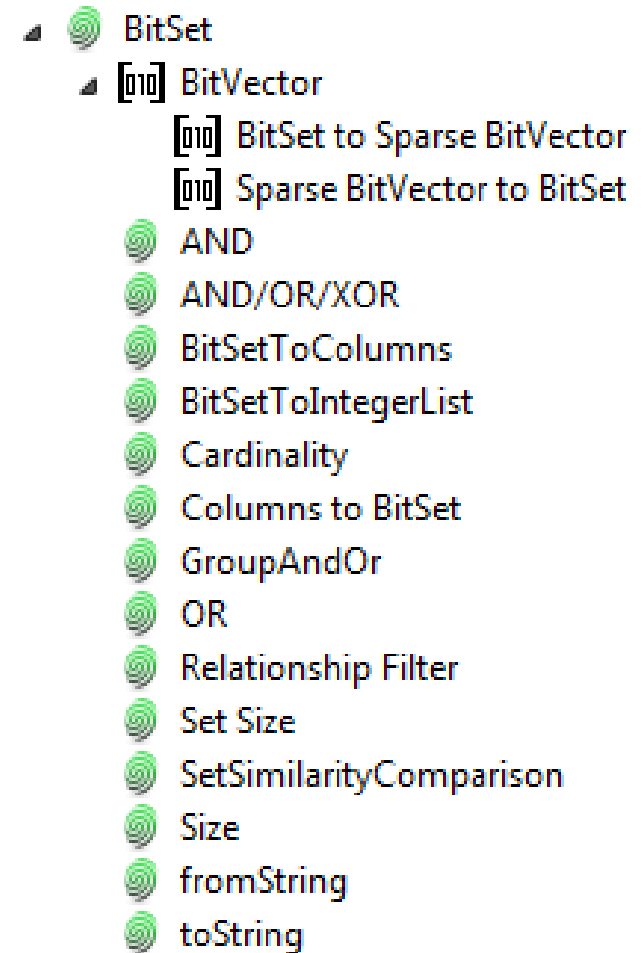
---

- We licence a number of ChemAxon components
- We develop our own nodes...
- We didn't want to have to pay to licence the ChemAxon tool twice
- We wrote our own ChemAxon based nodes



# BitSet integration

- A lot of our code uses BitSets or SparseBitSets
- The KNIME BitVector wasn't very well developed in functionality
- We created a number of nodes operating on BitSets and conversion between BitVector and BitSet
- BitVectors are now more developed in functionality and some overlap exists
- Some of this functionality would be better implemented as extensions e.g. new actions for the GroupBy node



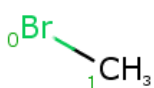
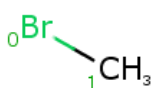
# BitSet example

- Same feature occurs in multiple locations
- Group on the feature ID and perform an OR operation on the BitSet

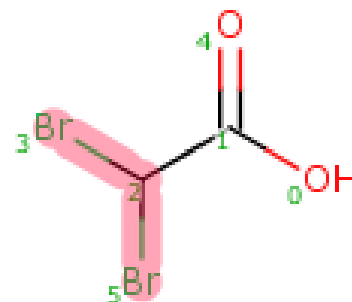
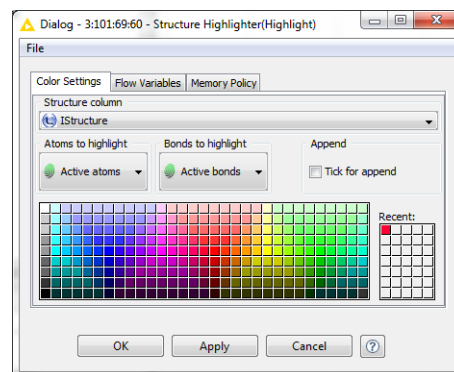
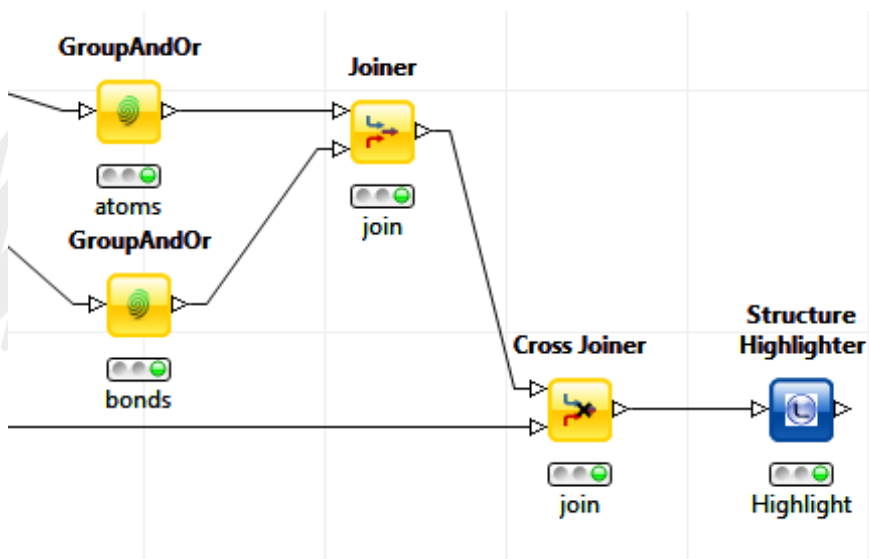
Network nodes - 3:101:69:48 - Hierarchical Interpretation

File

Table "default" - Rows: 2 Spec - Columns: 8 Properties Flow Variables

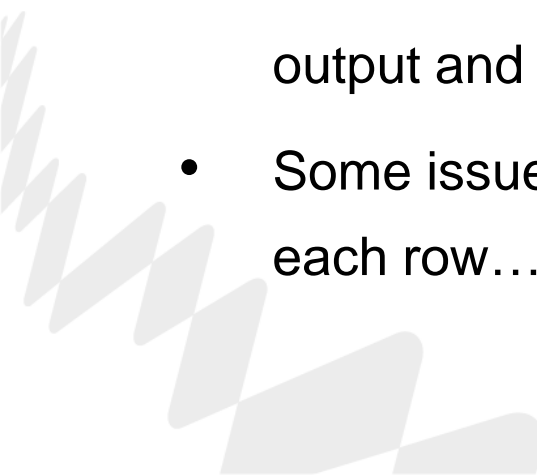
Row ID	ID	Type	Active fragment	Active atoms	Active bonds
0	67027	ACTIVATING		{2, 3}	{3}
1	67027	ACTIVATING		{2, 5}	{4}

**We can now highlight all the ACTIVATING atoms**

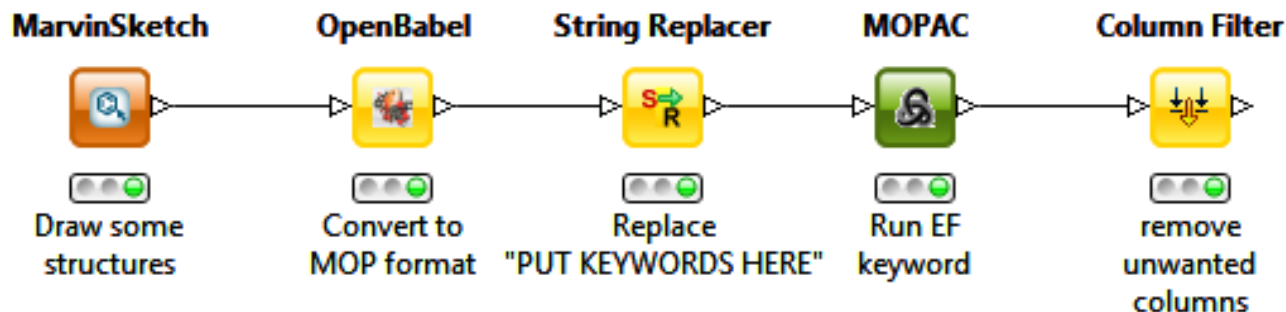




# MOPAC integration

- “MOPAC (Molecular Orbital PACkage) is a semiempirical quantum chemistry program based on Dewar and Thiel's NDDO approximation” - <http://openmopac.net/>
  - We wanted to access to some of the values MOPAC can calculate
  - Tried to make it easier for people to use
  - Made a node / workflow that calls off to MOPAC, parses the output and creates a table of results
  - Some issues such as it opening a new instance of MOPAC for each row...
- 

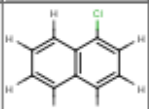
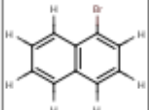
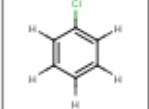
# MOPAC example



Filtered table - 0:14 - Column Filter(remove)

File

Table "default" - Rows: 3 | Spec - Columns: 11 | Properties | Flow Variables

Row ID	Mol Molecule	D Energy ...	D HOMO ...	D LUMO (...)	D SOMO ...
Row0		-1,589.388	-9.096	-0.862	?
Row1		-1,544.722	-9.043	-0.863	?
Row2		-1,071.14	-9.818	-0.249	?

