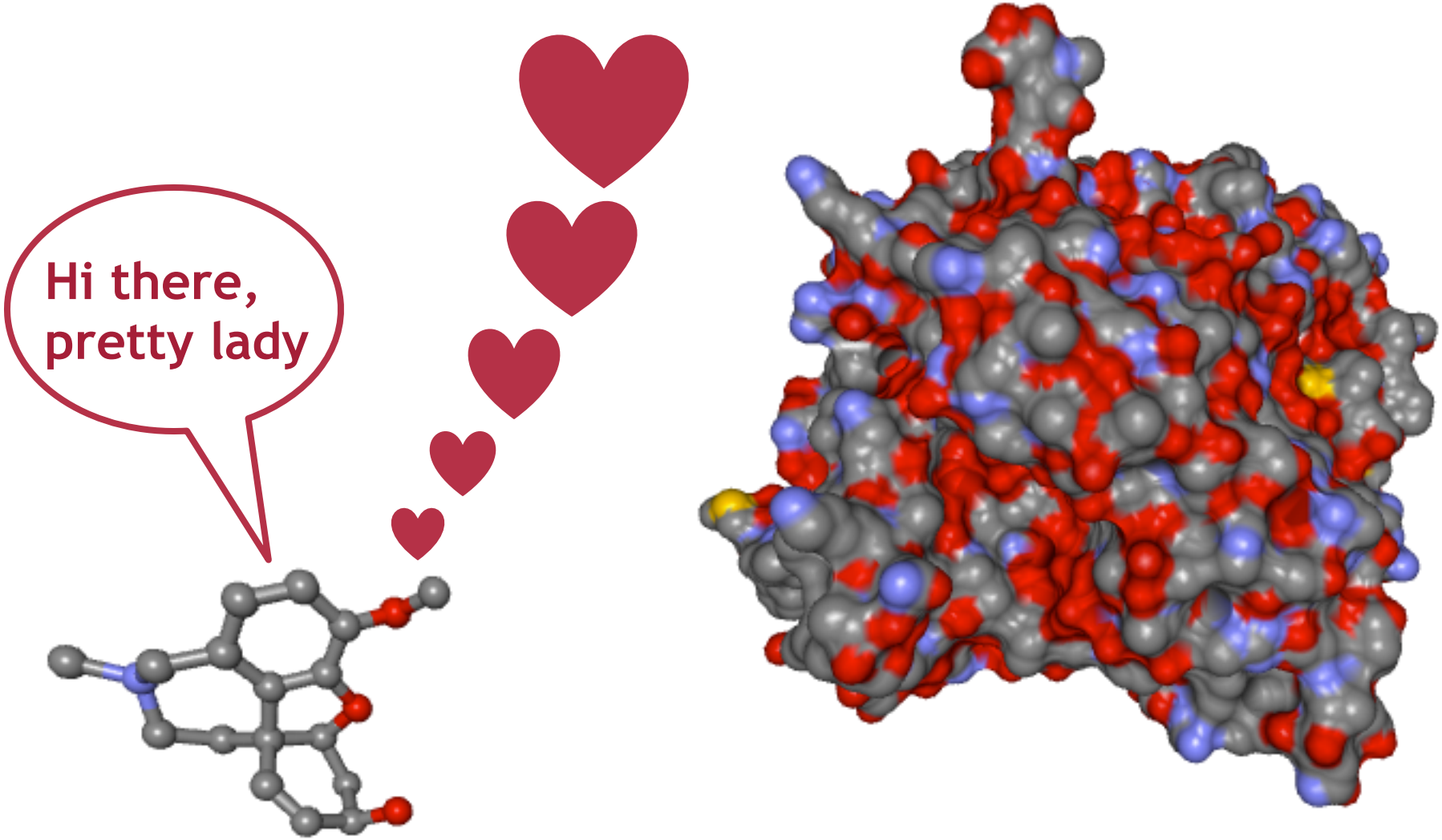
From the Desktop to the Grid: Conversion of KNIME Workflows to gUSE

Luis de la Garza, Jens Krüger, Charlotta Schärfe, Marc Röttig, Stephan Aiche, Knut Reinert, Oliver Kohlbacher

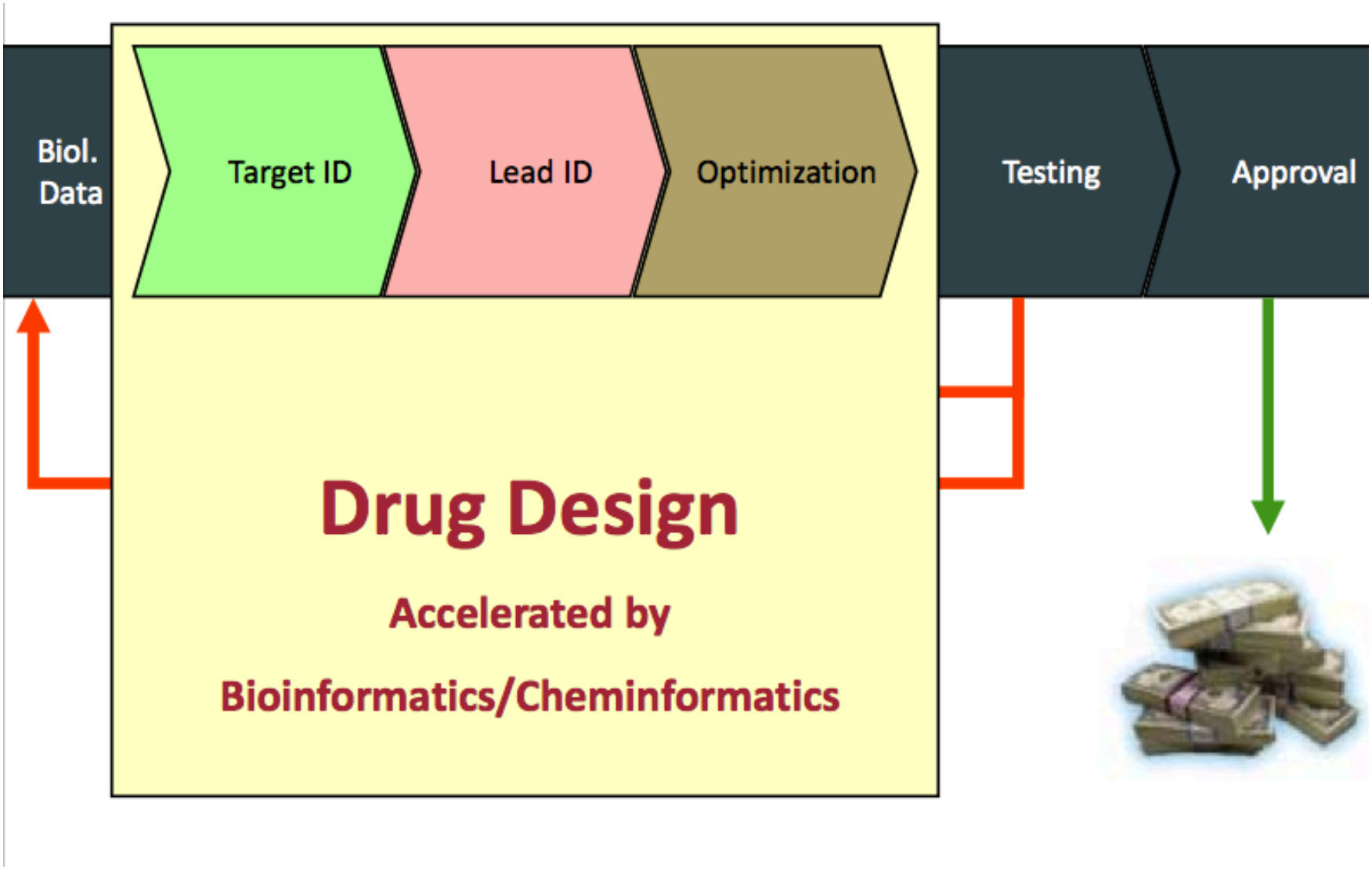
Department of Applied Bioinformatics
University of Tübingen

EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

What are we researching?

