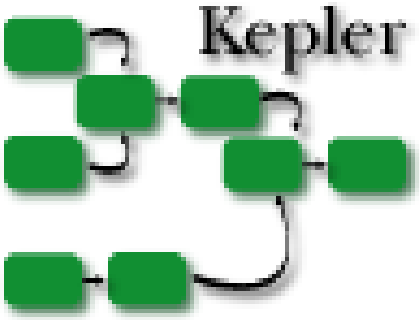


Accelerating the Scientific Exploration Process with Kepler Scientific Workflow System

Jianwu Wang, Ilkay Altintas
Scientific Workflow Automation Technologies Lab
SDSC, UCSD

Outline

- Scientific Workflow and Kepler
- Kepler in UCGrid
- Use Cases
  - Ecology Use Case
  - Chemistry Use Case

Part I: Scientific Workflow Systems and Kepler

Scientific Workflow Systems

- Mission of scientific workflow systems
  - Promote “scientific discovery” by providing tools and methods to generate larger, automated "scientific process"
  - Provide an extensible and customizable graphical user interface for scientists from different scientific domains
  - Support workflow design, execution, sharing, reuse and provenance
  - Design efficient ways to connect to the existing data and integrate heterogeneous data from multiple resources

Scientific Workflow

Capture how a scientist works with data and analytical tools

- data access, transformation, analysis, visualization
- possible worldview: dataflow-oriented (cf. *controlflow-oriented*)

Scientific workflow (wf) benefits (v.s. script-based approaches):

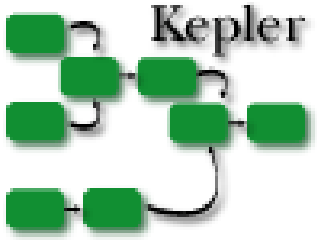
- wf & component **reuse, sharing, adaptation, archiving**
- wf **design, documentation**
- **built-in (model) concurrency**
- **provenance** support
- **distributed** & **parallel** exec: Grid & cluster support
- wf **fault-tolerance, reliability**

…

Why a W/F System?

Higher-level "language" vs. assembly-language nature of scripts

Kepler Scientific Workflow System



http://www.kepler-project.org

- Kepler is a cross-project collaboration: over 20 diverse projects and multiple disciplines.
- Open-source project; latest release available from the website
- Builds upon the open-source Ptolemy II framework
- Vergil is the GUI, but Kepler also runs in non-GUI and batch modes.

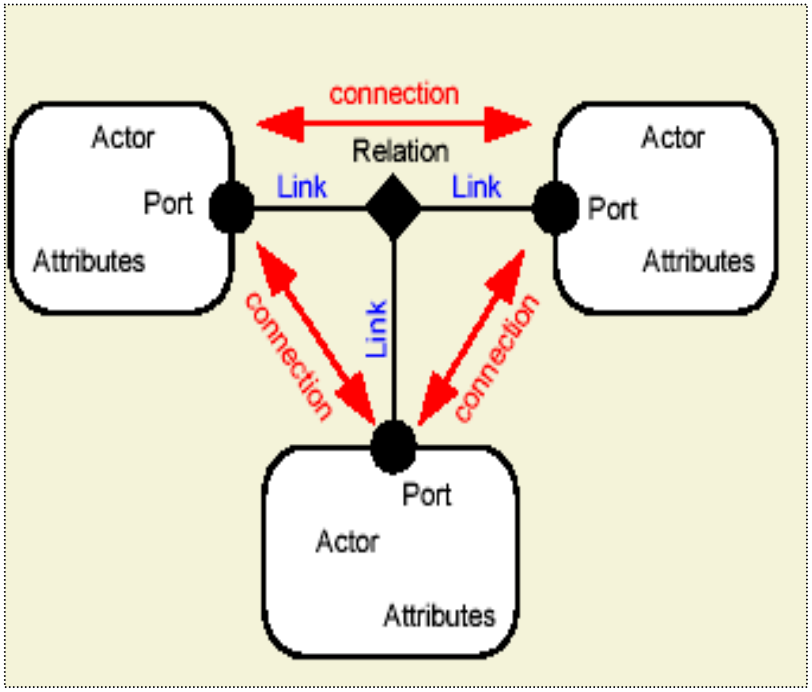
… initiated August 2003

- 1st release: May 13th, 2008
  - *More than 20 thousand downloads!*

Ptolemy II: A laboratory for investigating design
KEPLER: A problem-solving support environment for Scientific Workflow development, execution, maintenance