File viewer - 0:13 - MOPAC(Run EF)

File

```

SUMMARY OF PM7 CALCULATION, Site No: 19412

MOPAC2012 (Version: 14.212W)
Thu Feb 05 12:08:37 2015
No. of days left = 176

Empirical Formula: C10 H7 Cl = 18 atoms

EF

GEOMETRY OPTIMISED USING EIGENVECTOR FOLLOWING (EF).
SCF FIELD WAS ACHIEVED

HEAT OF FORMATION = 29.88288 KCAL/MOL = 125.02998 KJ/MOL
TOTAL ENERGY = -1589.38799 EV
ELECTRONIC ENERGY = -8109.87628 EV
CORE-CORE REPULSION = 6520.48829 EV
GRADIENT NORM = 0.75979
DIPOLE = 2.01313 DEBYE POINT GROUP: Cs
NO. OF FILLED LEVELS = 27
IONIZATION POTENTIAL = 9.096430 EV
HOMO LUMO ENERGIES (EV) = -9.096 -0.862
MOLECULAR WEIGHT = 162.618
COSMO AREA = 182.07 SQUARE ANGSTROMS
COSMO VOLUME = 186.05 CUBIC ANGSTROMS

MOLECULAR DIMENSIONS (Angstroms)
  
```

# SmartCyp & WhichCyp

## Predicting CYP sites of metabolism

<http://www.farma.ku.dk/smartcyp/>

WhichCyp 1.0



SmartCyp2\_4\_2



SmartCyp-Ext



Out-Port - 3:103 - SmartCyp-Ext

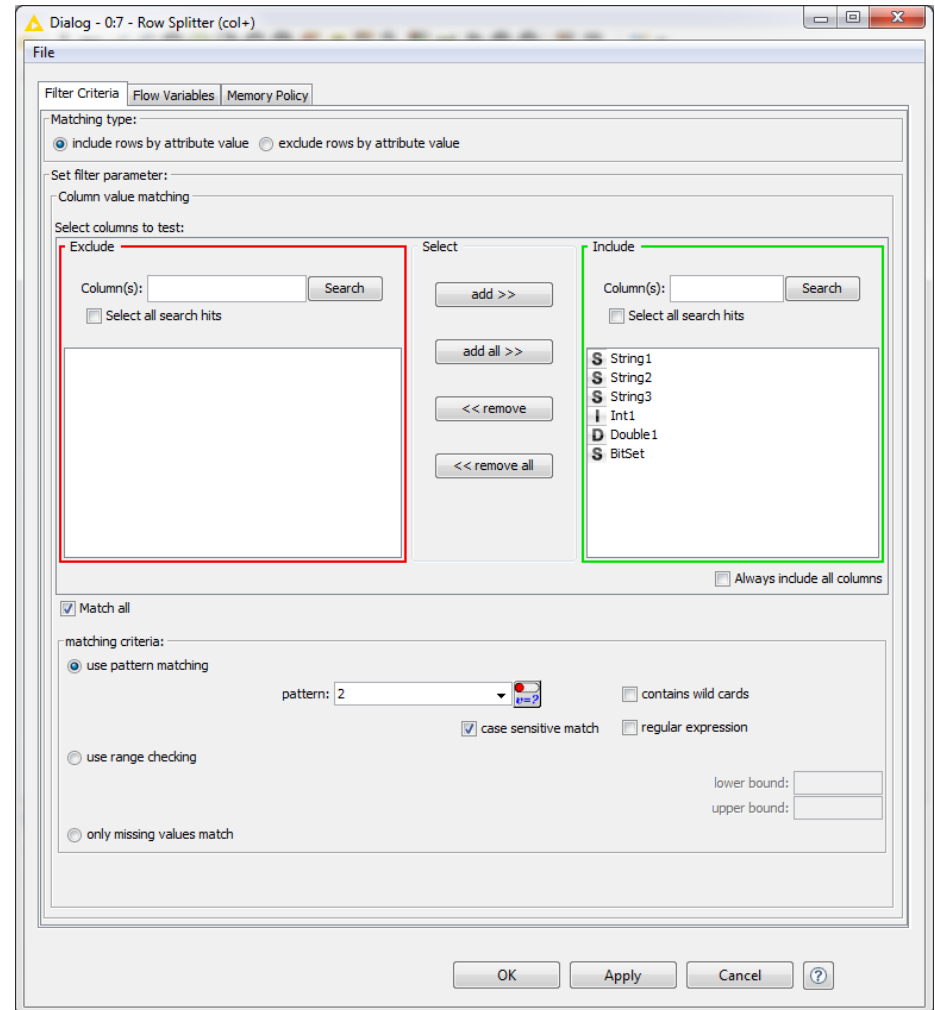
File

Table "default" - Rows: 3 Spec - Columns: 21 Properties Flow Variables

Row ID	Substrate id	Substrate	Atom ID	Atom t...	Ranking	Score	Energy	Relativ...	2D6 ra...	2D6 score	Span2End	N+ Dist	2C9 ra...	2C9 score	CO Dist	2C19 r...	2C19 s...	CO Dist 2
Row0_0	Row0, component 0		14	N	2	47.88	54.1	0.78	3	67.5	2	0	3	67.5	0	3	67.5	0
Row0_1	Row0, component 0		18	C	3	51.7	59.7	1	2	59.7	0	0	2	59.7	0	2	59.7	0
Row0_2	Row0, component 0		20	C	1	32.69	39.8	0.89	1	46.5	1	0	1	46.5	0	1	46.5	0

# Generic nodes: multi column row splitter

- Attribute matching taken from the row filter
- Select multiple columns to apply filter to
- Include / exclude based on matching one or all columns



# Generic nodes: model performance

- Similar functionality to the Scorer node
- Calculates various performance metrics for binary classification models
- Can choose multiple prediction columns

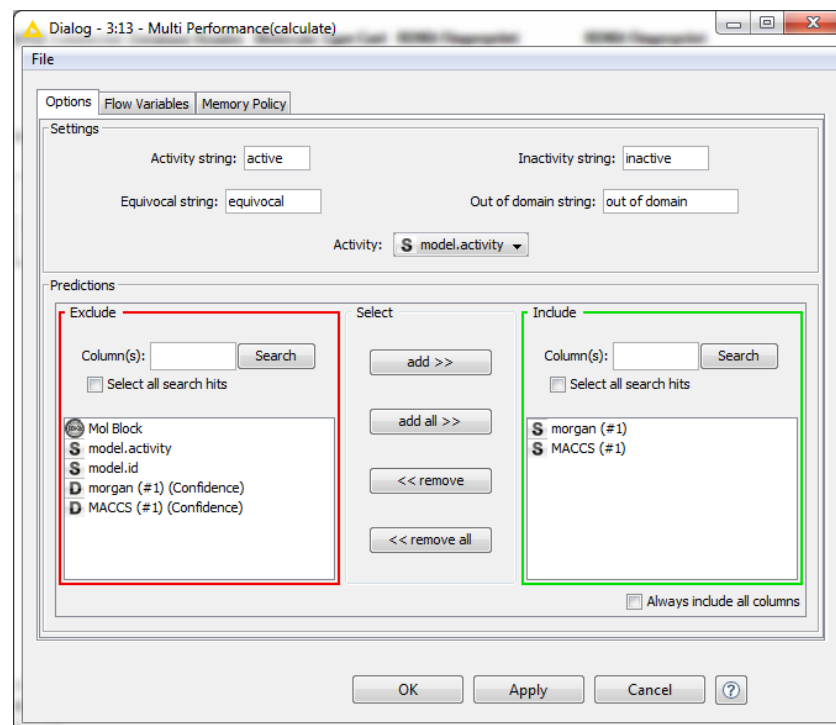


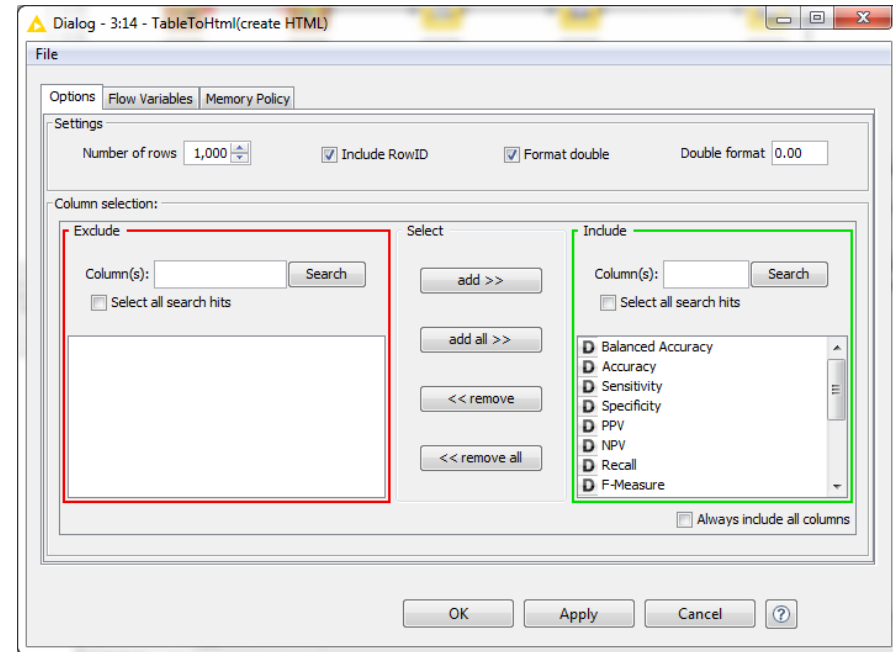
Table "default" - Rows: 2 | Spec - Columns: 15 | Properties | Flow Variables

