Dynamic Job Expansion: Experiences using Makeflow in Galaxy

Nicholas Hazekamp, Joseph Sarro, Olivia Choudhury, Sandra Gesing, Scott Emrich and Douglas Thain

Galaxy Dev Meeting

CCTools

UNIVERSITY OF NOTRE DAME

The Cooperative Computing Lab

- We *collaborate with people* who have large scale computing problems in science, engineering, and other fields.
- We *operate computer systems* on the O(10,000) cores: clusters, clouds, grids.
- We *conduct computer science* research in the context of real people and problems.
- We *develop open source software* for large scale distributed computing.

CCTools

UNIVERSITY OF NOTRE DAME

Our Philosophy:

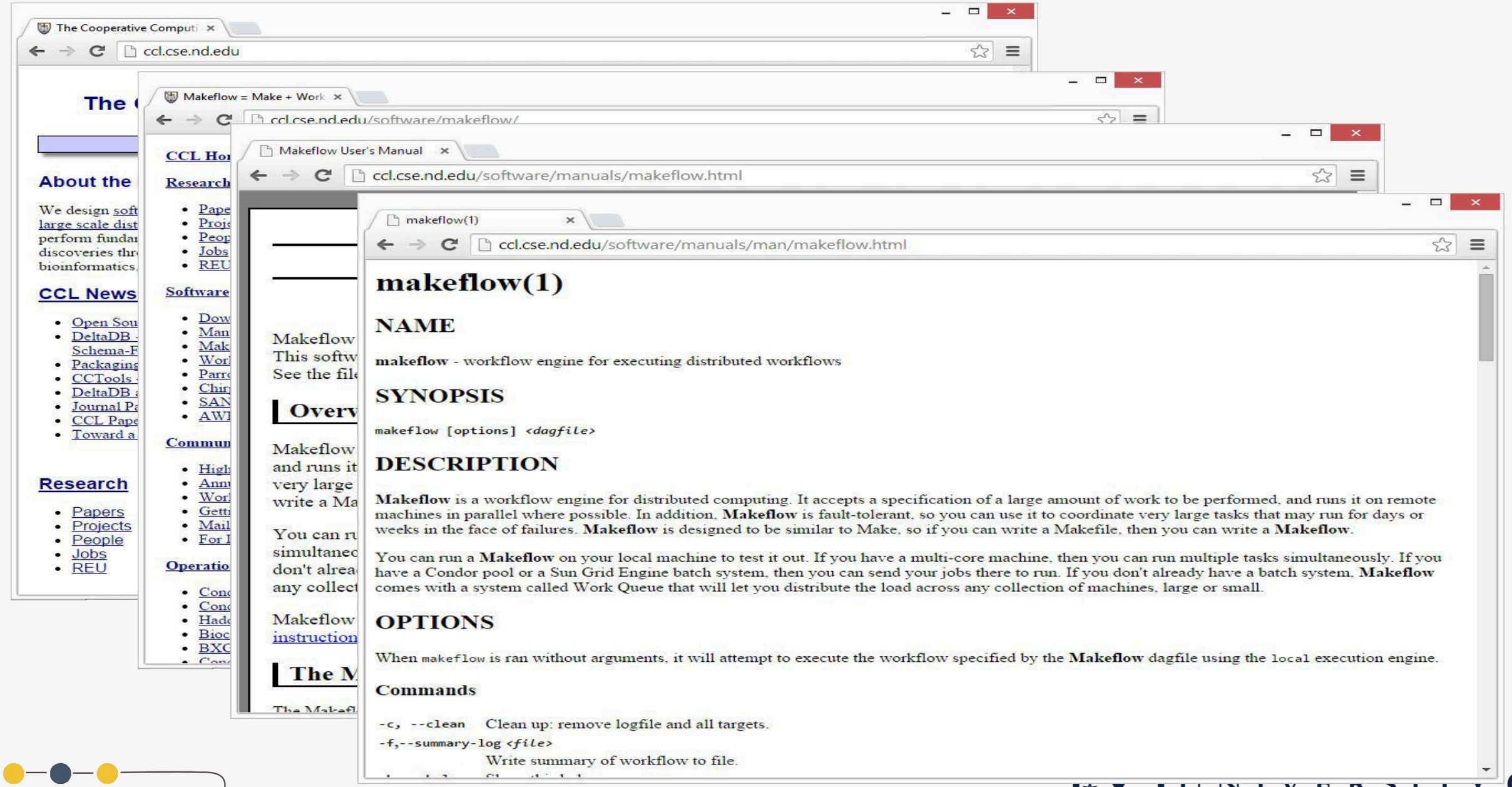
- Harness all the resources that are available: desktops, clusters, clouds, and grids.
- Make it easy to scale up from one desktop to national scale infrastructure.
- Provide familiar interfaces that make it easy to connect existing apps together.
- Allow portability across operating systems, storage systems, middleware…
- Make simple things easy, and complex things possible.
- ***No special privileges required.***