KEPLER = "Ptolemy II + X" for **Scientific Workflows**

Actors are the Processing Components



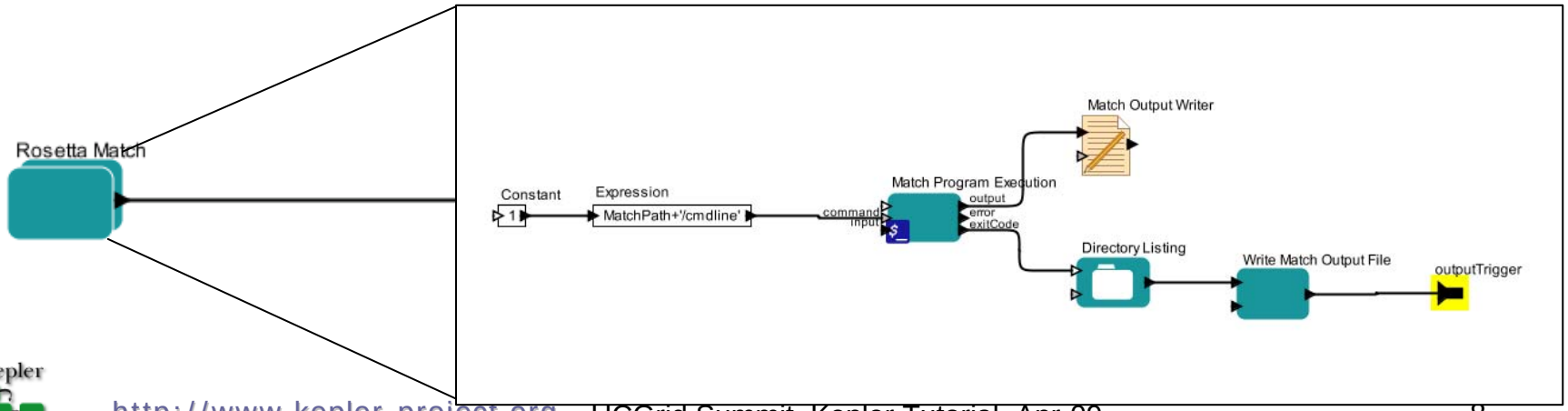
Actor-Oriented Design

- Actor
  - Encapsulation of parameterized actions
  - Interface defined by ports and parameters
- Port
  - Communication between input and output data
  - Without call-return semantics
- Relation
  - Links from output Ports to input Ports
  - Could be 1:1, m:n.
- Actor Examples
  - Web service Actor
  - Matlab Actor
  - File Read Actor
  - Local Execution Actor
  - Job Submission Actor
  - …

Adapted from the *.ppt slides by Edward A. Lee, UC Berkeley

Atomic and Composite Actors

- atomic actors: perform a single specific independent task.
- composite actors: collections or sets of atomic/composite actors bundled together to perform more complex operations.



Some actors in place for…

Currently more than 200 Kepler actors added!

- Generic Web Service Client
- Customizable RDBMS query and update
- Command Line wrapper tools (local, ssh, scp, ftp, etc.)
- Some Grid actors-*Globus Job Runner, GridFTP-based file access, Proxy Certificate Generator*
- SRB support
- Native R and Matlab support
- Interaction with Nimrod and APST Grid Environments
- Imaging, Gridding, Vis Support
- Textual and Graphical Output
- Python, JNI
- …more generic and domain-oriented actors…

Directors are the WF Engines that…

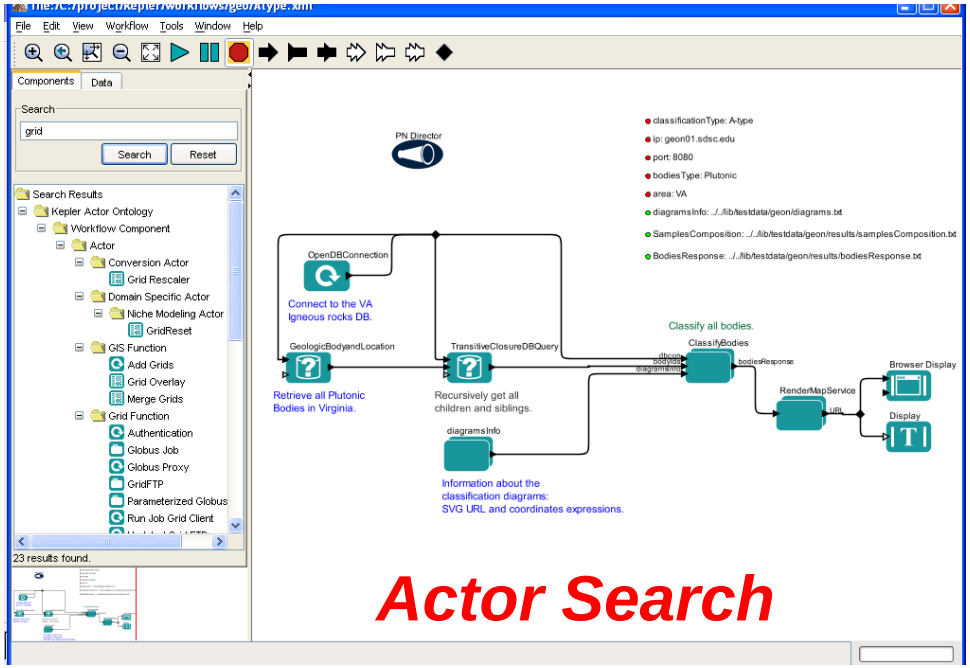
- Implement different computational models
- Define the semantics of
  - execution of actors and workflows
  - interactions between actors