Row ID	D Balance...	D Accuracy	D Sensitivity	D Specificity	D PPV	D NPV	D Recall	D F-Meas...	D Total	D TP	D FP	D TN	D FN	I Equivocal	I Out of ...
morgan (#1)	0.773	0.78	0.848	0.698	0.773	0.791	0.848	0.809	1,296	602	177	409	108	0	0
MACCS (#1)	0.788	0.793	0.838	0.739	0.795	0.79	0.838	0.816	1,296	595	153	433	115	0	0



# Generic nodes: table to HTML

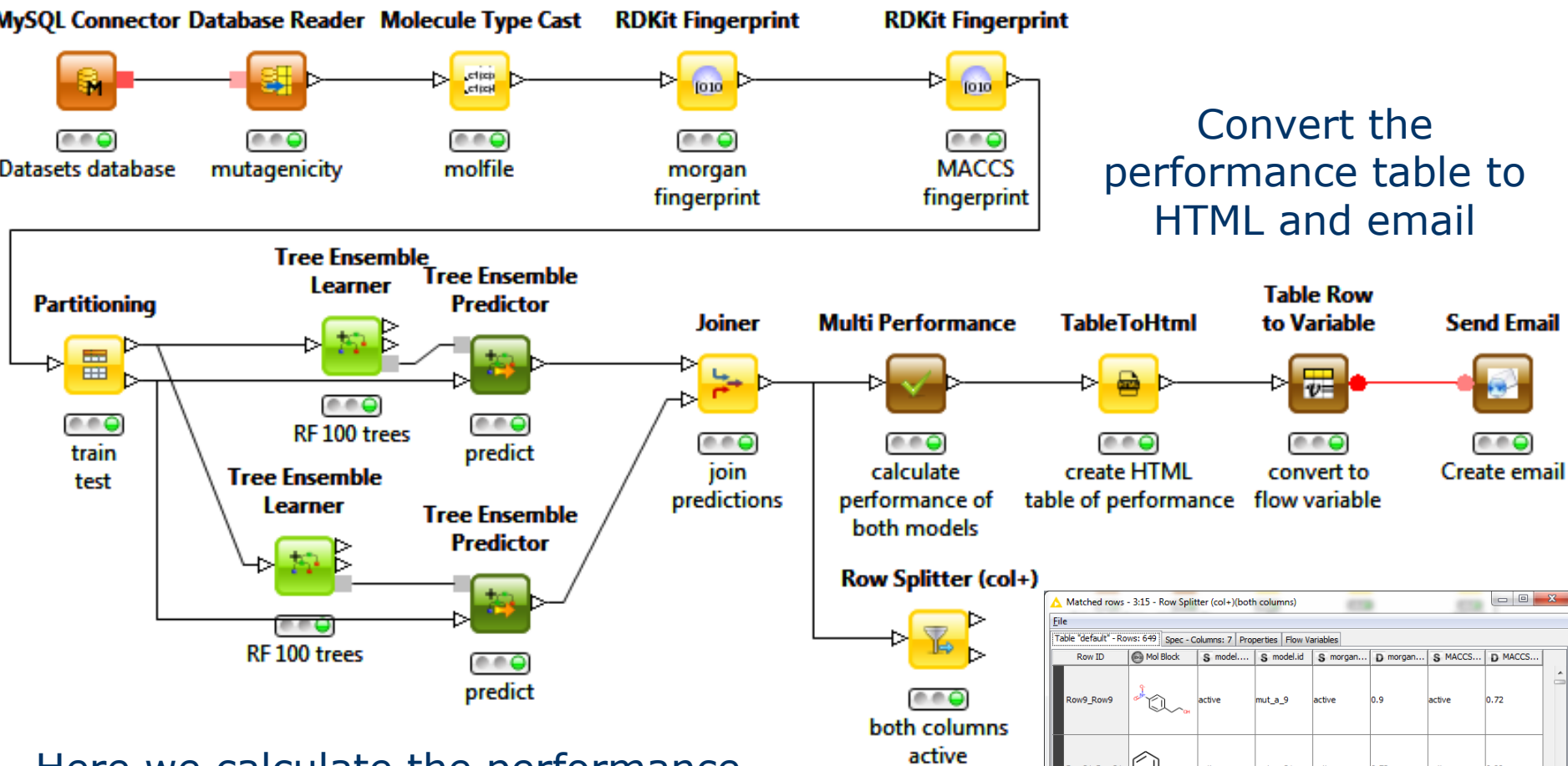
- Convert a table to a single HTML cell
- The String render will render HTML tags
- Select which columns to include
  - `StringValue`, `IntValue`, `DoubleValue`
- Creates a single cell output



18

# Where would you use these nodes?

Convert the performance table to HTML and email



Here we calculate the performance of the Random Forest with morgan and MACCS fingerprints

Filter out rows where either model predict active

Matched rows - 3:15 - Row Splitter (col+)(both columns)

Row ID	Mol Block	\$ model...	\$ model.id	\$ morgan...	\$ morgan...	\$ MACCS...	\$ MACCS...
Row9_Row9	<chem>Oc1ccc(cc1)C(=O)N</chem>	active	mut_a_9	active	0.9	active	0.72
Row21_Row21	<chem>Nc1ccccc1O</chem>	active	mut_a_21	active	0.72	active	0.88
					0.84	active	0.85
					0.96	active	0.91



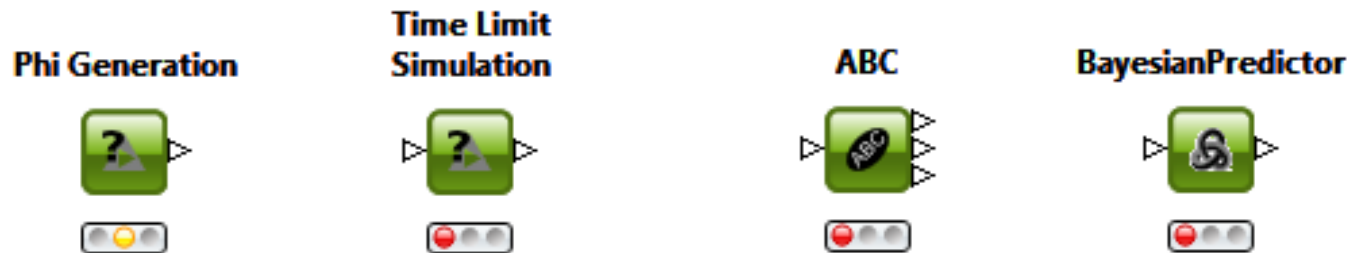
# PROOF OF CONCEPTS

# Bayesian networks: background

- Had a project investigating combining human experts opinion and data
- Some of this work could be done using R or python libraries
  - They were a bit messy to use and not as convenient
- Core Bayesian algorithms available from SMILE and Genie: <https://dslpitt.org/genie/>
- Implemented some algorithms in Java to learn the posterior
  - Markov Chain Monte Carlo simulation
  - Approximate Bayesian Computation

# Bayesian Networks: where does KNIME come in?

- Using KNIME as a method for allowing non coders to investigate various configurations of experimental codebases
- KNIME nodes developed providing a user interface for the Bayesian Network libraries (in house)



- We would end up doing the data analysis in KNIME anyway
- Can now let users who are less comfortable with command line applications use the code













# Bayesian networks

- I'm beginning to dislike loops.
  - If it needs a complex loop maybe I should write a node to do it?
  - Looping in loops can be very slow!
    - One particularly bad loop had a 6 hour runtime, it takes < 5 seconds as its own node. I suspect I made bad choices in the development of the workflow...
  - Complex loops may result in multiple points of failure
- We had a parameter grid for the Bayesian network optimisation.
  - We could use existing KNIME nodes to loop over the grid.
  - We could do batch processing!

# Bayesian networks: batch processing

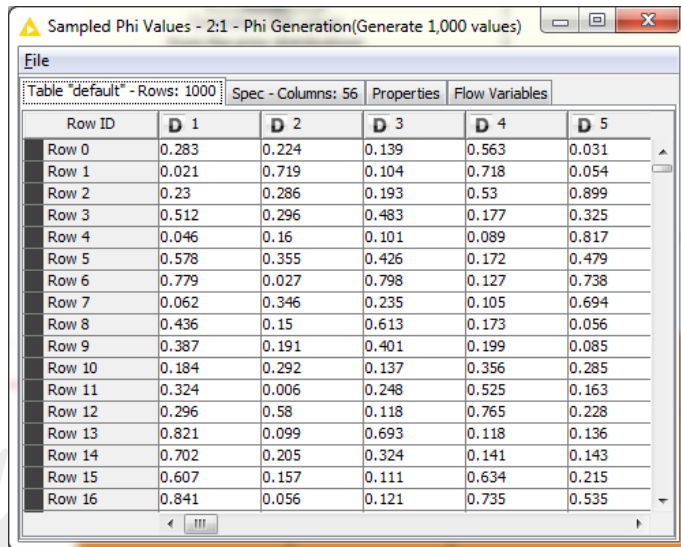
- Write a batch script to iterate through the parameters
- The workflow saves a number of files (csv, svg and png) per run

```
1 set WORKFLOW_FILE=workflow.zip
2 set saveLoc=C:\\container\\current\\bayesian\\skinIrritation\\irrCorr\\experiments\\priorDistributions\\uniform_expert\\
3
4 set TRAIN_FILE=%saveLoc%train.csv
5 set PRIOR_FILE=%saveLoc%prior.csv
6 set TEST_FILE=%saveLoc%test.csv
7
8 set METHOD=JAVA
9 set distance=1.8
10 set numSamples=1000
11 set folder=output_%method%_%numSamples%_%distance%
12 set workflowDest=%saveLoc%%folder%\\workflow\\
13 C:\\container\\knime\\knime_2.10.1\\knime.exe -destDir=%workflowDest% -reset -preferences="prefs.epf" -consoleLog -nosplash -application org.knime.product.KNIME_BATCH_APPLICATION -work:
14
```