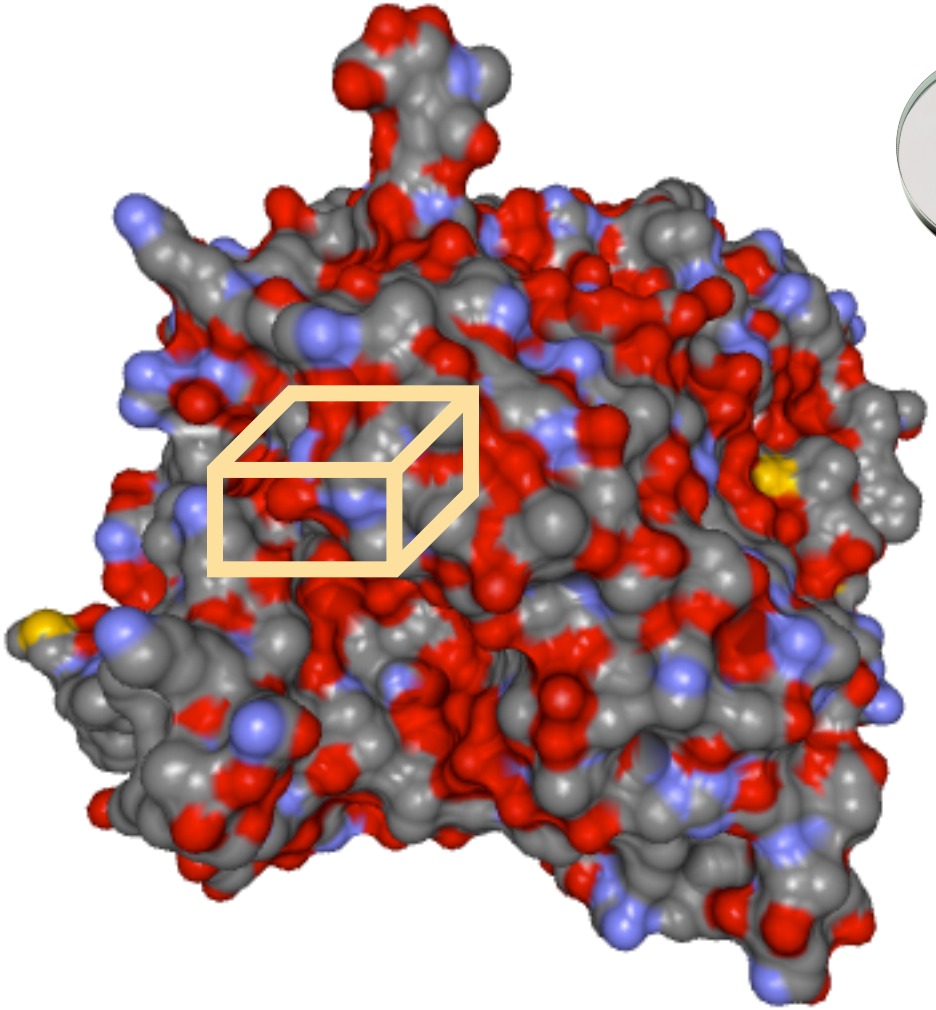


Why is Docking important?