A Quick Tour of the CCTools

- Open source, GNU General Public License.
- Compiles in 1-2 minutes, installs in $HOME.
- Runs on Linux, Solaris, MacOS, Cygwin, FreeBSD, …
- Interoperates with many distributed computing systems.
  - Condor, SGE, SLURM, TORQUE, Globus, iRODS, Hadoop…
- Components:
  - Makeflow – A portable workflow manager.
  - Work Queue – A lightweight distributed execution system.
  - All-Pairs / Wavefront / SAND – Specialized execution engines.
  - Parrot – A personal user-level virtual file system.
  - Chirp – A user-level distributed filesystem.

CCTools

UNIVERSITY OF NOTRE DAME



CCTools

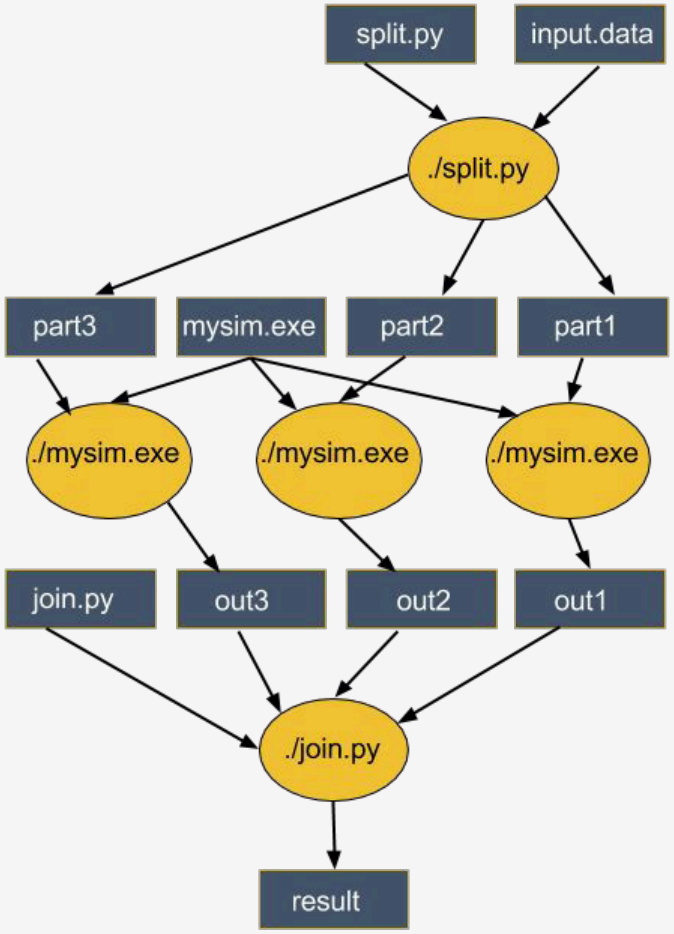
UNIVERSITY OF NOTRE DAME

Makeflow:
A Portable Workflow System

CCTools

UNIVERSITY OF NOTRE DAME

An Old Idea: Makefiles

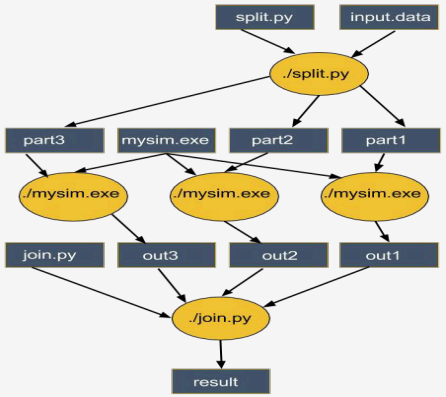



```
part1 part2 part3: input.data split.py
    ./split.py input.data

out1: part1 mysim.exe
    ./mysim.exe part1 >out1

out2: part2 mysim.exe
    ./mysim.exe part2 >out2

out3: part3 mysim.exe
    ./mysim.exe part3 >out3

result: out1 out2 out3 join.py
    ./join.py out1 out2 out3 > result
```

CCTools

UNIVERSITY OF NOTRE DAME

Makeflow = Make + Workflow



- Provides portability across batch systems.
- Enable parallelism (but not too much!)
- Trickle out work to batch system.
- Fault tolerance at multiple scales.
- Data and resource management.