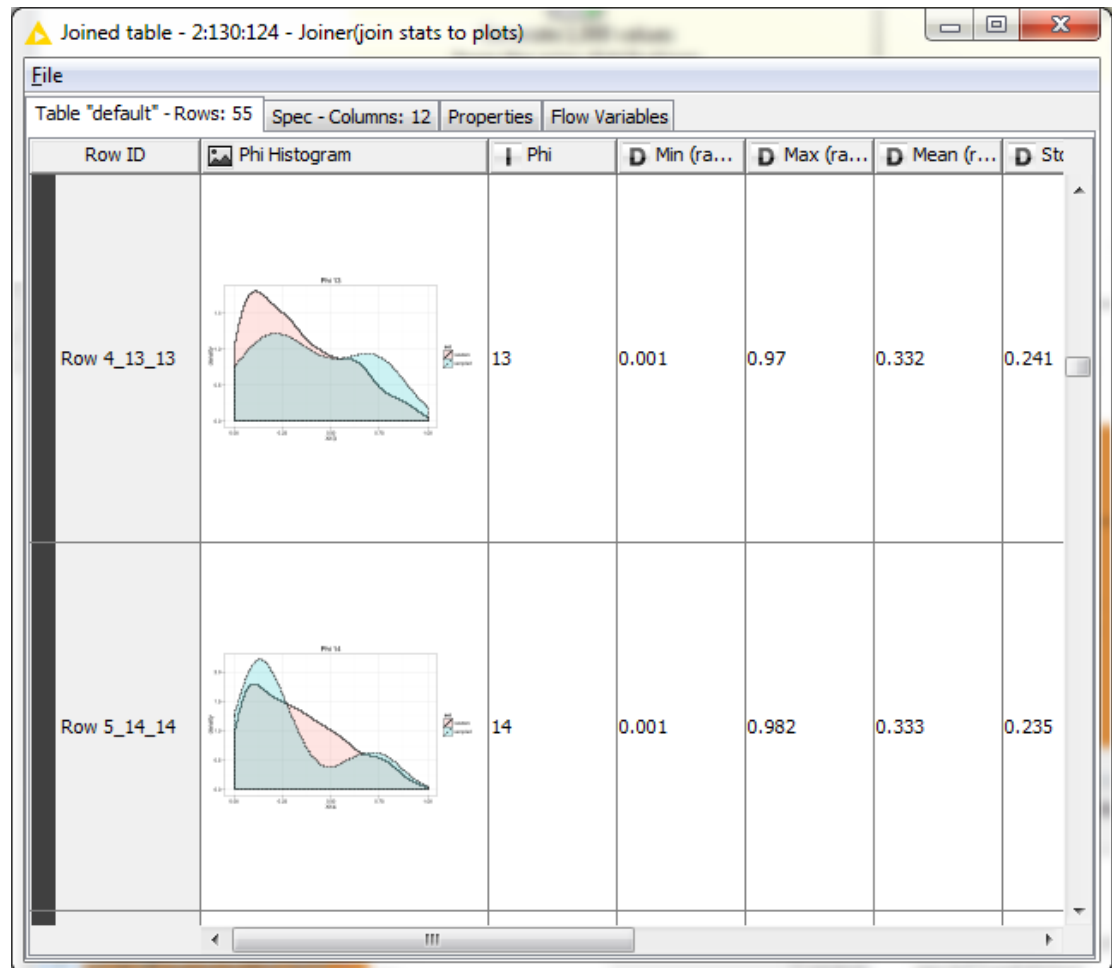
	phiValues	04/12/2014 14:04	File folder	
	workflow	04/12/2014 14:18	File folder	
	confusionMatrix	04/12/2014 14:14	Microsoft Excel C...	1 KB
	distanceHistogram	04/12/2014 14:04	PNG image	6 KB
	phiStatistics	04/12/2014 14:04	Microsoft Excel C...	12 KB
	phiValues	04/12/2014 14:04	Microsoft Excel C...	1,093 KB
	scorer	04/12/2014 14:14	Microsoft Excel C...	1 KB
	simulatedData	04/12/2014 14:04	Microsoft Excel C...	339 KB
	simulationLinePlot	04/12/2014 14:05	PNG image	10 KB
	uncertaintyStatistics	04/12/2014 14:17	Microsoft Excel C...	1,097 KB

# Bayesian networks: workflow snippets

**Sample from 56  
distributions 1000  
times**

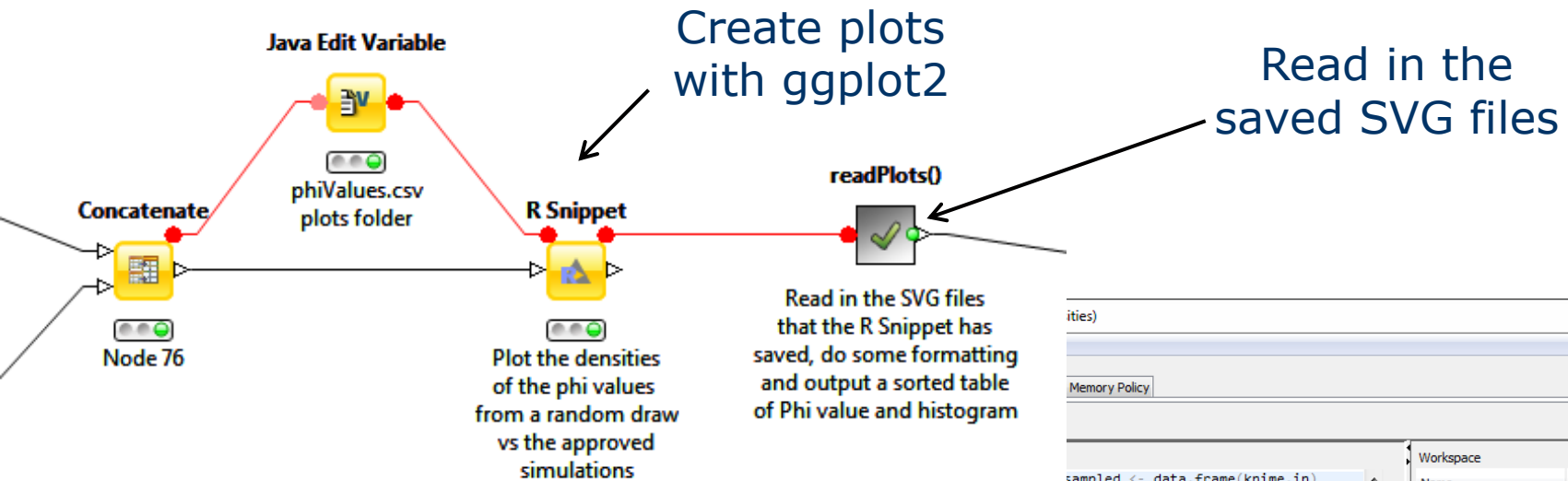


**Calculate statistics and density plots**





# Bayesian networks: workflow snippets



**We are creating 1 image per column. We could loop and get the image back in KNIME.**

**Instead we do the loop in R and save to a temp location.**

```
3 library(ggplot2)
4 print(random_sampled[,2])
5 ls(random_sampled)
6
7
8
9
10 for(i in 1:(length(random_sampled) - 2))
11 {
12   val <- paste("X", i, sep=" ")
13
14   myPlot <- ggplot(random_sampled, aes(x=val))
15   geom_density(alpha = 0.2, size=1)
16   theme_bw() +
17   xlim(0, 1) +
18   ggtitle(paste("Phi ", i, sep=" "))
19
20   ggsave(file=paste(knime.flow.in[["v_plots"]], i, ".svg", sep=" "))
21 }
22
```