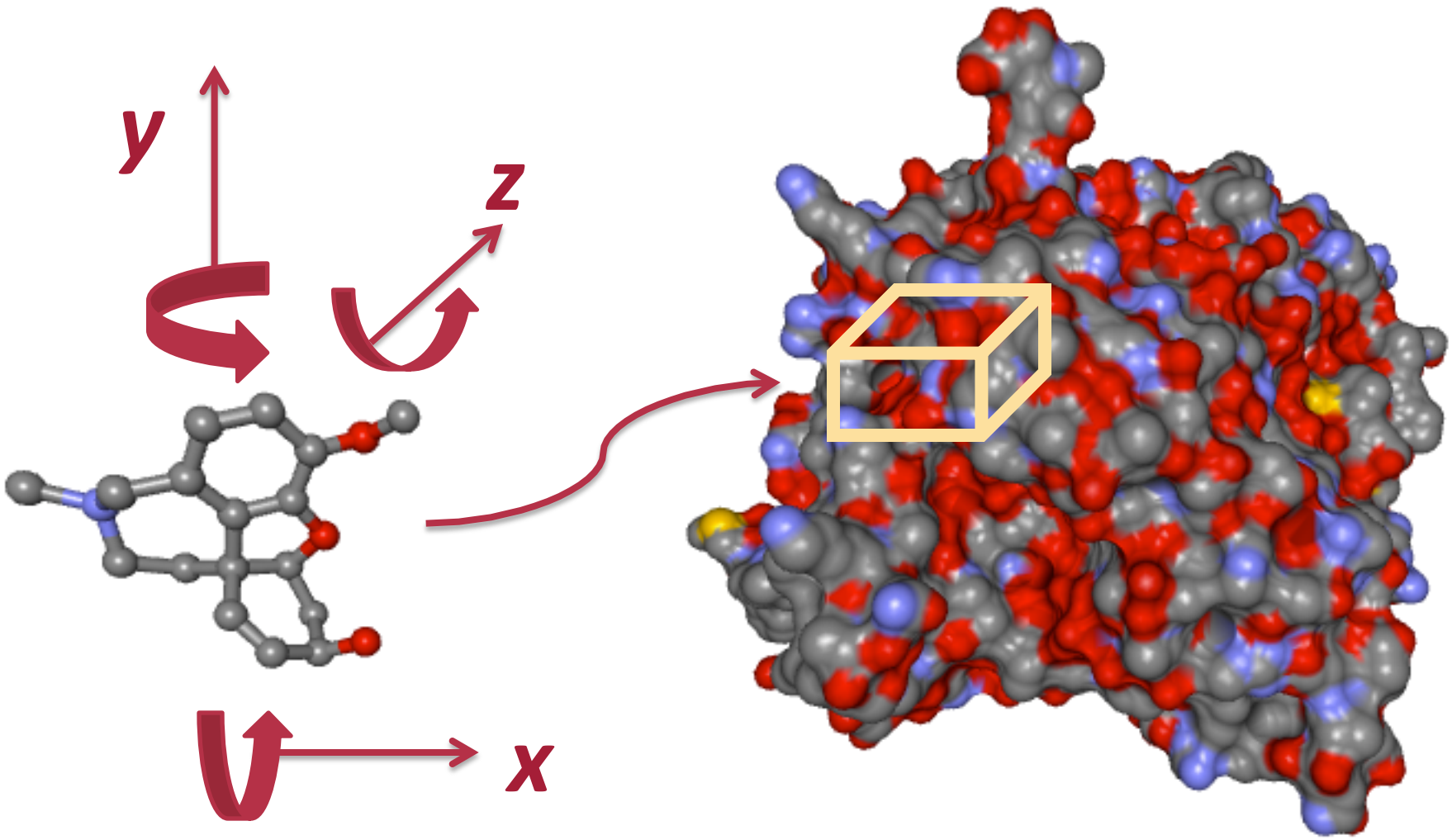


Oliver Kohlbacher, Drug Design I, 2010

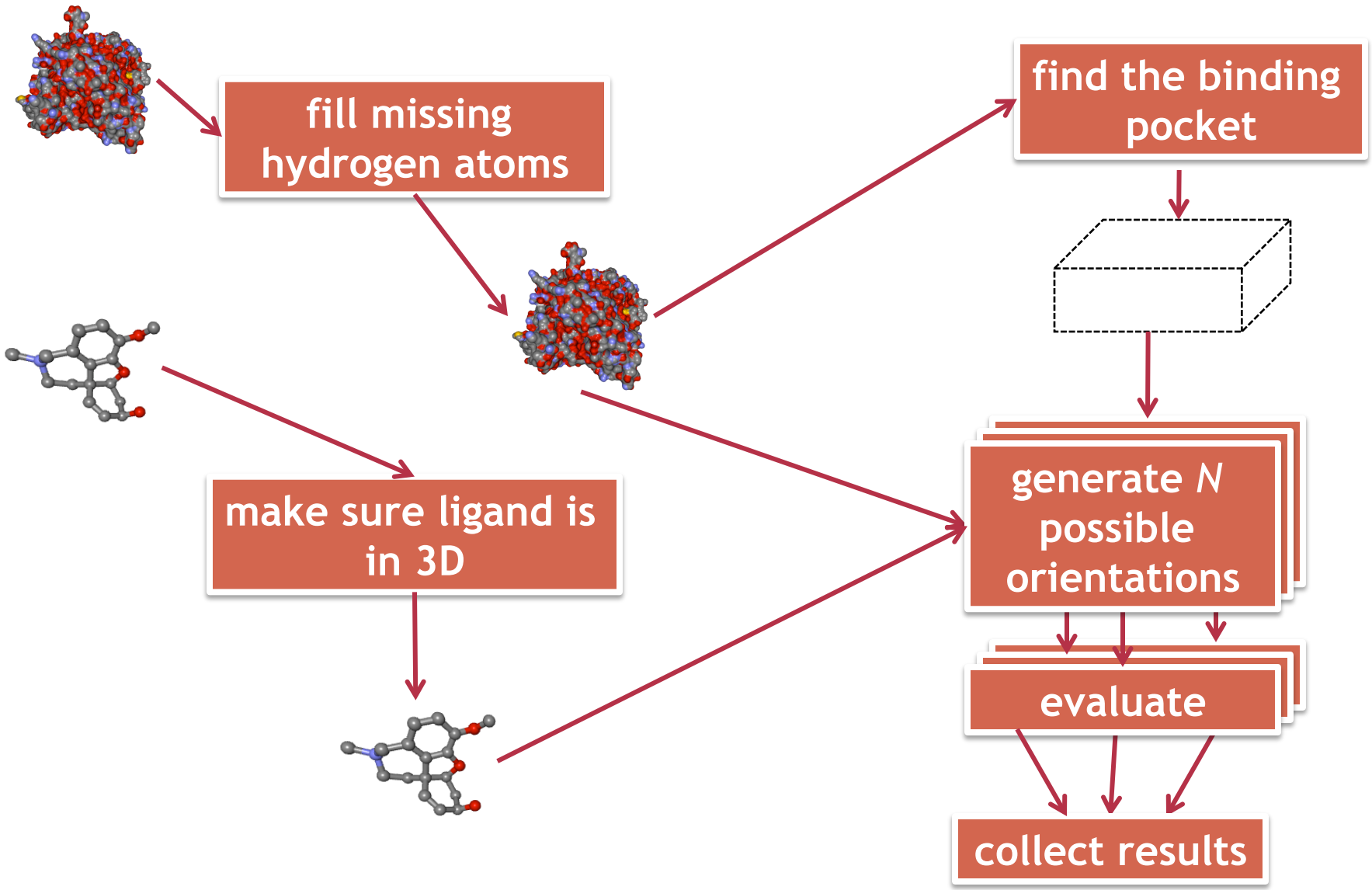
A Docking Recipe



Evaluation Orientations



It's begging to be a Workflow



Why do we use Workflows?

EBERHARD KARLS UNIVERSITÄT TÜBINGEN

- Docking can be broken down as a series of small tasks; some of them can be executed in parallel
- We need access to resources offering High Performance Computing (HPC)
- We want to store intermediate results for further analysis (*i.e.*, binding pocket computation)
- We need flexibility – we would like to test different docking algorithms without making lots of changes

Workflows everywhere

EBERHARD KARLS UNIVERSITÄT TÜBINGEN

- Workflows have applications in other areas of bioinformatics and in other fields, such as:
  - Data mining
  - Business process automation
  - Customer relationship management
  - Business intelligence

- If you can split a process in small automated tasks, then it can be put into a workflow

Workflow Managers

- Several needs have produced different workflow managers

Kepler

Galaxy

gUSE

Taverna

KNIME

Konstanz Information Miner

KNIME Desktop

EBERHARD KARLS UNIVERSITÄT TÜBINGEN

KNIME

KNIME Explorer | Favorite Nodes

- Personal favorite nodes
- Most frequently used nodes
- Last used nodes

Node Repository

- IO
- Database
- Data Manipulation
- Data Views
- Statistics
- Mining
- Chemistry
- ChemAxon / Infocom
- Distance Matrix
- Meta
- Flow Control
- Misc
- KNIME Labs
- Time Series
- Quick Form
- R
- Reporting
- Testing
- Weka
- XML

New

Select a wizard

Wizards:

type filter text

- New KNIME Workflow
- New KNIME Workflow Group
- General
- Business Intelligence and Reporting Tools
- Connection Profiles
- Java
- KNIME
- Plug-in Development
- User Assistance

Show All Wizards.

< Back | Next > | Cancel | Finish

Node Description

Interactive Table

Displays data in a table view. If the number of rows is unknown, the view counts the number of rows when opened. Furthermore, rows can be selected and hilited.

Ports

Input Ports

0 Input table to display.

Views

Table View

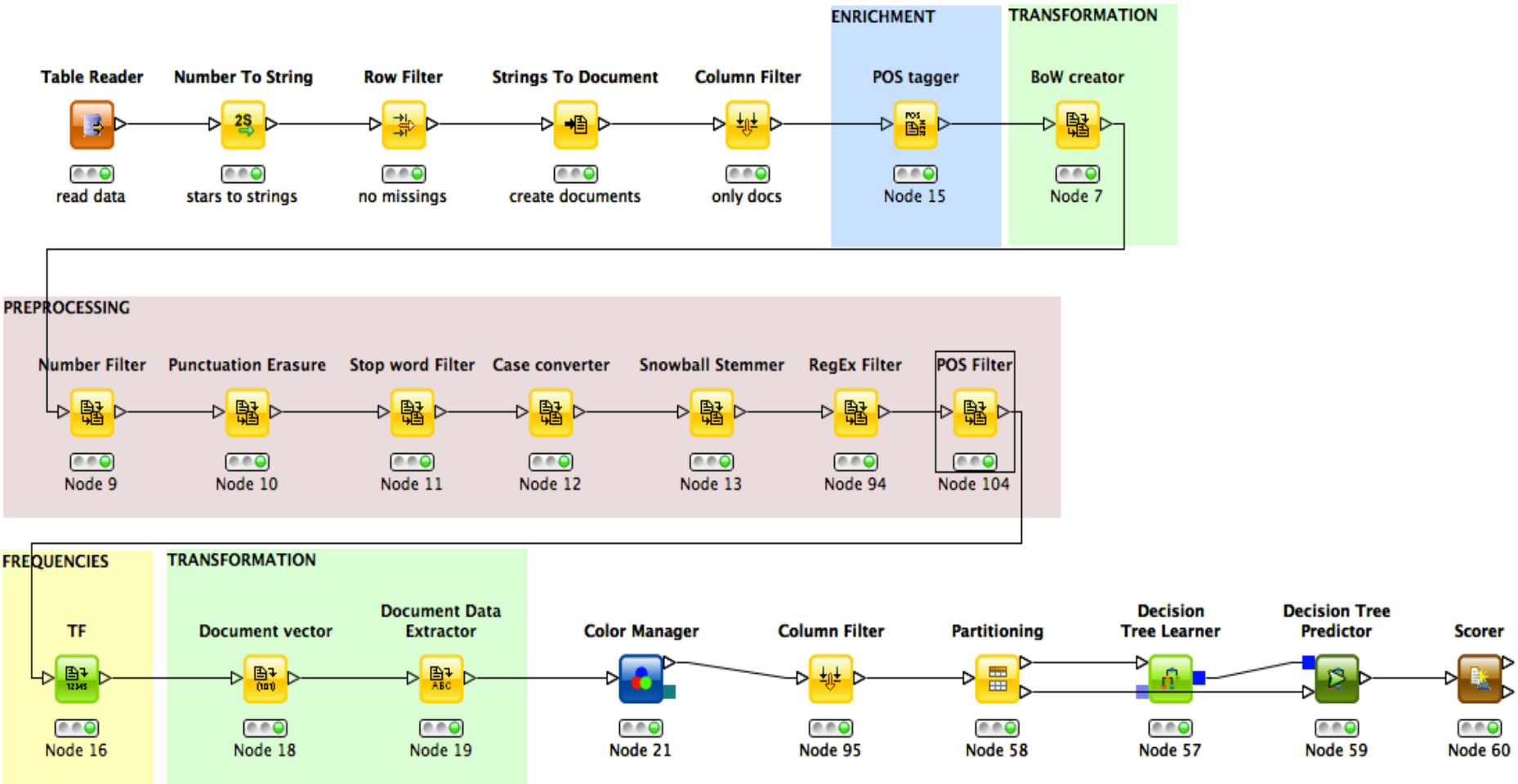
Displays the data in a table view.

This node is contained in KNIME Base Nodes provided by KNIME GmbH, Konstanz, Germany.

Console | Progress

No operations to display at this time.

A more elaborated Workflow



Workflow created during training at the KNIME User Meeting 2013, Zurich

KNIME Desktop

- Create and execute workflows on your desktop computer; free, as in "*free beer*", open source, available under the GPL license

- Easy-to-use interface (if you know how to *click*, you know how to KNIME)

- Already one of the most commonly used workflow management systems in the field of e-Science systems

KNIME Desktop

- Lots of KNIME Community Nodes available (*e.g.* R, NGS, Schrödinger, etc.)