Makeflow

Local | SLURM | TORQUE | Work Queue

CCTools

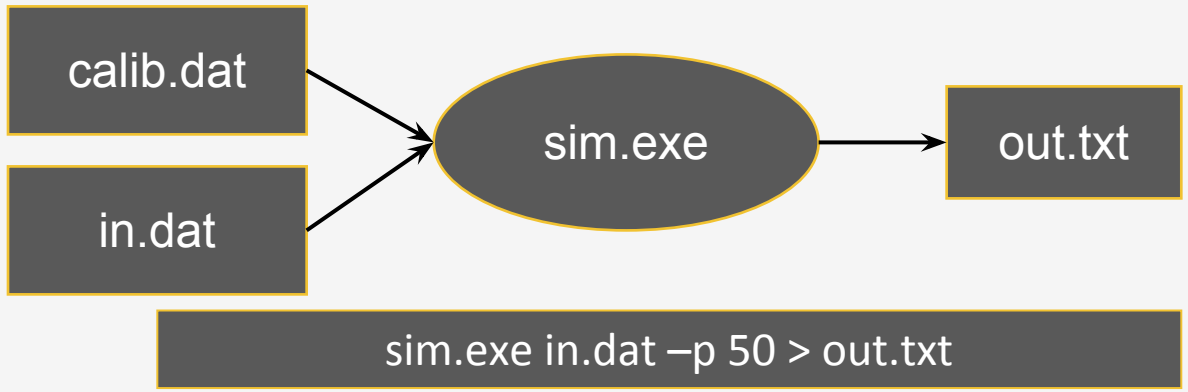
UNIVERSITY OF NOTRE DAME

Makeflow Syntax

[output files] : [input files]

[command to run]