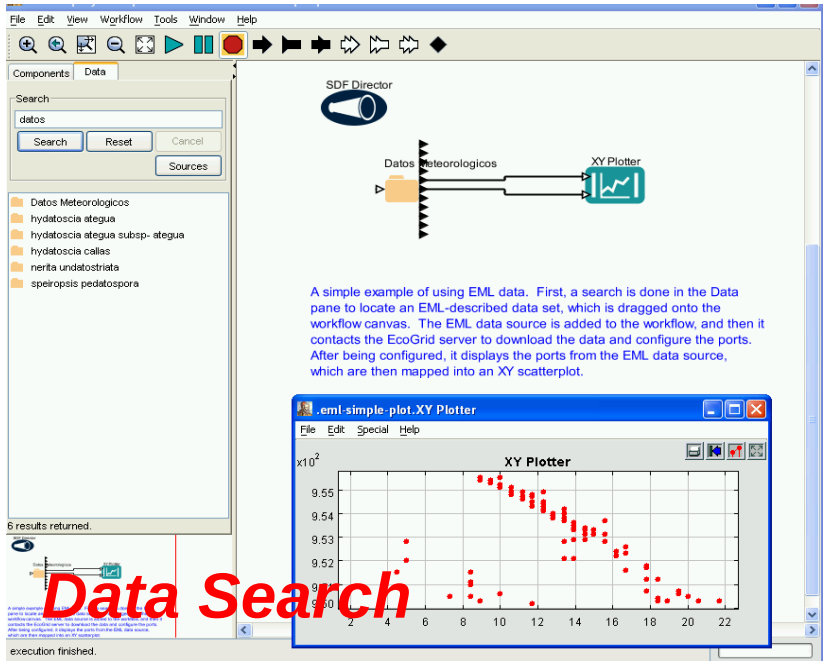
Ptolemy and Kepler are **unique** in combining different execution models in heterogeneous models!

- Kepler is extending Ptolemy directors with specialized ones for distributed workflows.

• Dataflow	• Process Networks
• Time Triggered	• Rendezvous
• Synchronous/reactive model	• Publish and Subscribe
• Discrete Event	• Continuous Time
• Wireless	• Finite State Machines

Kepler Modeling with GUI





- Actor ontology and semantic search for actors
- Search -> Drag and drop -> Link via ports
- Metadata-based search for datasets