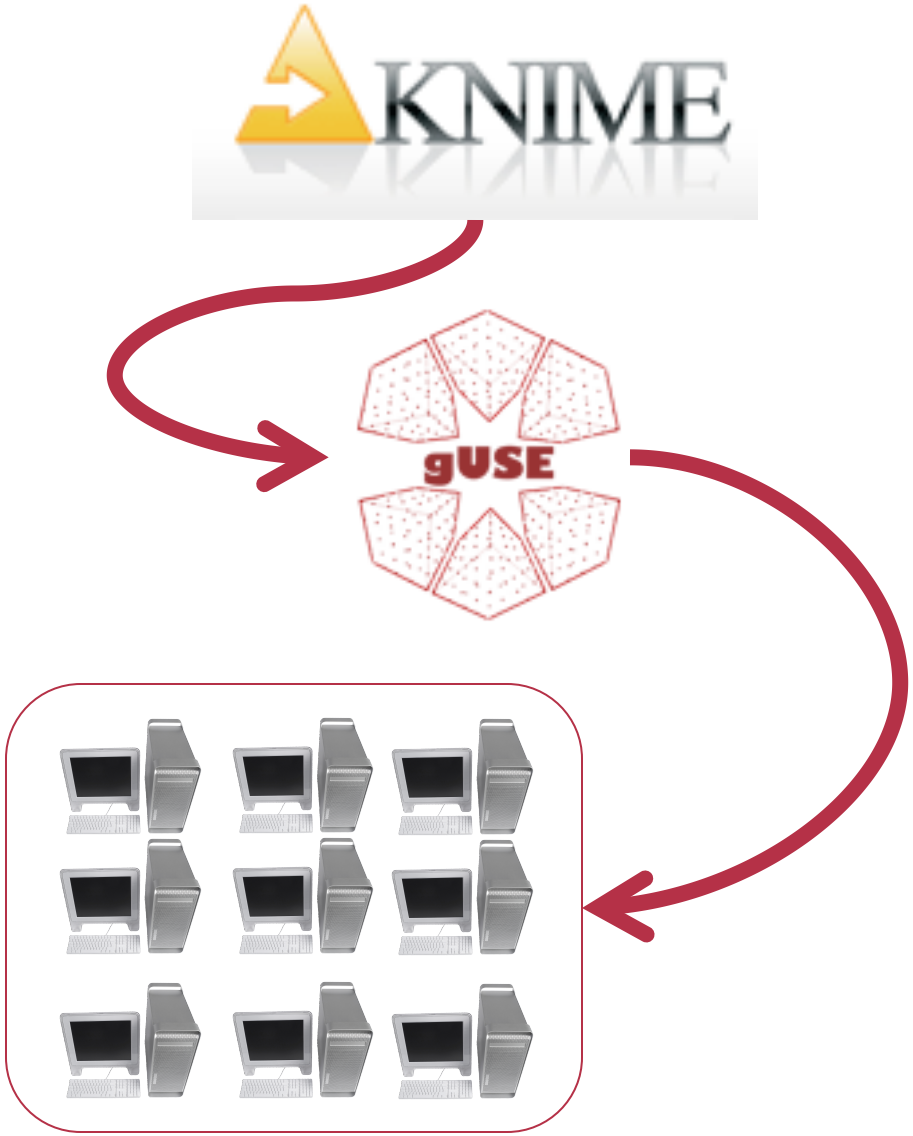
- Users can easily extend KNIME by creating new nodes (Java API) and extensions (based on Eclipse)

KNIME Cluster Execution

- Offers the capability to run KNIME Workflows on a cluster
- Works together with KNIME Server and KNIME Desktop to offer all features
- Not free, as in *"VISA or MasterCard?"*
- Needs to be installed as a cluster resource

Choices, choices...

- We want to use KNIME as a workflow editor
- We want access to computing resources
- We want gUSE to execute our workflows



KNIME - External Tool Integration

EBERHARD KARLS UNIVERSITÄT TÜBINGEN

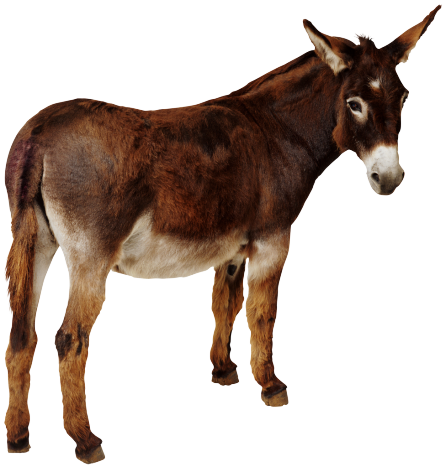
- Wouldn’t it be great if arbitrary external command line tools could be integrated into KNIME?

KNIME - External Tool Integration

EBERHARD KARLS UNIVERSITÄT TÜBINGEN

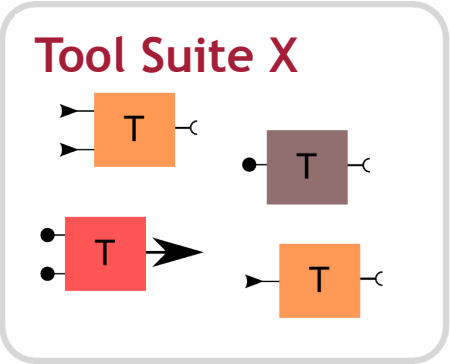
- KNIME is based on data tables, while most command line tools are based on files

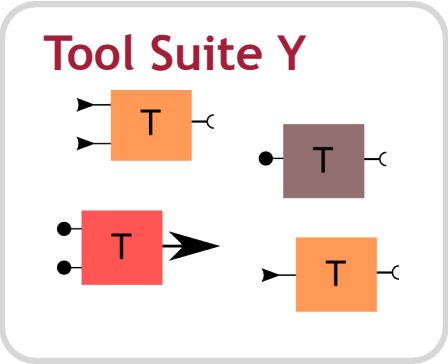
- It would be possible, yet time consuming and error prone, to manually generate a KNIME node for each tool to be integrated



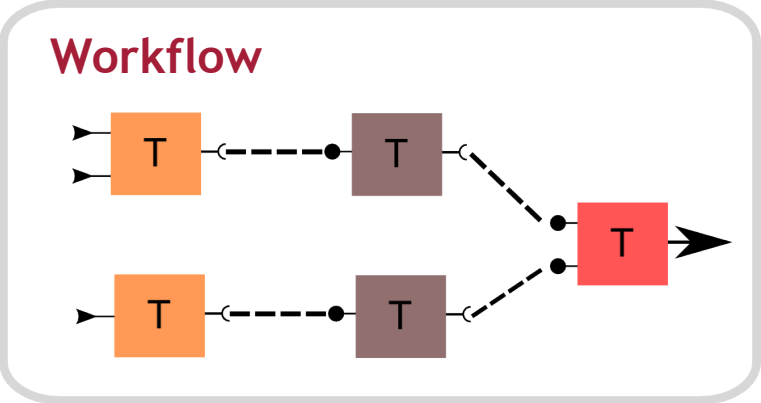
Tool Integration

- We have developed several tool suites, each containing several tools





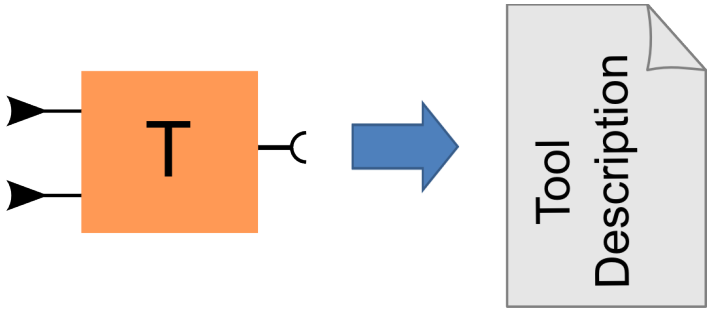
- Each tool should be usable as a single job in a workflow



Tool Integration – our Approach

EBERHARD KARLS UNIVERSITÄT TÜBINGEN

- Give every tool a self-describing output format: semantic annotation of inputs, outputs and parameters



- Common Tool Description (CTD) was initially developed by the Open-Source Framework for Mass Spectrometry (OpenMS) team

Common Tool Description

- Each tool can “tell” its requirements and options, thus allowing easy integration and a more robust approach than “classical” tool stubs (*e.g.*, scripting)

- XML Documents – easy to understand, parse and generate

Common Tool Description

```
<!-- pdbcutter_execution.xml -->
<tool status="internal">
  <name>PDBCutter</name>
  <PARAMETERS version="1.3">
    <NODE>
     <ITEM name="i" tags="input file" value="input.pdb"/>
     <ITEM name="rec" tags="output file" value="receptor.pdb"/>
     <ITEM name="lig" tags="output file" value="ligand.pdb"/>
     <ITEM name="lig_chain" tags="required" value="A"/>
     <ITEM name="lig_name" tags="required" value="GNT"/>
    </NODE>
  </PARAMETERS>
</tool>

# using PDBCutter with a CTD as input
$ PDBCutter –par pdbcutter_execution.xml

# classic invocation
$ PDBCutter –i input.pdb –rec receptor.pdb –lig ligand.pdb
            –lig_name GNT –lig_chain A
```