One rule



out.txt：in.dat calib.dat sim.exe

sim.exe –p 50 in.data > out.txt

CCTools

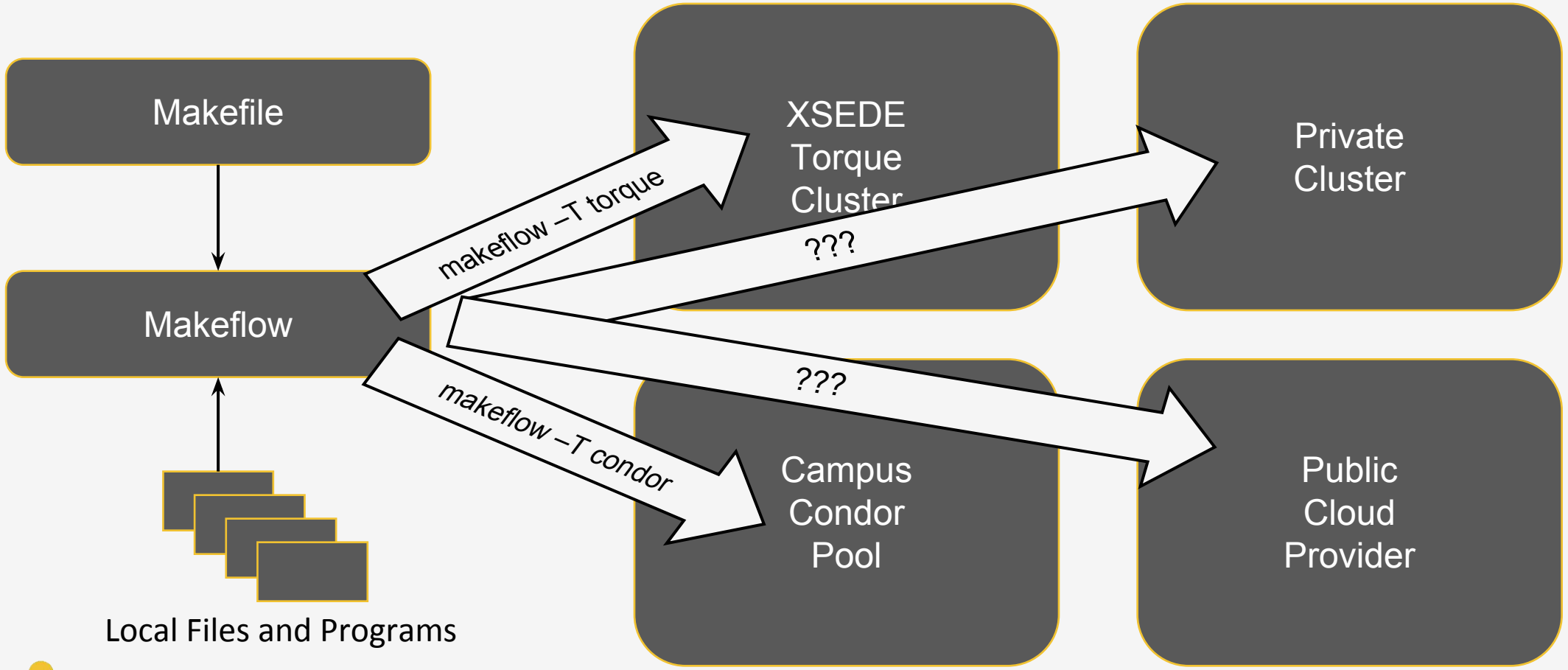
UNIVERSITY OF NOTRE DAME

Makeflow + Work Queue

CCTools

UNIVERSITY OF NOTRE DAME

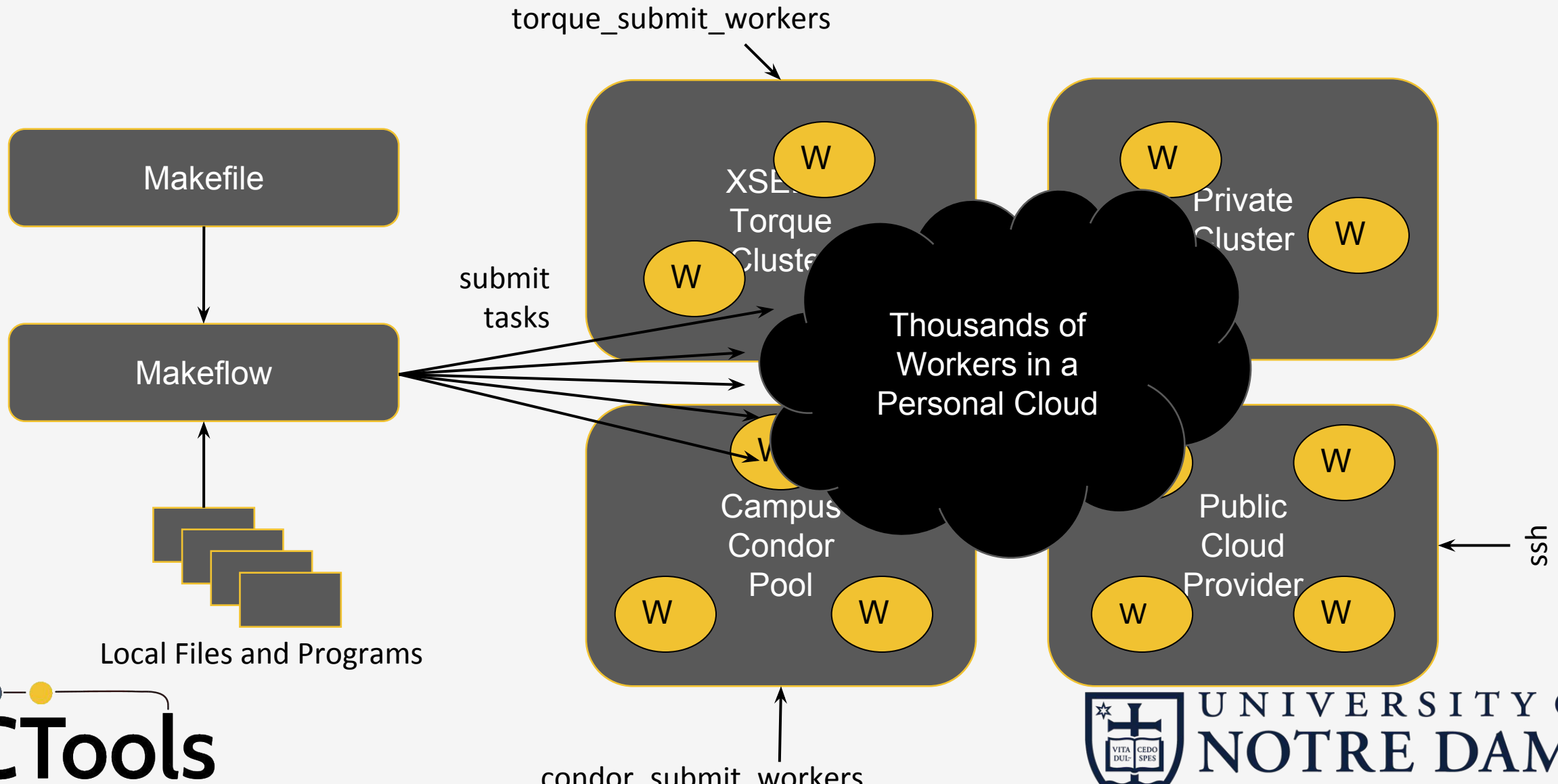
Makeflow + Batch System



CCTools

UNIVERSITY OF NOTRE DAME

Makeflow + Work Queue



CCTools

UNIVERSITY OF NOTRE DAME

Advantages of Work Queue

- Harness multiple resources simultaneously.
- Pilot jobs (Work Queue workers) hold on to cluster nodes to execute multiple tasks rapidly.
- Scale resources up and down as needed.
- Better management of data, with local caching for data intensive tasks.
- Matching of tasks to nodes with data.