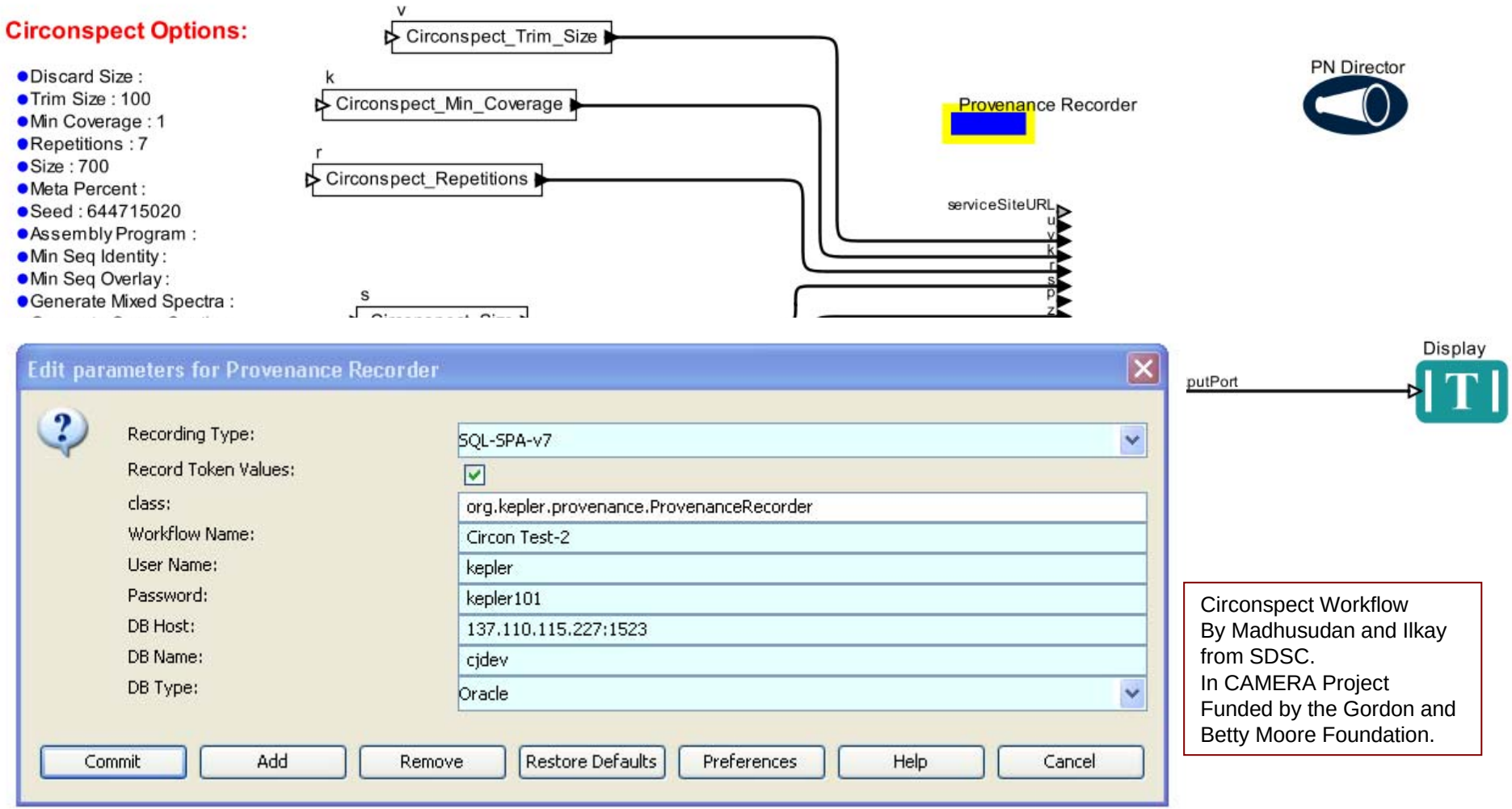
Kepler Execution

- From GUI: click execution button
- From Kepler Web Service: for detached execution
  - Synchronous: executByContent, executeByURI, …
  - Asynchronous: startExeByContent, getStatus, get Result, …
- Batch Mode: useful for command line and job submission
  - Kepler.sh [config] workflow.xml

Provenance of Workflow Related Data

- Provenance: A concept from art history and library
  - Inputs, outputs, intermediate results, workflow design, workflow run
- Collected information
  - Can be used in a number of ways
    - Validation, reproducibility, fault tolerance, etc…
  - Can be recorded in a number of ways
    - System.out, text file, databases, etc…
  - Viewable and searchable from outside of Kepler

Running Provenance Recorder

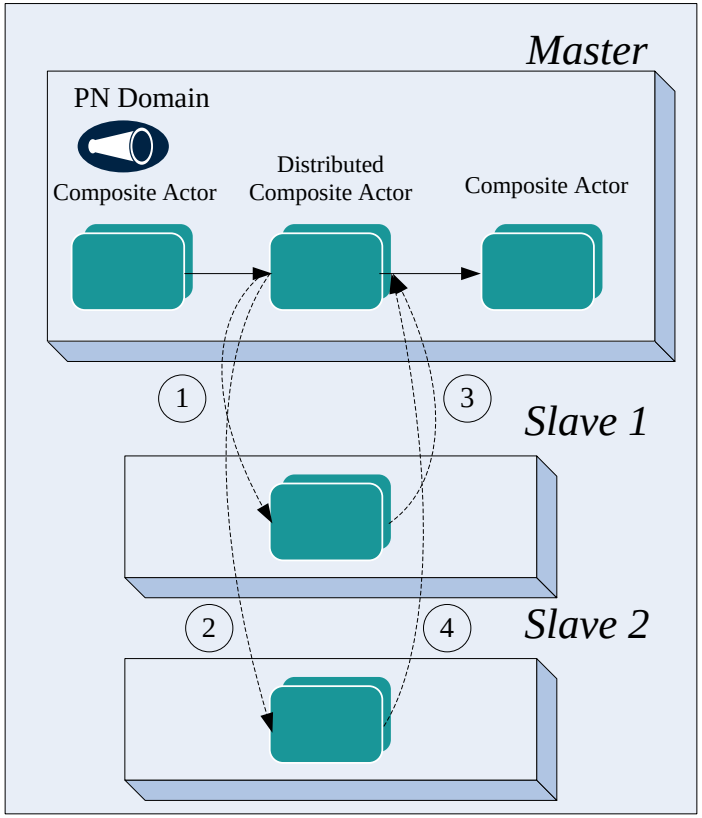


Circonspect Workflow
By Madhusudan and Ilkay from SDSC.
In CAMERA Project
Funded by the Gordon and Betty Moore Foundation.