Generic KNIME Nodes

- Leaning on CTDs, external tools can be integrated into KNIME

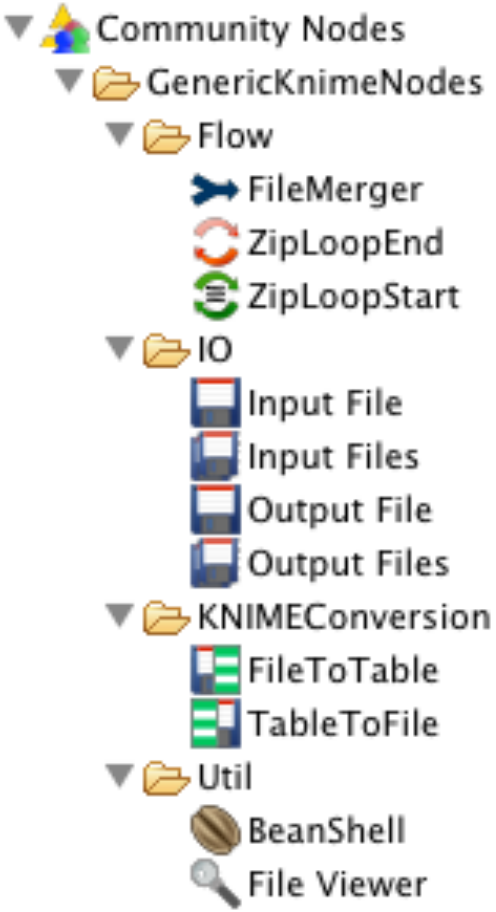
- Generic KNIME Nodes (GKN) are 100% compatible with other KNIME nodes

- Any tool described by a CTD can be used in KNIME

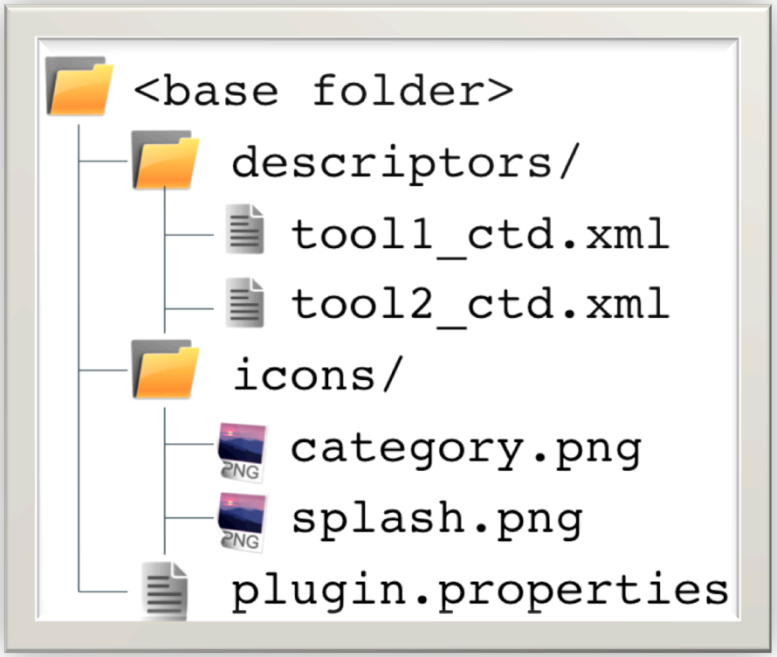
Generic KNIME Nodes

- Remember that KNIME relies on data tables and most command line tools rely on files?

GKN bridges this gap



Generic KNIME Nodes



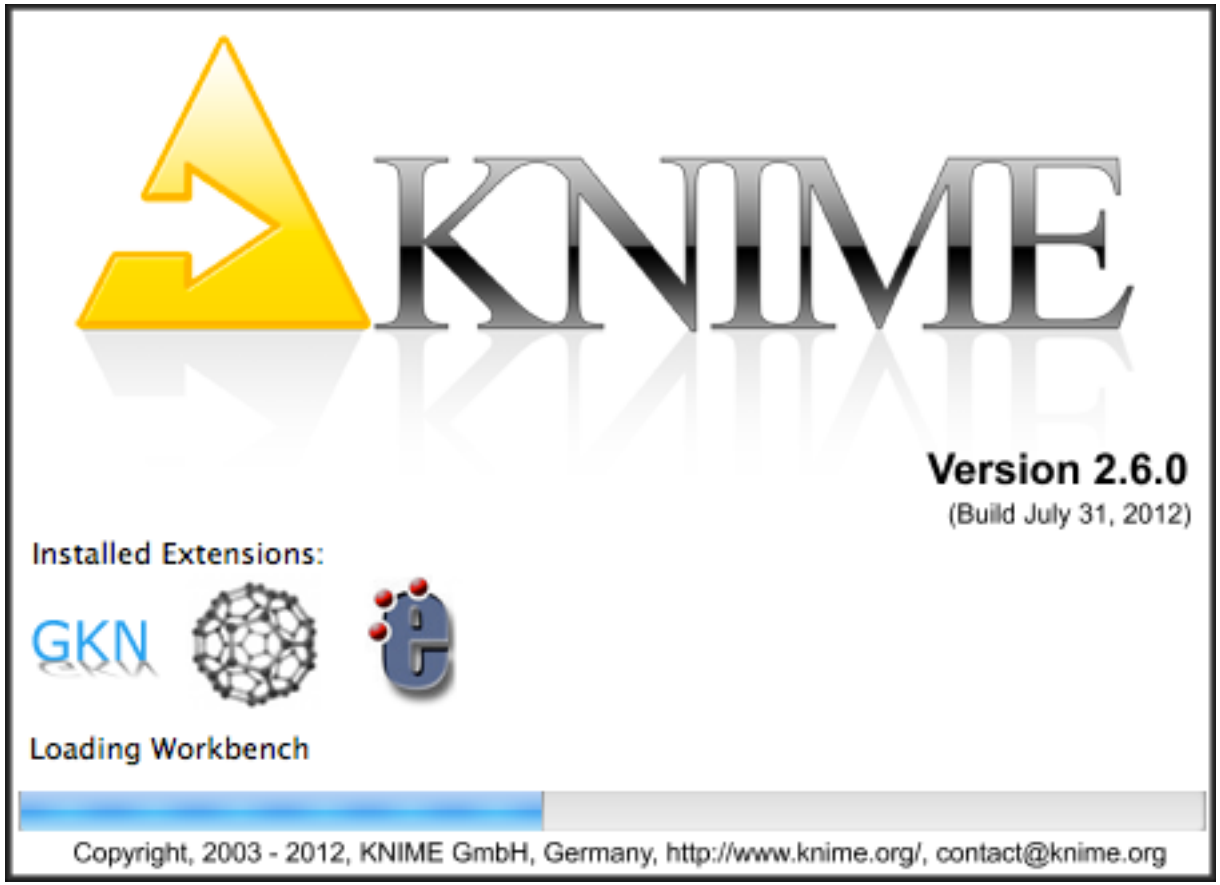
GKN
plugin generator

new_nodes.jar

install as KNIME plugin

KNIME

Generic KNIME Nodes



Computer Aided Drug Design Suite (CADDSuite)

emboss

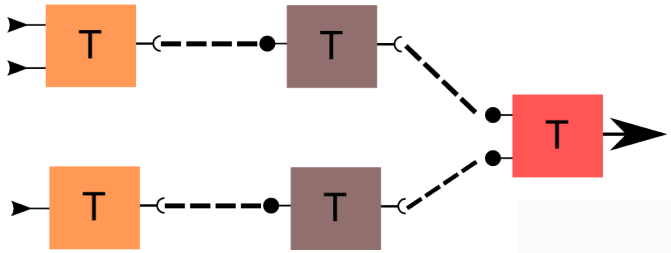
The European Molecular Biology Open Software Suite (EMBOSS)