Workspace

Name	Type
knime.flow.in	pairlist
knime.in	data.frame

Flow Variable List

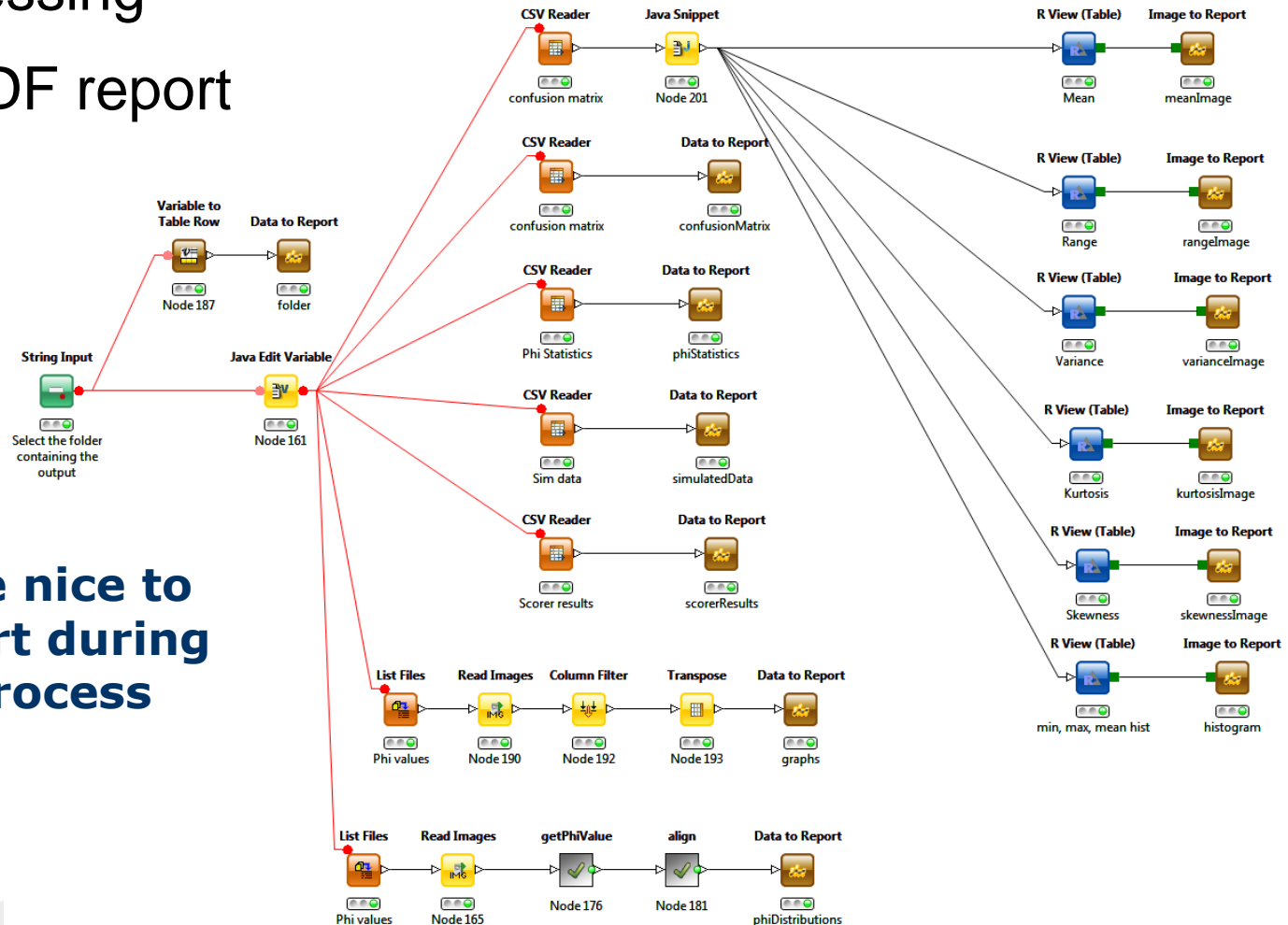
- v\_plots
- saveDir
- numSamples
- distanceCutoff
- simulationMethod
- knime.workspace
- uncertaintyPredictionsFile
- phiStatisticsFile

Console

OK Apply Cancel ?

# Bayesian networks: reporting

- Point to a directory and read in the files saved in the batch processing
- Create a PDF report



**It would be nice to  
save a report during  
a batch process**

# Bayesian networks: reporting

## ABC run output

Location: C:\container\current\bayesian\skinIrritation\IrrCorr\expertDistributions\train\_test  
 \noMissing\batch\output\_JAVA\_1000\_0.4

The folder name indicates the setup of the ABC job. output\_METHOD\_NUMSAMPLES\_DISTANCE. So JAVA\_1000\_1.0 is the JAVA simulation method, with 1000 samples at a distance of 1.0.

## Performance

Row ID	TP	FP	TN	FN	Sen	Spec	ACC	Cohen's kappa
NON_IRRIT	701	318	820	161	0.81	0.72		
CORROSIVE	324	114	1452	110	0.75	0.93		
IRRITANT	393	150	1146	311	0.56	0.88		
Overall							0.71	54.34%

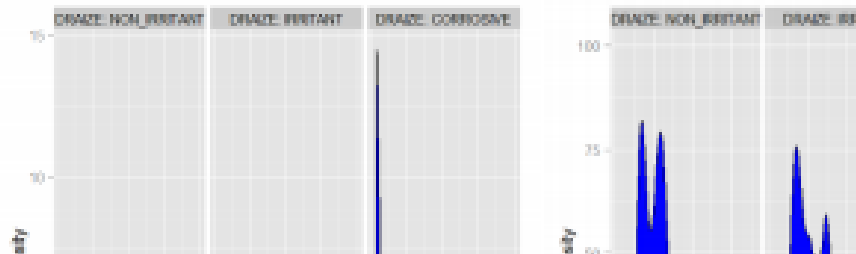
  

Experimental	NON_IRRITANT	CORROSIVE	IRRITANT
NON_IRRITANT	701	65	96
CORROSIVE	56	324	54
IRRITANT	262	49	393

## Uncertainty

An evaluation of the uncertainty of the predictions has also been performed. The KNIME statistics node was run on a per query basis on the test set. Each query having x predictions based upon each set of approved sampled phi values.

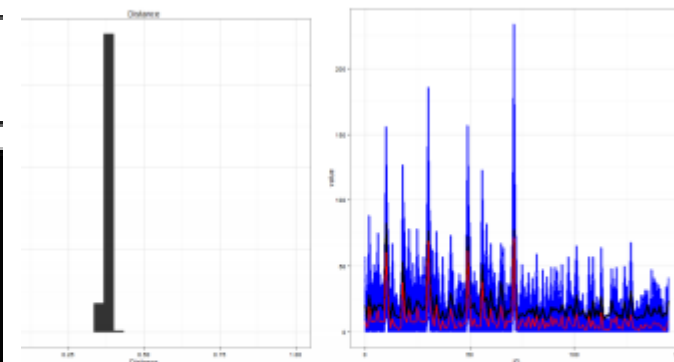
Density plots have then been calculated using ggplot2 (in R) on the entire test set (2,000 Drazize class is predicted separately).



## Simulated data

Distance histogram

Summary statistics. Black line data, red line mean, blue lines simulations

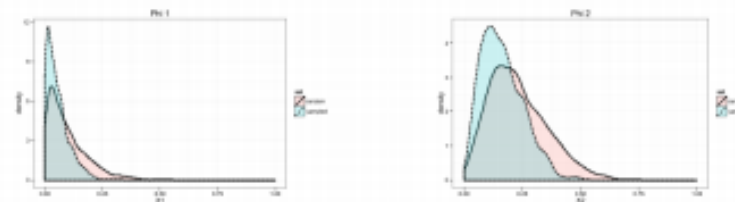


## Phi values

Min (random)	Max (random)	Mean (random)	Std. deviation (random)	Variance (random)	Min (sampled)	Max (sampled)	Mean (sampled)	Std. deviation (sampled)	Variance (sampled)
0.00	0.52	0.09	0.08	0.01	0.00	0.32	0.05	0.05	0.00
0.01	0.70	0.23	0.12	0.01	0.02	0.49	0.16	0.09	0.01
0.00	0.63	0.10	0.09	0.01	0.00	0.35	0.06	0.05	0.00
0.02	0.68	0.23	0.12	0.01	0.01	0.58	0.17	0.09	0.01
0.00	0.49	0.10	0.09	0.01	0.00	0.39	0.08	0.07	0.00
0.02	0.72	0.23	0.12	0.02	0.02	0.71	0.23	0.12	0.01
0.00	0.54	0.14	0.10	0.01	0.00	0.46	0.13	0.08	0.01

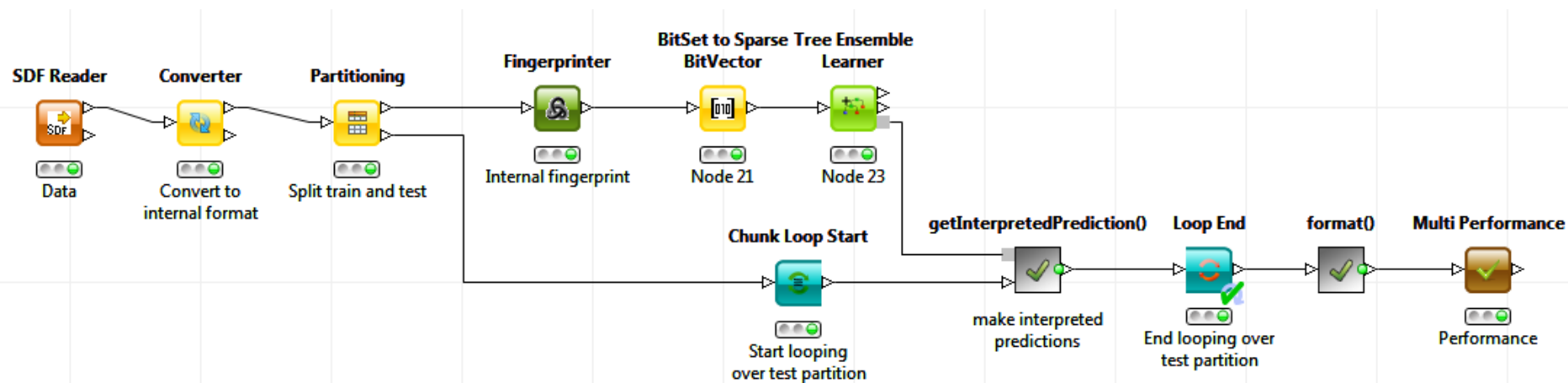
## Distributions

Comparison of posterior and prior distributions, the prior represents 1,000 samples from the prior and the posterior is the the distribution of the approved samples in the ABC run.