Part II: Kepler in UC Grid

Master-Slave Distributed Execution Framework

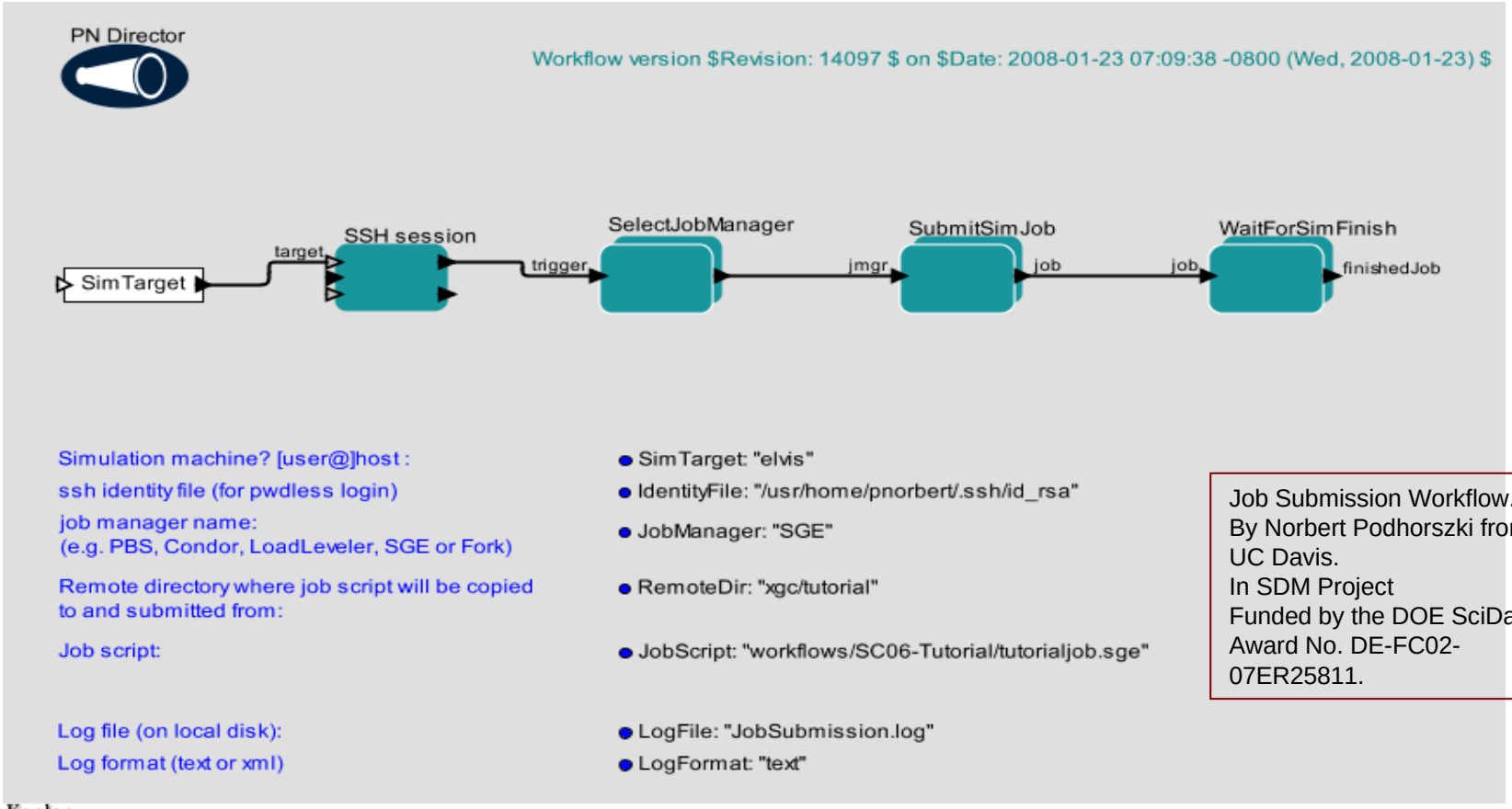
- Utilize distributed resources to accelerate workflow execution
- Smooth transition between different execution environments, such as local, ad-hoc network, cluster, grid and cloud



Cluster Job Submission Actors

- Adaptable for different cluster schedulers, such as SGE and PBS
- Adaptable for local execution and ssh execution

Example of Job Submission Actors



Job Submission Workflow. By Norbert Podhorszki from UC Davis.
In SDM Project
Funded by the DOE SciDac Award No. DE-FC02-07ER25811.

Grid Actors

- Actors: Grid Authentication, Globus Job, Grid Proxy, GridFTP, …
- Support both Pre-WS and WS Globus Resource Invocation

Collaboration of Kepler and UCGrid

- UCGrid provides abundant computing and software resources for scientists
- Kepler provides a bridge for scientists to easily utilize the above resources according to their domain problems
- Scientists compose individual tasks by Kepler workflows and run them in UCGrid

Usage Modes of Kepler in UCGrid

- **Kepler Application in UCGrid**: Users model workflows from Kepler GUI, upload them to UCGrid portal, and execute them through Kepler batch-mode command

- **Kepler Globus Web Service in UCGrid**: With UCGrid authentication, We can integrate user applications with UCGrid, their tasks be executed through deployed Kepler WS