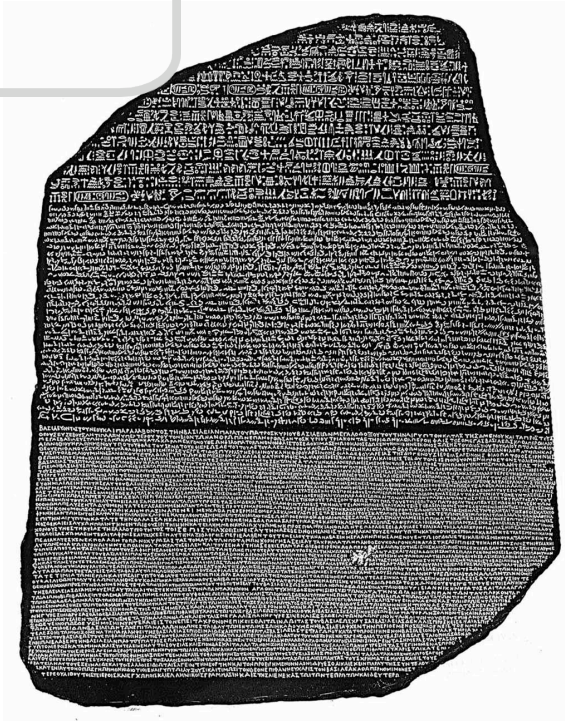
CADDSuite and EMBOSS running as extensions in KNIME

Rosetta Stone still missing

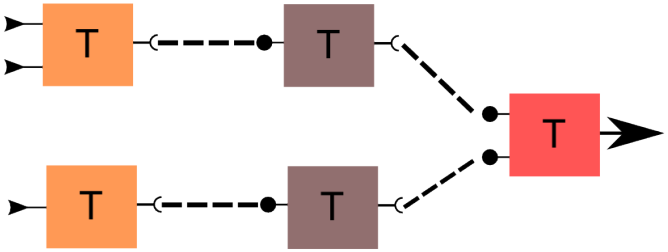
EBERHARD KARLS UNIVERSITÄT TÜBINGEN

gUSE Workflow





KNIME Workflow



http://liology.files.wordpress.com/2010/01/rosetta-stone.jpeg

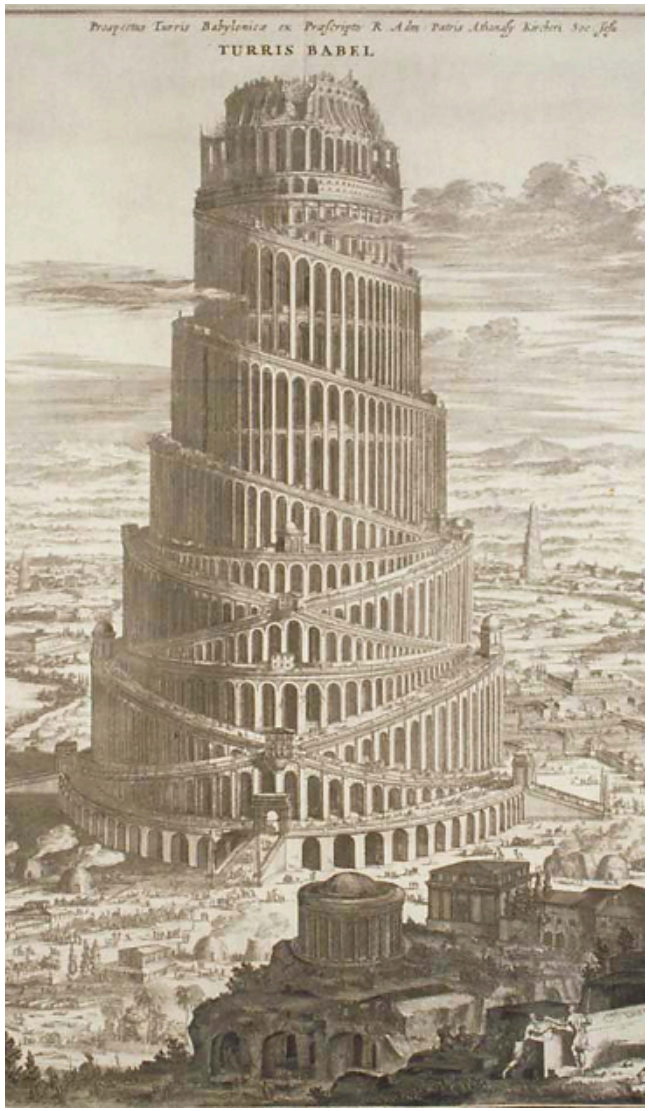
Workflow Interoperability

- In general, workflows are not compatible across different workflow managers

- It is not trivial to convert workflows between workflow managers

What's the Problem?

- Topology has to be converted
- Jobs have their own configuration settings and must be converted too
- Parameter sweep implementation might vary across managers
- Architecture on which jobs are executed might differ



http://cerasi.files.wordpress.com/2011/05/babel.jpg

SHIWA - Sharing Workflows

EBERHARD KARLS UNIVERSITÄT TÜBINGEN

- Sharing Interoperable Workflows for large-scale scientific Simulations on available Distributed Computing Interfaces (SHIWA)

- Allows workflows to be reused across different workflow languages

From KNIME to gUSE

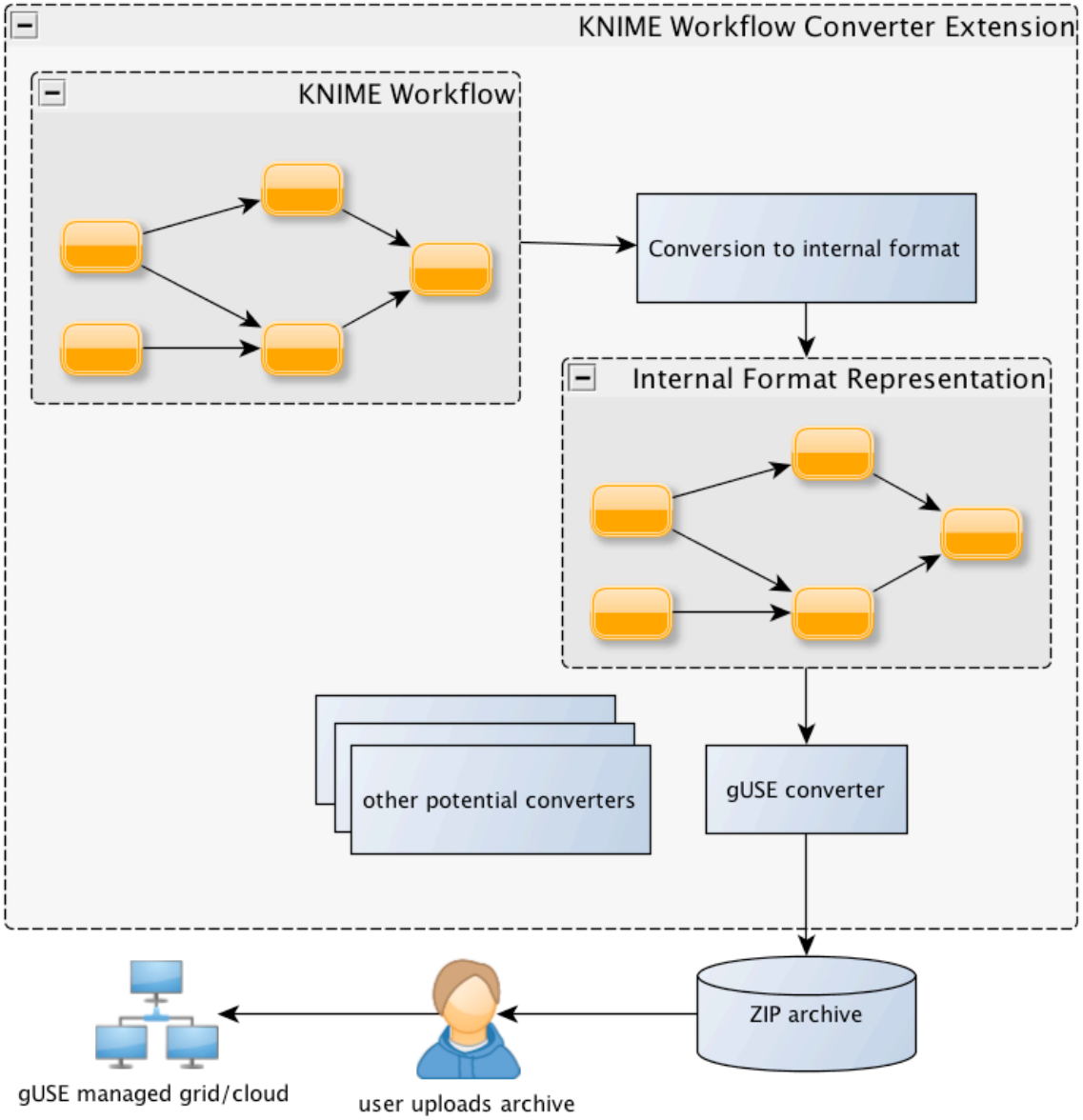
EBERHARD KARLS UNIVERSITÄT TÜBINGEN

- Creating and editing workflows in KNIME has a gentle learning curve

- However, desktop computers don't offer too much computing power for our needs

- **If only it were possible to upload a KNIME workflow to gUSE...**

From KNIME to gUSE



Internal Workflow Format

- KNIME has its own development cycles
- An internal workflow format is used in order to shield converters from any possible changes in KNIME's API
- Separation of concerns