Dynamic Job Expansion

CCTools

UNIVERSITY OF NOTRE DAME

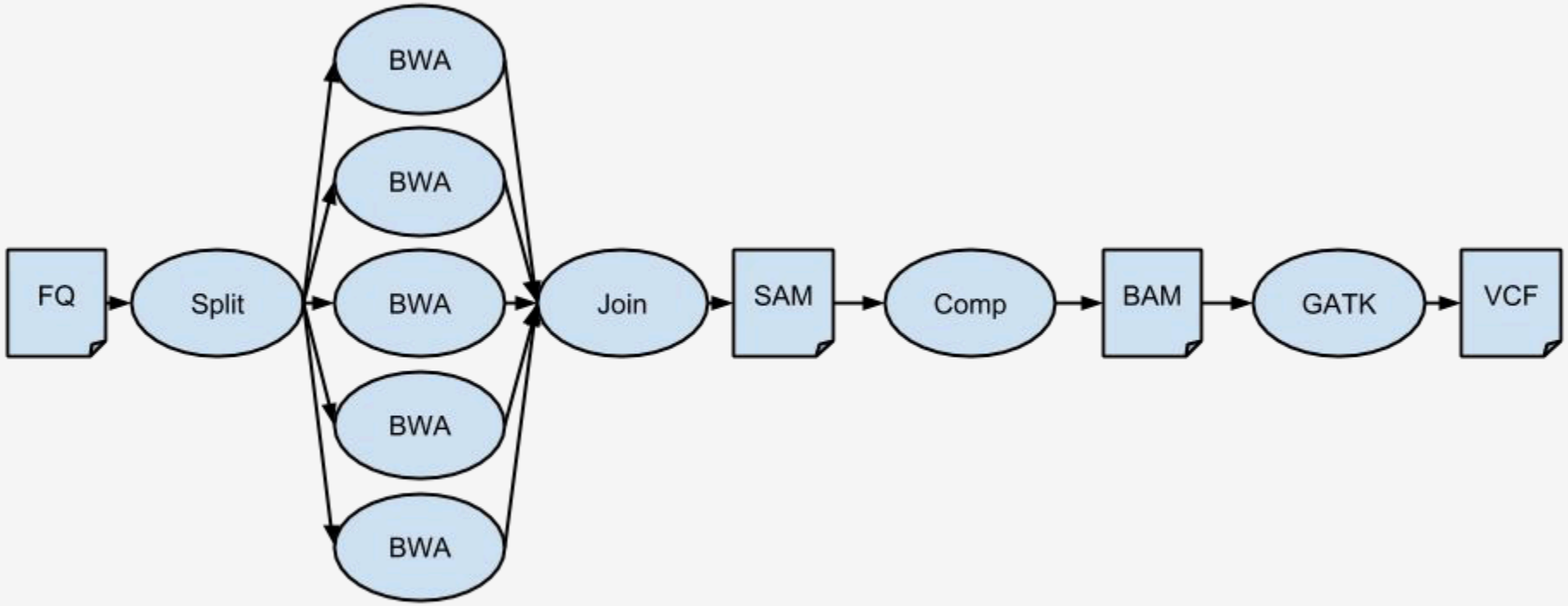
Simple Workflow in Galaxy

Problem: As Size increases so does Time

CCTools

UNIVERSITY OF NOTRE DAME

Workflow with Parallelism added in Galaxy