- **Direct Execution from Kepler GUI**: With UCGrid authentication, users can model workflows that submit jobs to UCGrid, and execute them from Kepler GUI

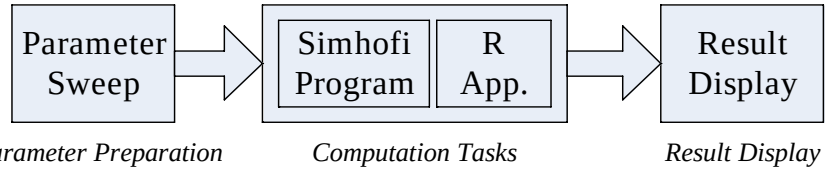
Part III: Use Cases

Theoretical Ecology Use Case

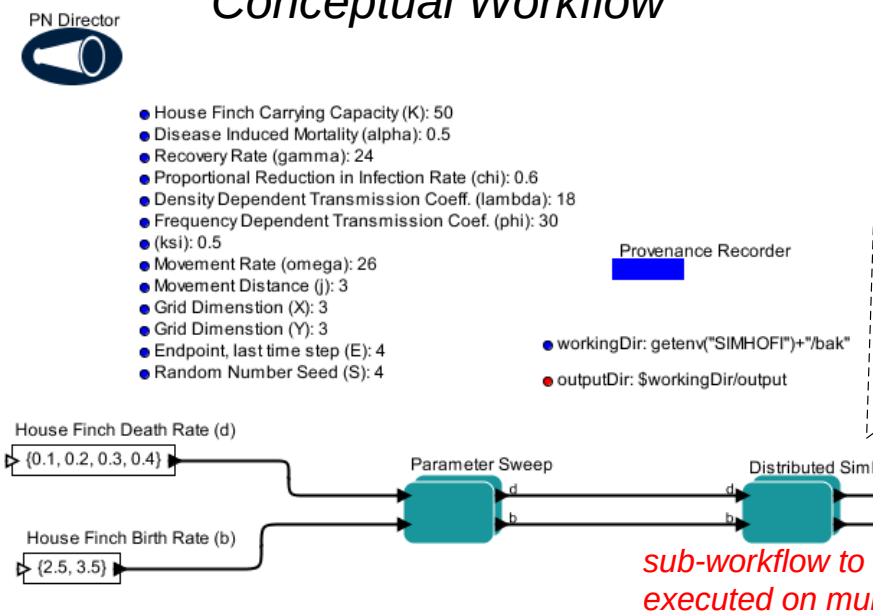
- It is a spatial stochastic birth-death process that simulates the dynamics of Mycoplasma gallisepticum in House Finches (Carpodacus mexicanus)
- The simulation code is written in GNU C++, and involves file reads, relatively complex mathematical operations
- The execution results were visualized using the R statistical system
- It needs to be run with a broad range of parameter sweep, namely the computing code may be iterated for over hundreds times with different parameter configurations

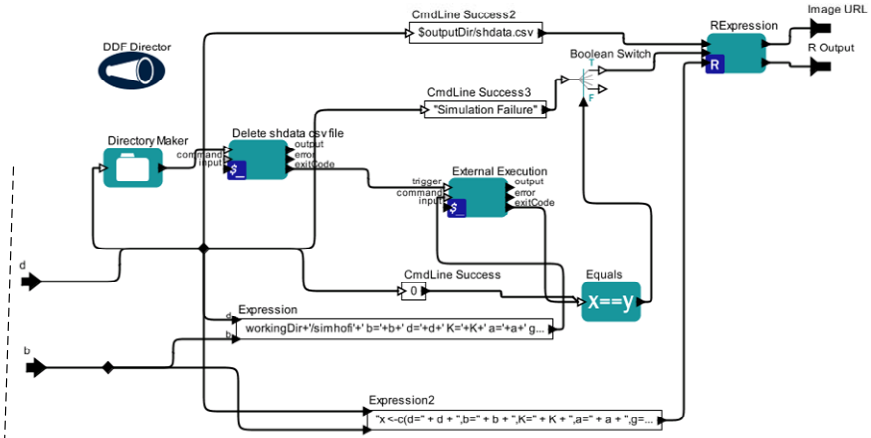
Collaboration with Parviez R. Hosseini (Princeton Univ.), Derik Barseghian (UCSB)
In REAP (Realtime Environment for Analytical Processing) project (http://reap.ecoinformatics.org/)
Funded by NSF CEO:P Award No. DBI 0619060