# Model interpretation

- Implemented the Similarity maps method by Riniker & Landrum
- <http://www.jcheminf.com/content/5/1/43>
- Summary:
  - Assign a contribution to an atom as the difference between the active class probability with the atom vs without the atom



# Model interpretation

Renamed/Retyped table - 2:46 - Column Rename

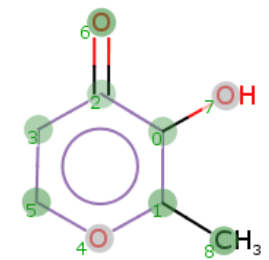
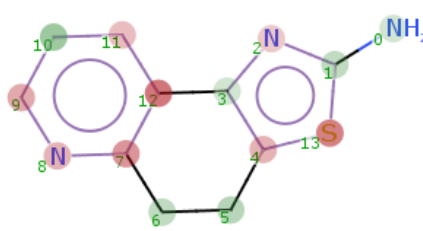
File Table "default" - Rows: 324 Spec - Columns: 7 Properties Flow Variables

Row ID	S model....	S model.id	D ProbA	D ProbI	S Prediction	Map	S HTML
Row0#186	inactive	mut_i_3759	0.5	0.5			
Row0#321	active	mut_a_6414	0.5	0.5	active		
Row0#16	active	mut_a_280	0.49	0.51	inactive		

**TableToHtml**

Convert atom number table to a single HTML cell

Table converted to a HTML string to then be rendered in the cell

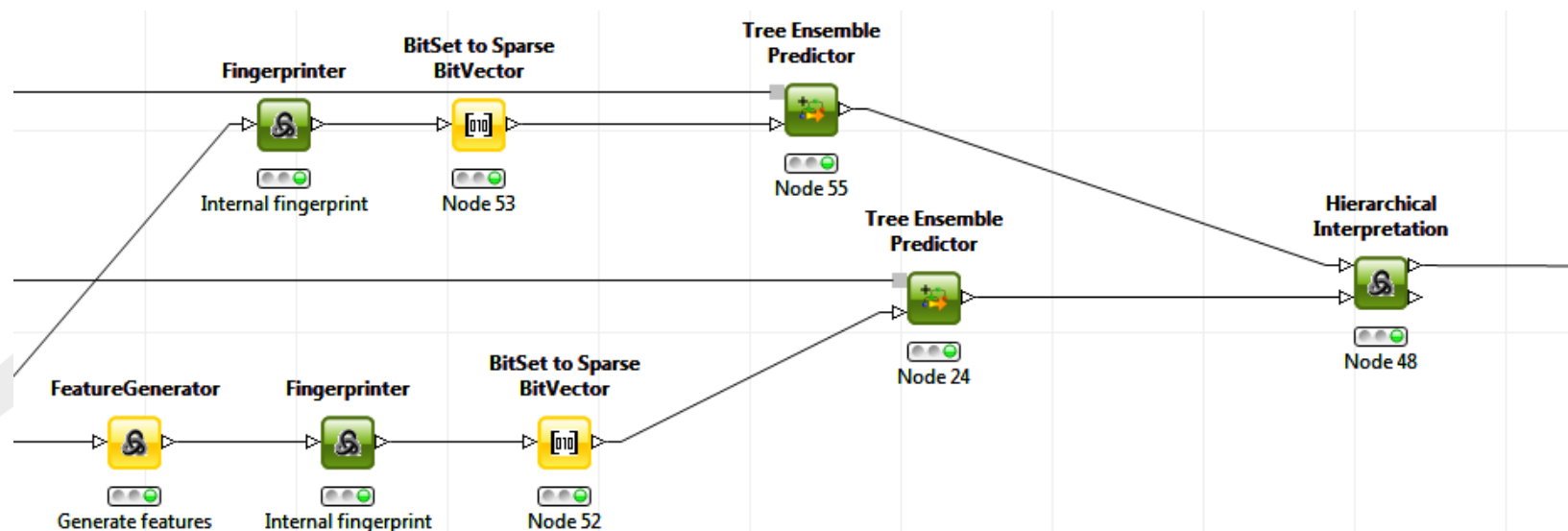


Atoms(5)	5	0.9000000000000005
Atoms(12)	12	0.6000000000000001
Atoms(13)	13	0.5
Atoms(2)	2	-0.1111111111111124
Atoms(3)	3	0.1111111111111124
Atoms(4)	4	-0.3333333333333376
Atoms(5)	5	0.4444444444444436
Atoms(6)	6	0.3333333333333376
Atoms(7)	7	-0.6666666666666675
Atoms(8)	8	-0.3333333333333376
Atoms(9)	9	-0.4444444444444445
Atoms(10)	10	0.7777777777777781

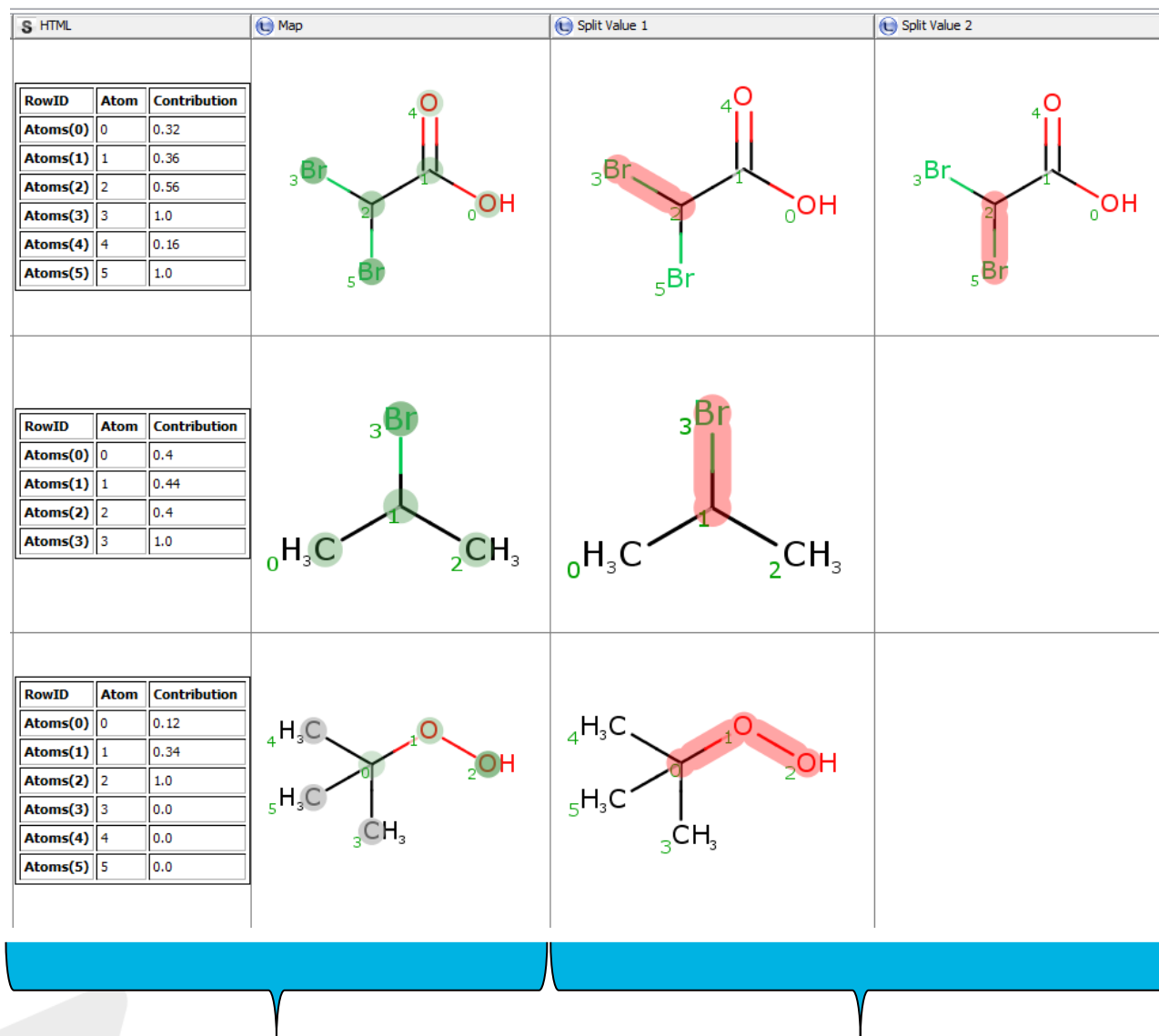
RowID	Atom	Contribution
Atoms(0)	0	0.5714285714285711
Atoms(1)	1	0.857142857142857
Atoms(2)	2	0.4285714285714281
Atoms(3)	3	0.5714285714285711
Atoms(4)	4	0.0
Atoms(5)	5	0.857142857142857
Atoms(6)	6	1.0
Atoms(7)	7	0.0
Atoms(8)	8	1.0
Atoms(6)	6	-0.727272727272727

# Model interpretation

- Implemented the Feature combination networks interpretation method (easy to do as we developed it)
- <http://www.jcheminf.com/content/6/1/8>
- Summary:
  - Elucidate the models behaviour based on fragments not individual atoms