Handling gUSE Middlewares

EBERHARD KARLS UNIVERSITÄT TÜBINGEN

- There is a disparity between the architecture of the desktop computer used to create the workflow and the architecture of the machines on the grid

- Different middlewares (*e.g.*, UNICORE, LSF) require different job configuration – solved using conversion tables

Handling of Generic KNIME Nodes

EBERHARD KARLS UNIVERSITÄT TÜBINGEN

- Nodes imported via GKN use CTDs, therefore, generating the appropriate command line for the gUSE format is not complicated

Handling of non-Generic KNIME Nodes

EBERHARD KARLS UNIVERSITÄT TÜBINGEN

- Non-Generic KNIME Nodes are not command line tools, they are Java objects inside KNIME

- There is a way to execute KNIME via command line – KNIME Desktop must be also installed on the grid

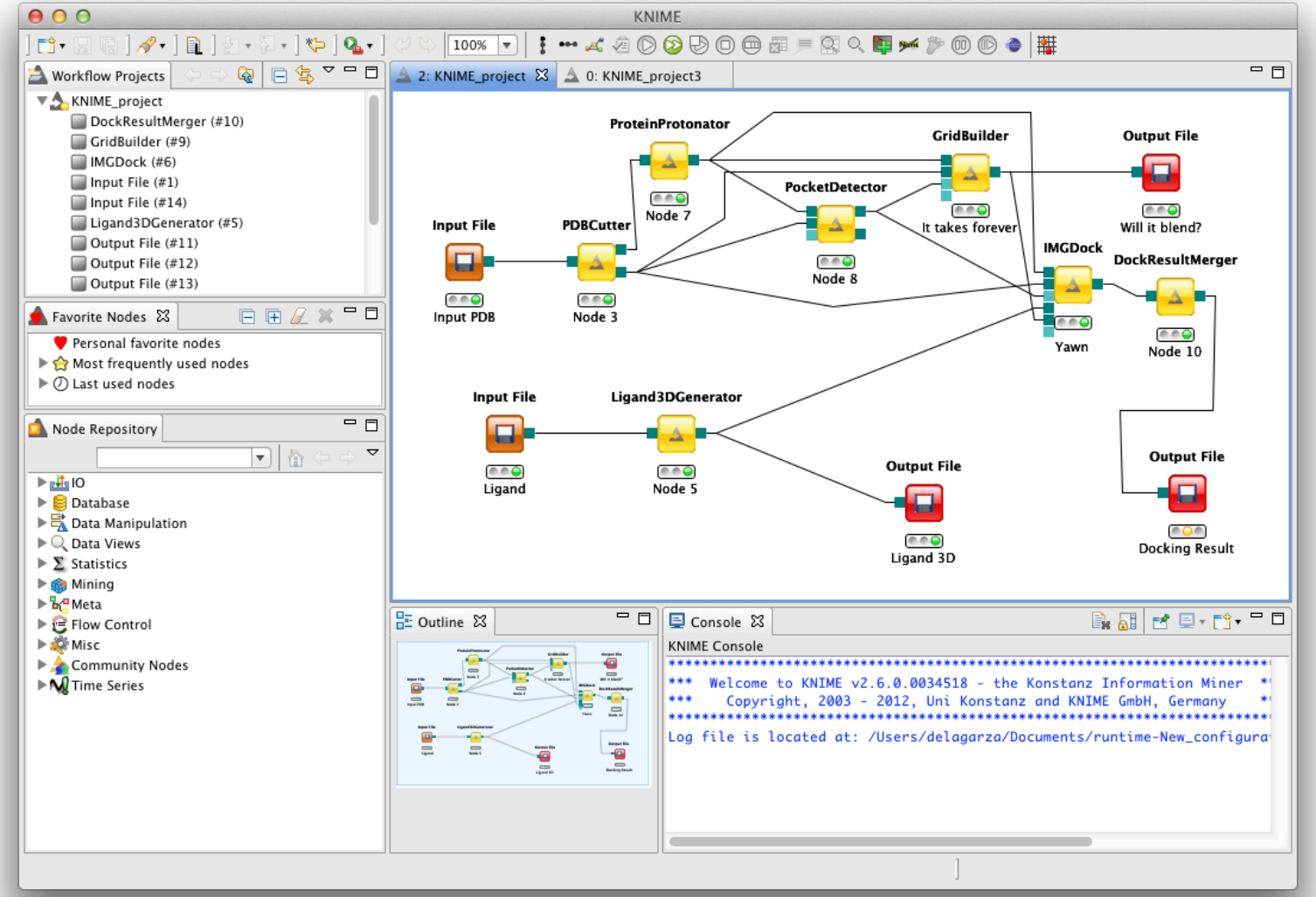
- Workflows containing both types of nodes must be split into homogeneous sub-workflows and handled accordingly*