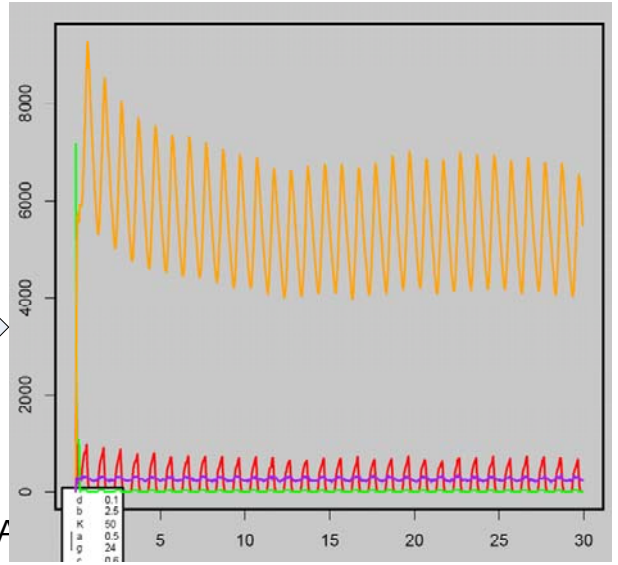
Conceptual and Kepler Workflow



Conceptual Workflow







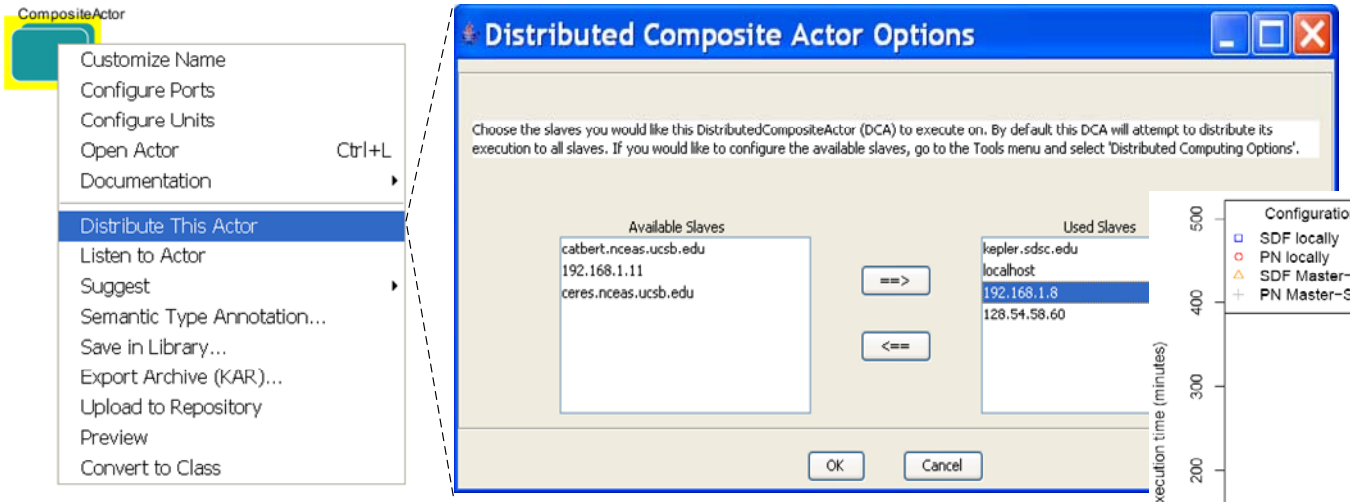
sub-workflow to be executed on multiple nodes.