# Model interpretation

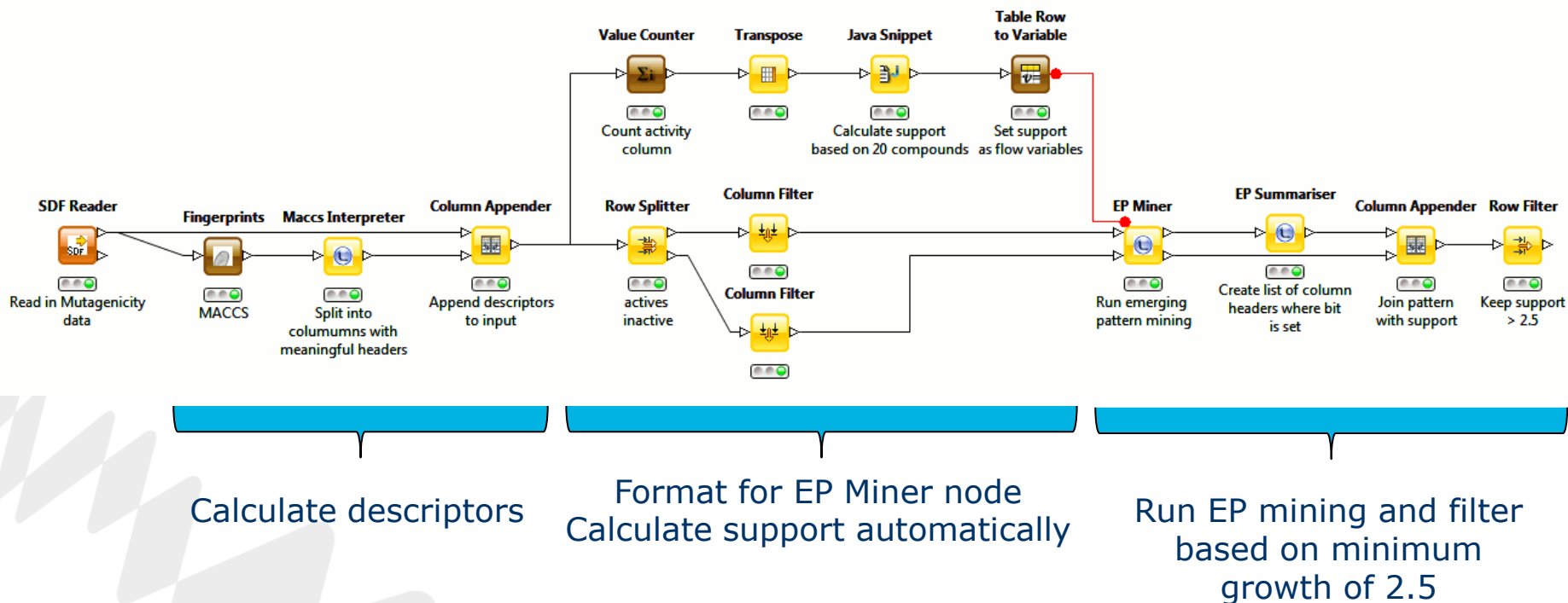


Similarity maps

Feature combination networks

# Emerging Pattern mining

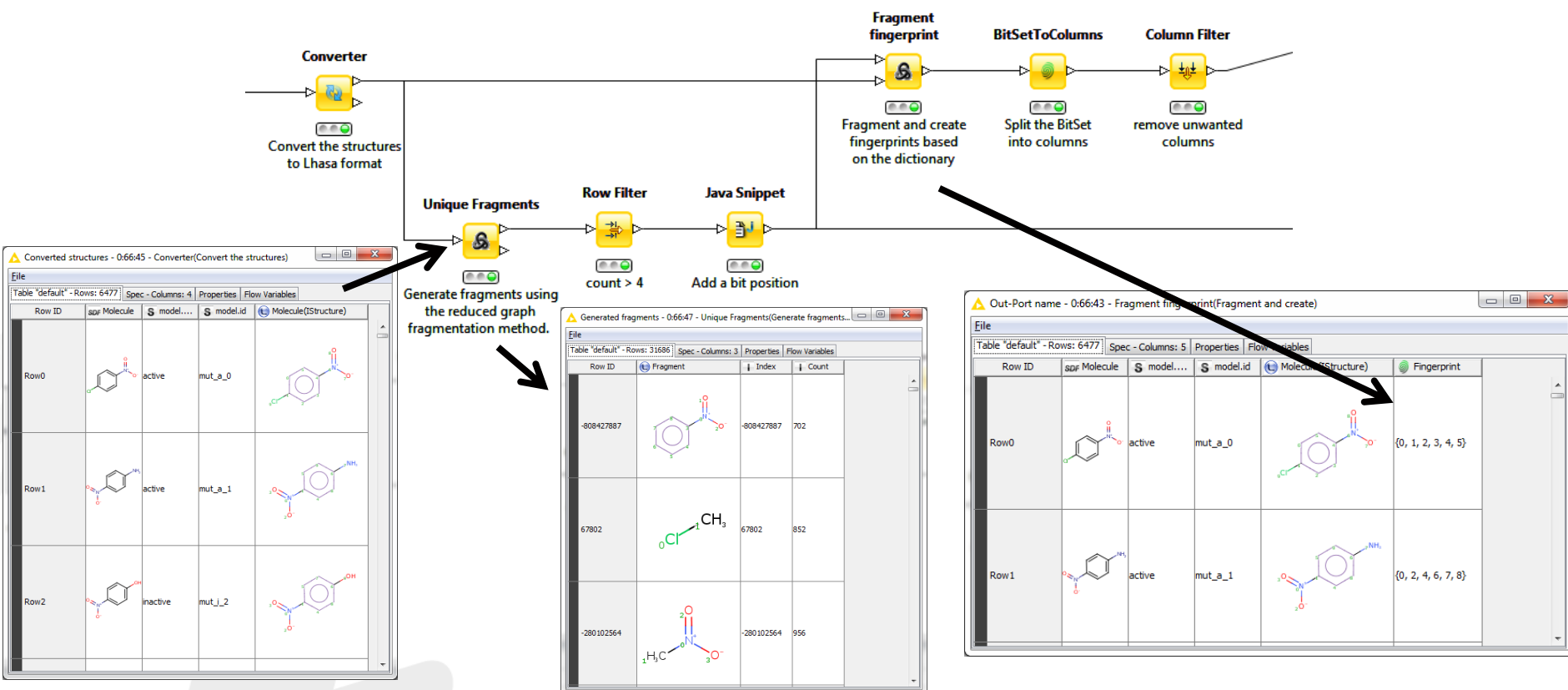
- Emerging Pattern mining algorithm has been implemented in a couple of nodes (R. Sherhod ~ now at Vernalis)
- <http://pubs.acs.org/doi/abs/10.1021/ci5001828>





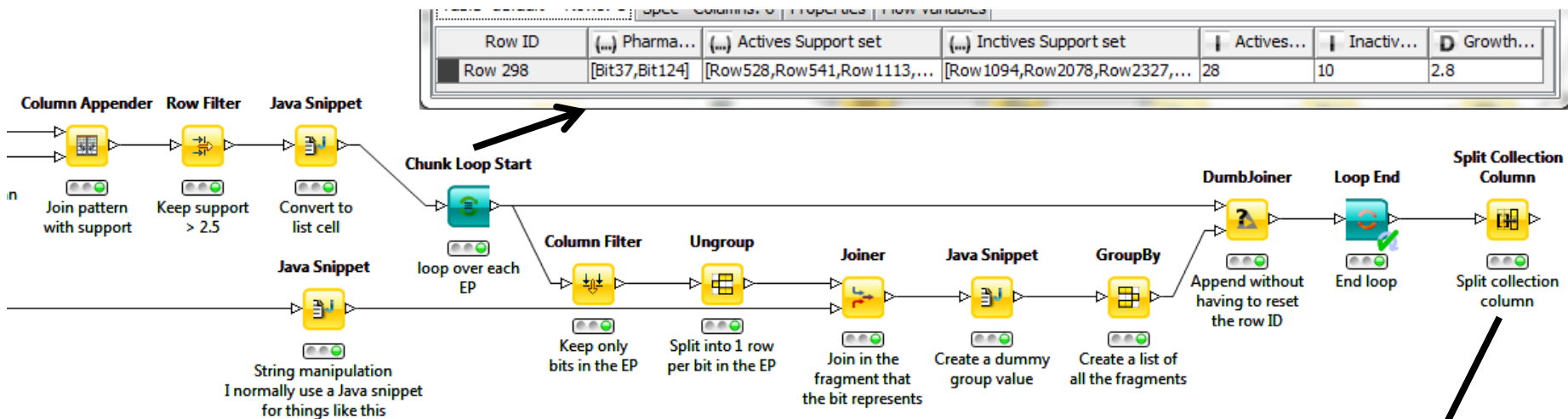
# Emerging Pattern mining

- Fragment dictionary approach
- Fragment the dataset using a fragmentation approach
  - We use the reduced graph approach developed in house



# Emerging Pattern mining: visualisation of EP

- This is where it becomes a bit tricky in KNIME



Appended table - 0:67 - Java Snippet(String manipulation)

Row ID	Fragment	Index	Count	bitPosition	Column...
-808427887	<chem>CC(=O)Nc1ccc(N)cc1</chem>	-808427887	702	0	Bit0
67802	<chem>CC(=O)Nc1ccc(N)cc1</chem>	67802	852	1	Bit1

Joined table - 0:73 - Joiner(Join in ...)