*work in progress

Docking on KNIME, then on the Grid

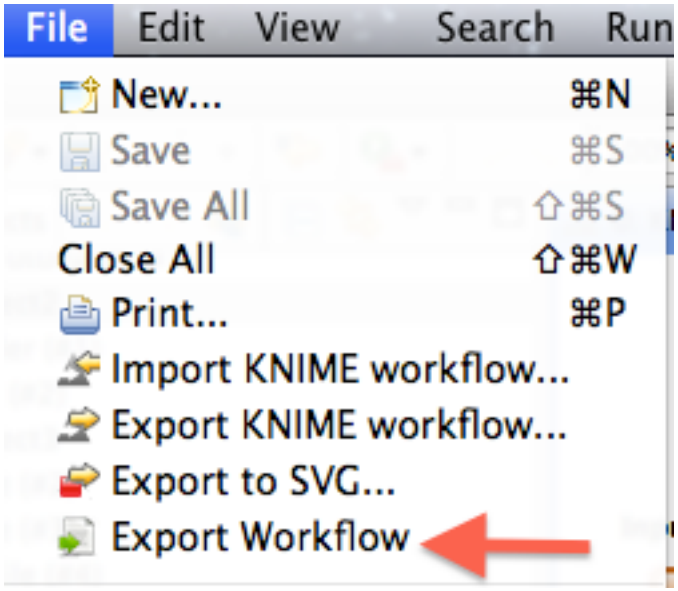
EBERHARD KARLS UNIVERSITÄT TÜBINGEN

- It all starts with a workflow in KNIME



Docking on KNIME, then on the Grid

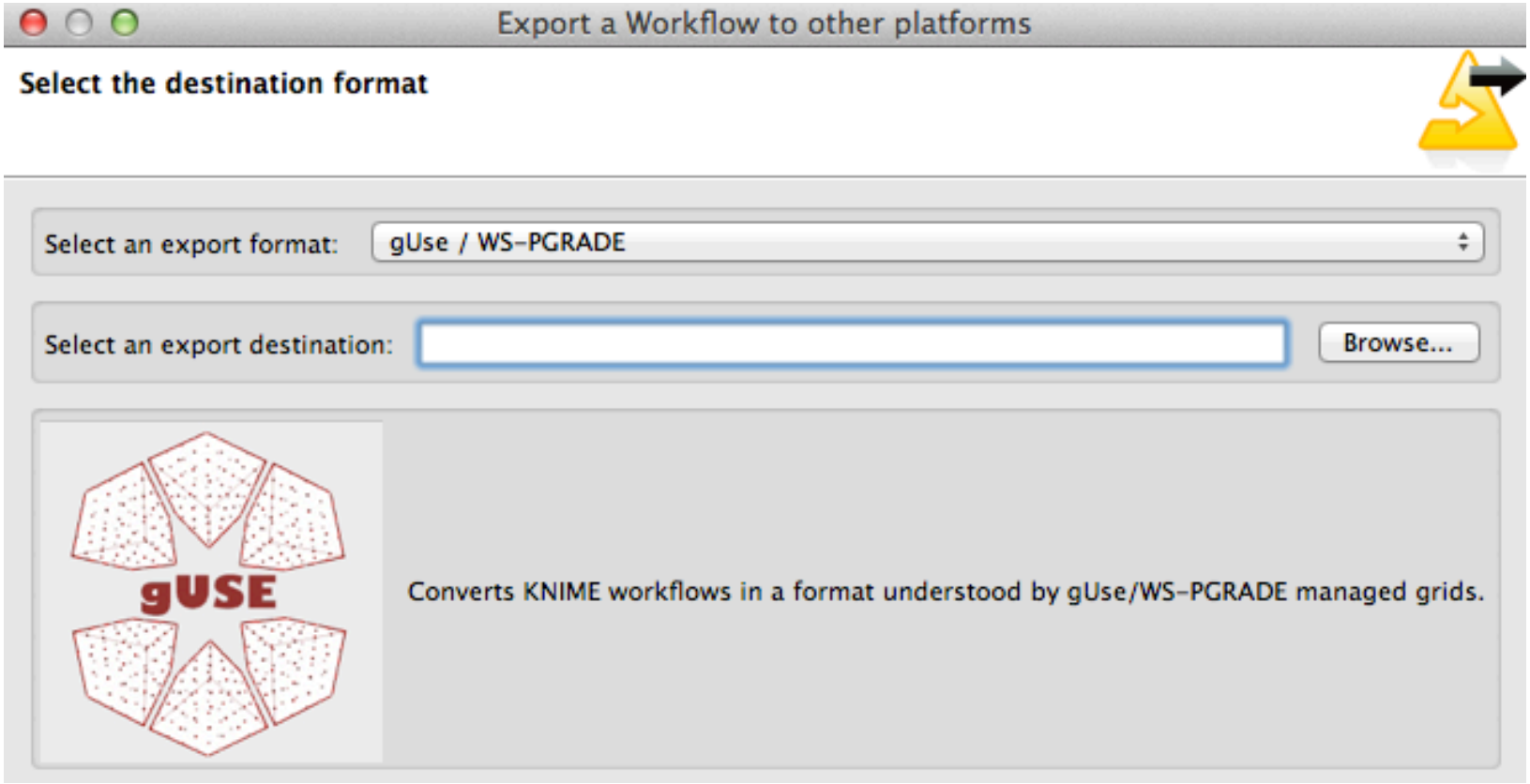
- The extension offers a simple user interface integrated into KNIME



Docking on KNIME, then on the Grid

EBERHARD KARLS UNIVERSITÄT TÜBINGEN

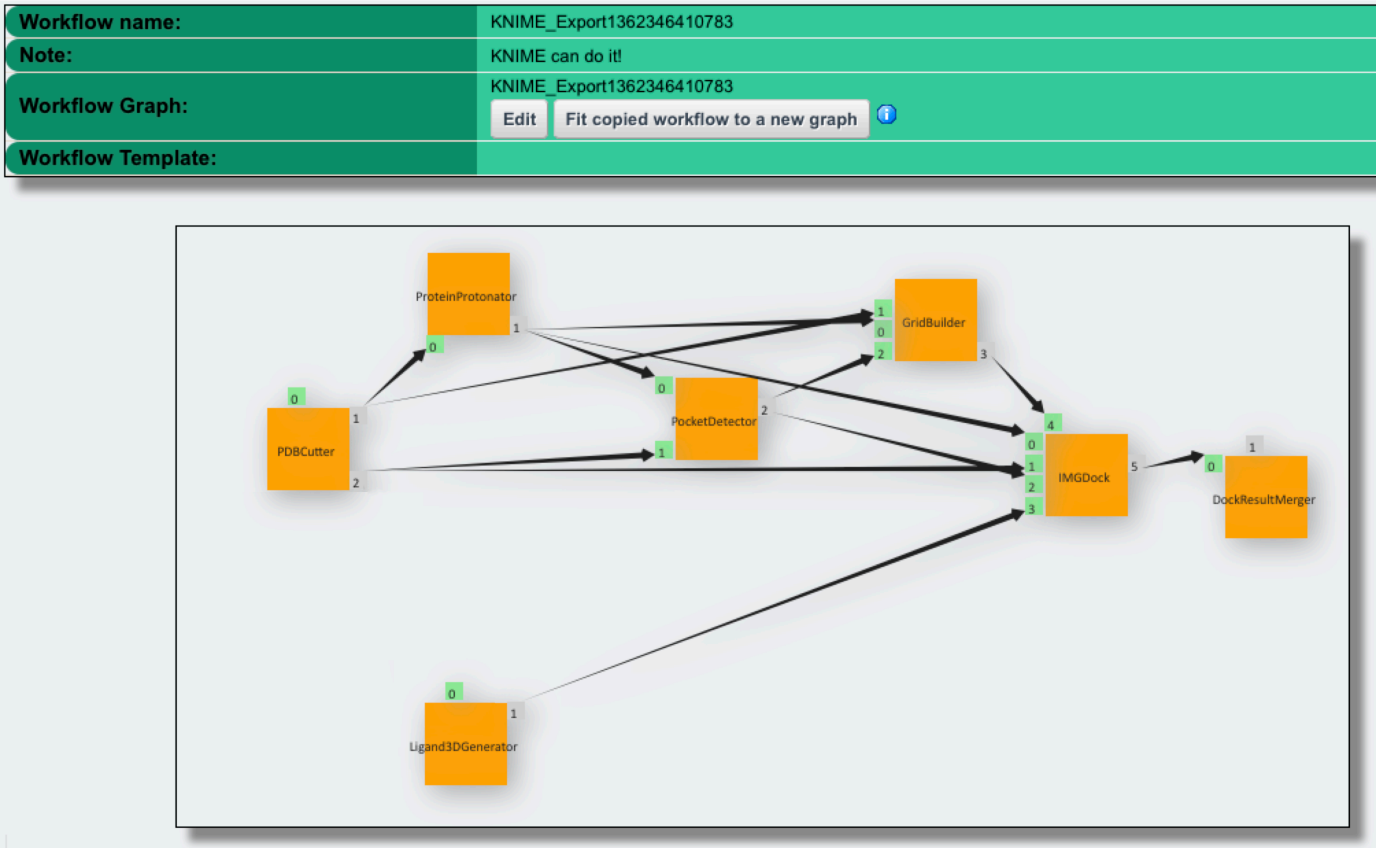
- The converter will generate a file in a specific format (*e.g.*, gUSE format) for a specific middleware (*e.g.* UNICORE)



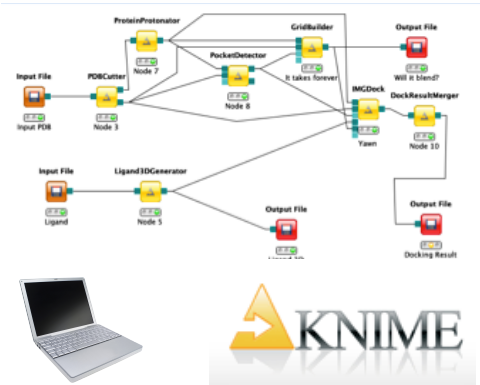
Docking on KNIME, then on the Grid

EBERHARD KARLS UNIVERSITÄT TÜBINGEN

- The generated file can be imported into WSPGRADE; input and output nodes in KNIME are converted to input and output ports in gUSE



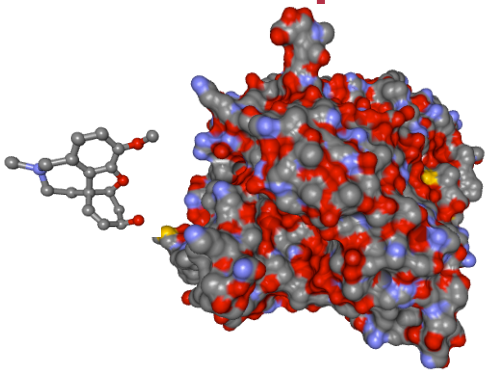
Recap



upload to gUSE
using novel converter

execute on a grid

easily create a workflow in a
desktop computer



scientific problem (docking)

Special Thanks to...

EBERHARD KARLS UNIVERSITÄT TÜBINGEN

University of Tübingen

Jens Krüger

Charlotta Schärfe

Marc Röttig

Sven Nahnsen

Oliver Kohlbacher

ETH Zürich

Béla Hullár

Peter Kunszt

Freie Universität Berlin

Stephan Aiche

Kurt Reinert

MTA SZTAKI

Zoltán Farkas

István Márton

Ákos Balaskó

Peter Kacsuk

University of Konstanz / KNIME.com

Peter Ohl

Throsten Meinl

Thomas Gabriel

Michael Berthold

The MoSGrid Project (BMBF 01IG09006)

The SCI-BUS Project (EU 283481)

Questions