Kepler Workflow

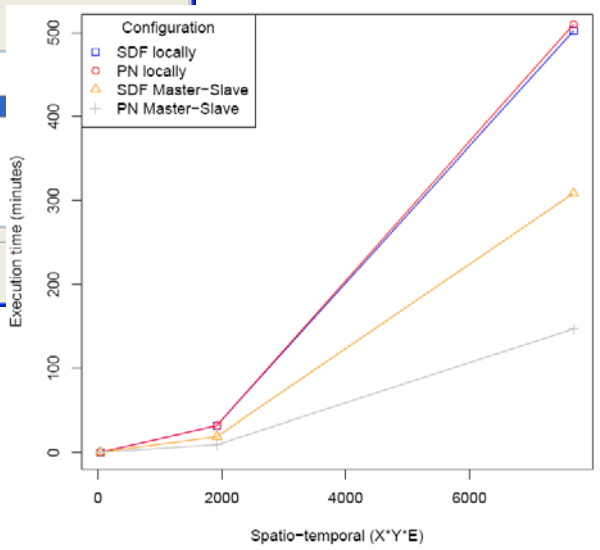
Kepler

http://www.kepler-project.org

Configuration and Experiments



Interaction for execution environment transition



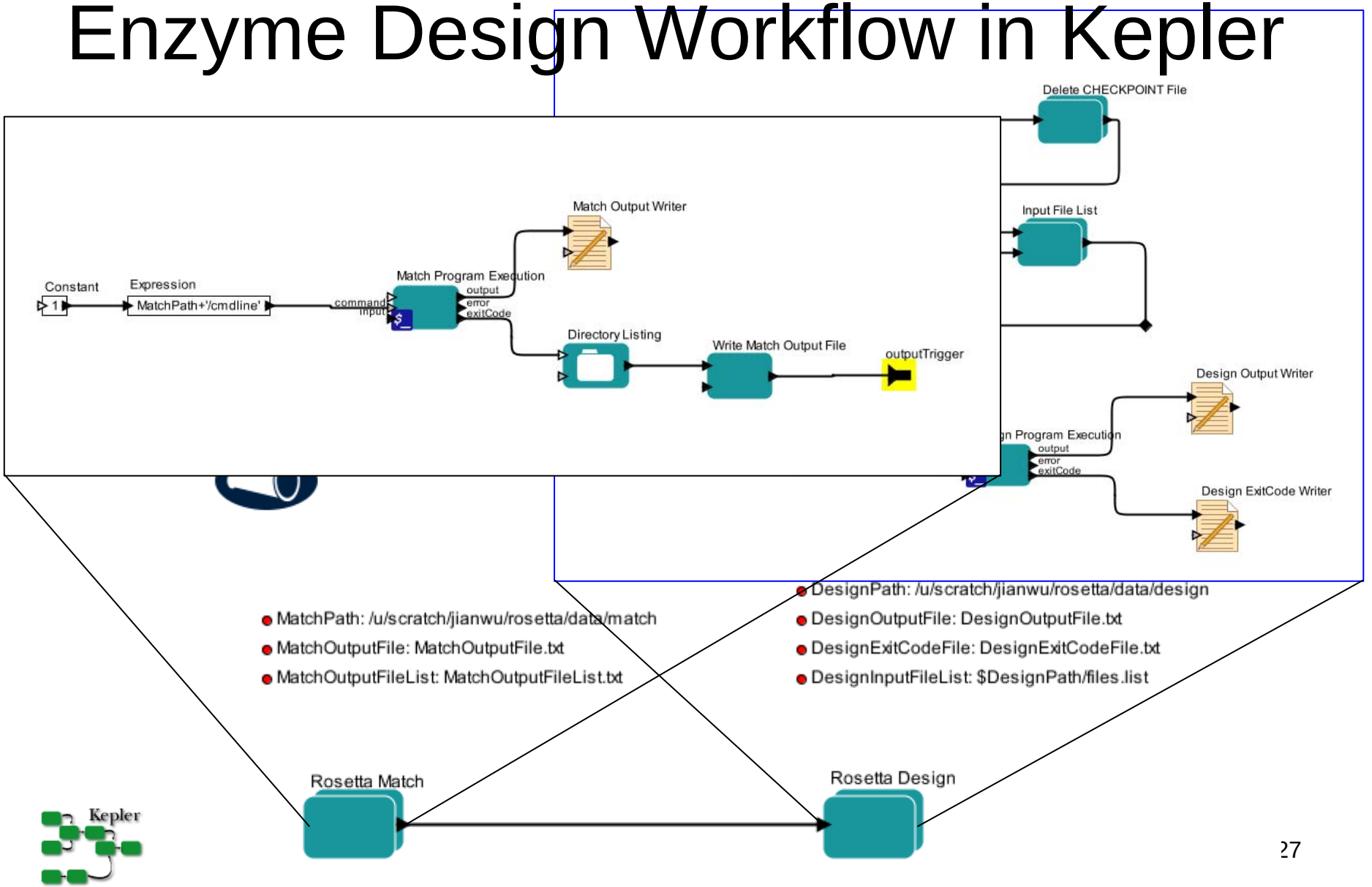
Experiment data

Computational Chemistry Use Case

- The whole goal is to (re)design existing enzymes to catalyze a novel chemical reaction
- The workflow will provide an automated way of generating enzyme designs from a model
  - allows scientists to focus on creating better models
  - rather than fussing with a number of different programs
- Each execution will generate over 4000 Protein Data Bank files which could be processed concurrently

Collaboration with Scott Johnson, Seonah Kim, Prakashan Korambath, Kejian Jin (UCLA) and Shava Smallen (SDSC).

Enzyme Design Workflow in Kepler



Main Work For Enzyme Design Workflow

- Three versions of Enzyme Design Workflow
  - Execute the Enzyme programs directly and locally – Done
  - Wrap the programs and submit as SGE jobs at Hoffman2 cluster – Done
  - Wrap the programs and submit as Globus jobs at UCGrid – On Going
- Accelerate Workflow with UCGrid
  - With Kepler Cluster Job Submission Actor and Hoffman2 cluster, the execution time is reduced from 2000 mins (in theory) to 80 mins
  - Using Kepler with Grid resources will enable better parallel execution among multiple Grid nodes and reduce the whole execution time largely
- Provenance Support
  - Each workflow execution will generate over 4000 pdb files and scientists need the workflow to executed for many times with different input model
  - Provenance can help scientists to track the data efficiently in the future

Thanks!

&

Questions…

Jianwu Wang
jianwu@sdsc.edu
+1 (858) 534-5110

Kepler Download:
https://kepler-project.org/users/downloads
Kepler Documents:
https://kepler-project.org/users/documentation