Row ID	Fragment
Row 298_1_7...	<chem>CC(=O)Nc1ccc(N)cc1</chem>
Row 298_2_...	<chem>CC(=O)Nc1ccc(N)cc1</chem>

Pharma...	Split Value 1	Split Value 2	Split Value 3	Split Value 4
[Bit6, Bit115]	<chem>CC(=O)Nc1ccc(N)cc1</chem>	<chem>CC(=O)Nc1ccc(N)cc1</chem>		
[Bit4, Bit6, Bit...	<chem>CC(=O)Nc1ccc(N)cc1</chem>	<chem>CC(=O)Nc1ccc(N)cc1</chem>	<chem>CC(=O)Nc1ccc(N)cc1</chem>	
[Bit1, Bit4, Bit...	<chem>CC(=O)Nc1ccc(N)cc1</chem>	<chem>CC(=O)Nc1ccc(N)cc1</chem>	<chem>CC(=O)Nc1ccc(N)cc1</chem>	<chem>CC(=O)Nc1ccc(N)cc1</chem>

# Emerging Pattern mining: visualisation of EP


- 23 structures contain Bit357
- Bit357 is a acid chloride motif
- We have two lists of RowID's: one for active support and one for inactive support
- We could make a report
  - Can this be automated? There's over 100 EP's!

## Emerging pattern report

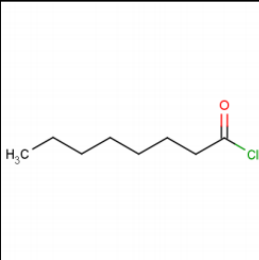
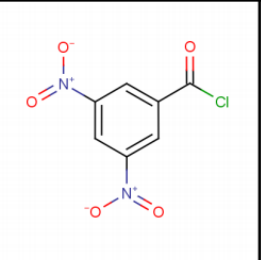
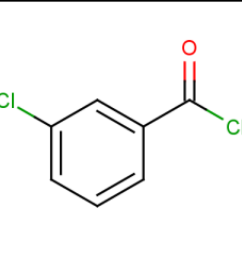
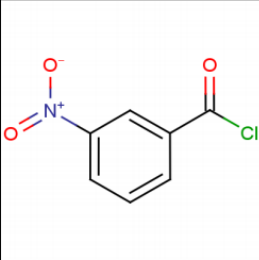
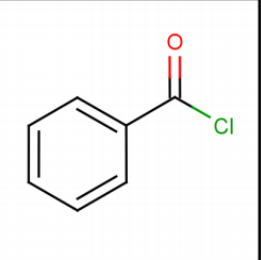
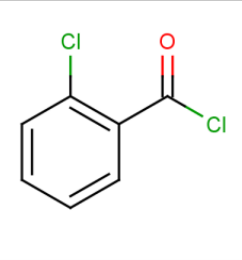
Emerging pattern mining completed in KNIME using a fragmentation dictionary.

The dictionary was built using a min depth of 0, max depth of 2, rings and functions were kept.

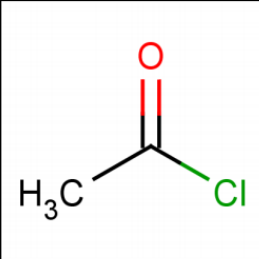
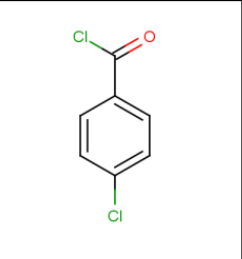
## Emerging pattern details

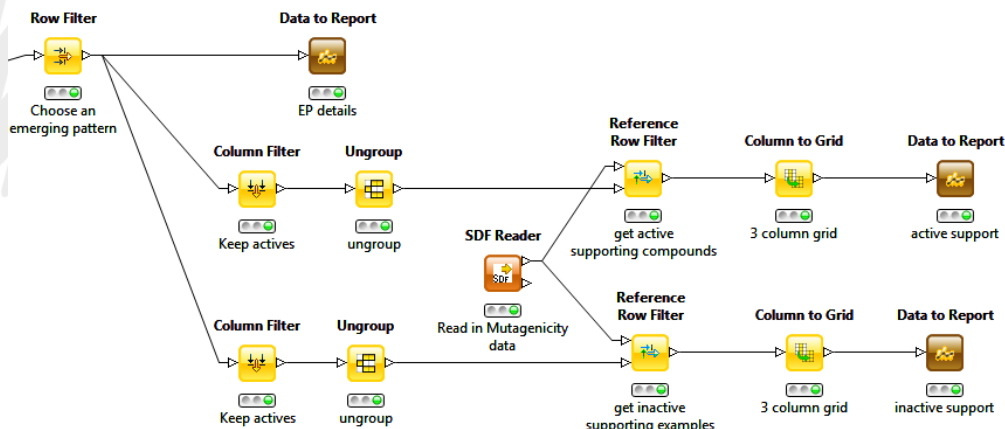
Actives count	Inactives count	Growth rate	Fragment 1	Fragment 2	Fragment 3	Fragment 4
21	2	10.5		The resource of this report item is not reachable.	The resource of this report item is not reachable.	The resource of this report item is not reachable.

## Active support

## Inactive support

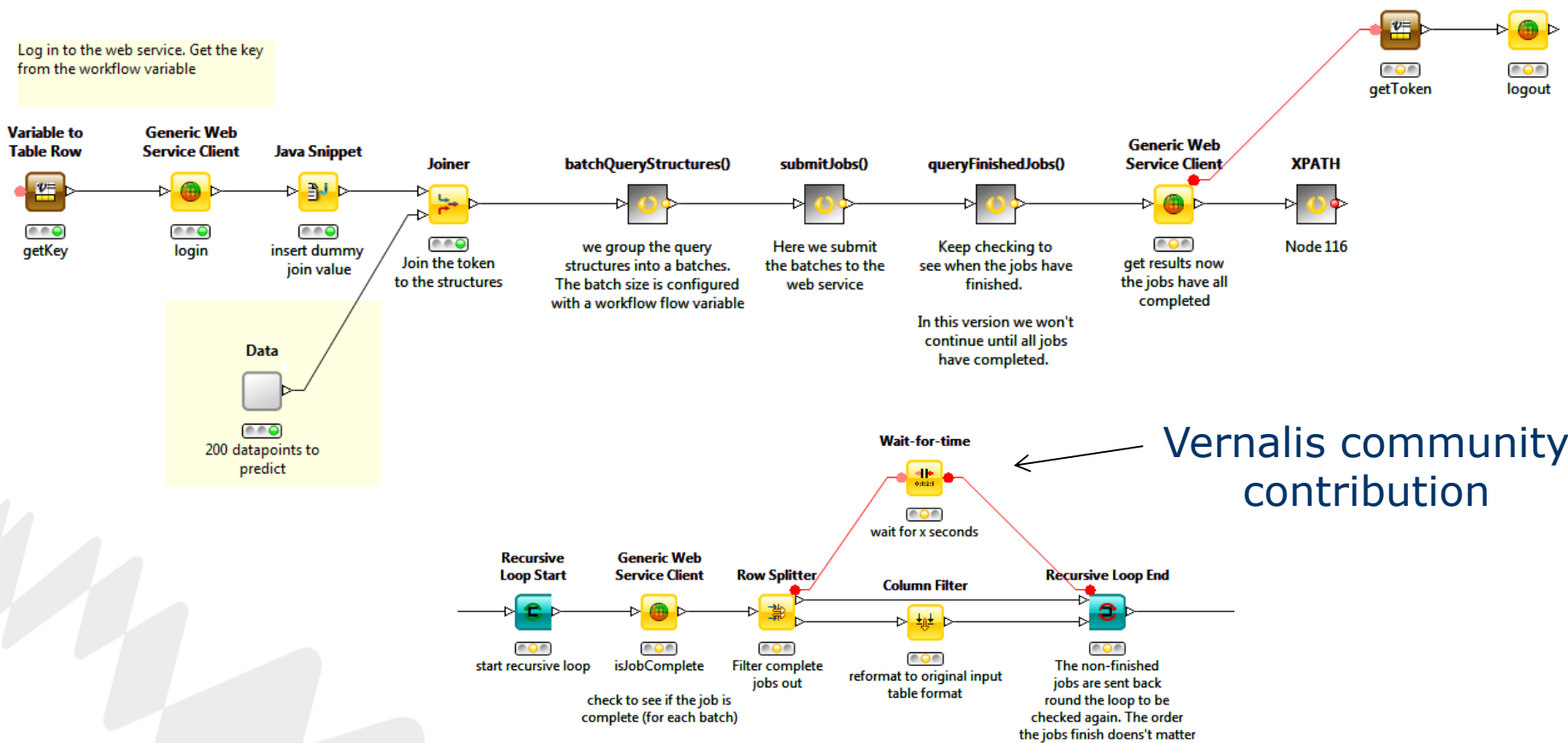
	The resource of this report item is not reachable.	
---	--	---



# Derek web service

- Web service (SOAP) available for the expert system for toxicity prediction Derek.

- <http://www.lhasalimited.org/products/derek-nexus.htm>





# Acknowledgements

---

- KNIME
- Community contributors
- Forum users
- All my colleagues who shared their use cases



# Thank you



shared **knowledge** • shared **progress**

Lhasa Limited Registered Office  
Granary Wharf House, 2 Canal Wharf, Leeds LS11 5PS  
UK Registered Charity (290866)

+44 (0)113 394 6020  
[info@lhasalimited.org](mailto:info@lhasalimited.org)  
[www.lhasalimited.org](http://www.lhasalimited.org)

Company Registration Number 01765239. Registered in England and Wales. VAT Registration Number GB 396 8737 77.



ISO 9001: 2008 CERTIFIED