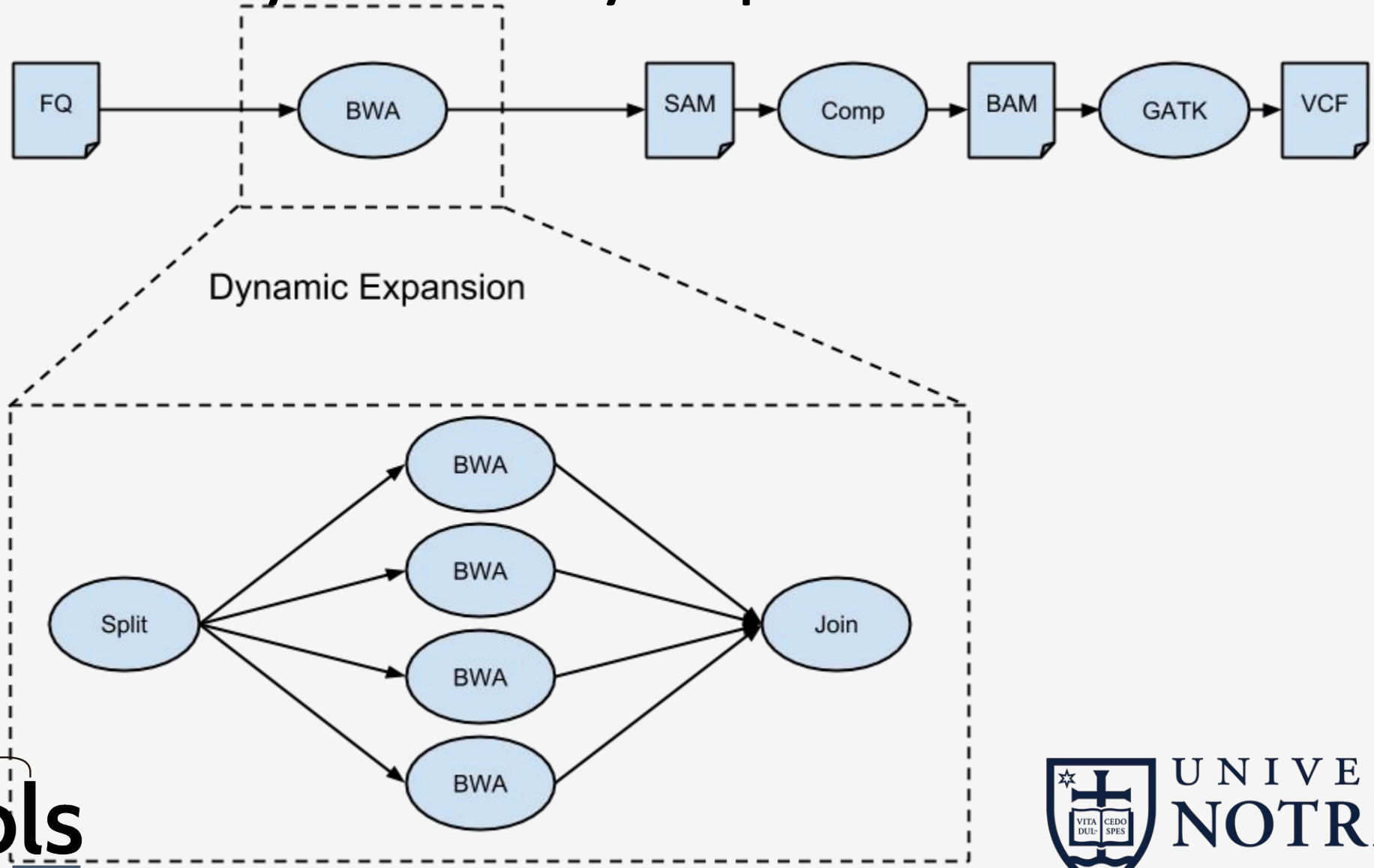


Problem: Tools must be updated every change in Parallelism/Relies on Scientist

CCTools

UNIVERSITY OF NOTRE DAME

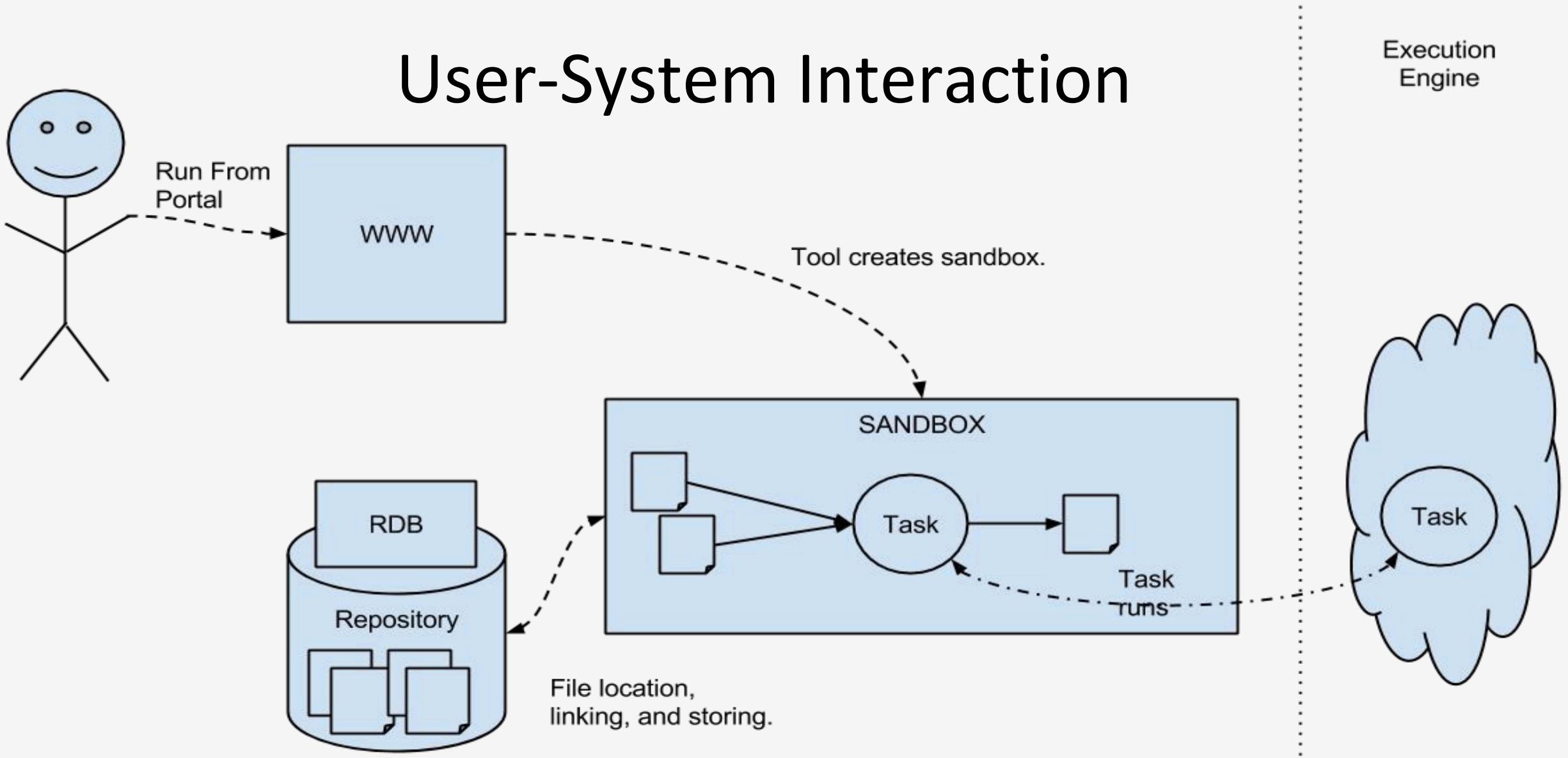
Workflow Dynamically Expanded behind Galaxy



CCTools

UNIVERSITY OF NOTRE DAME

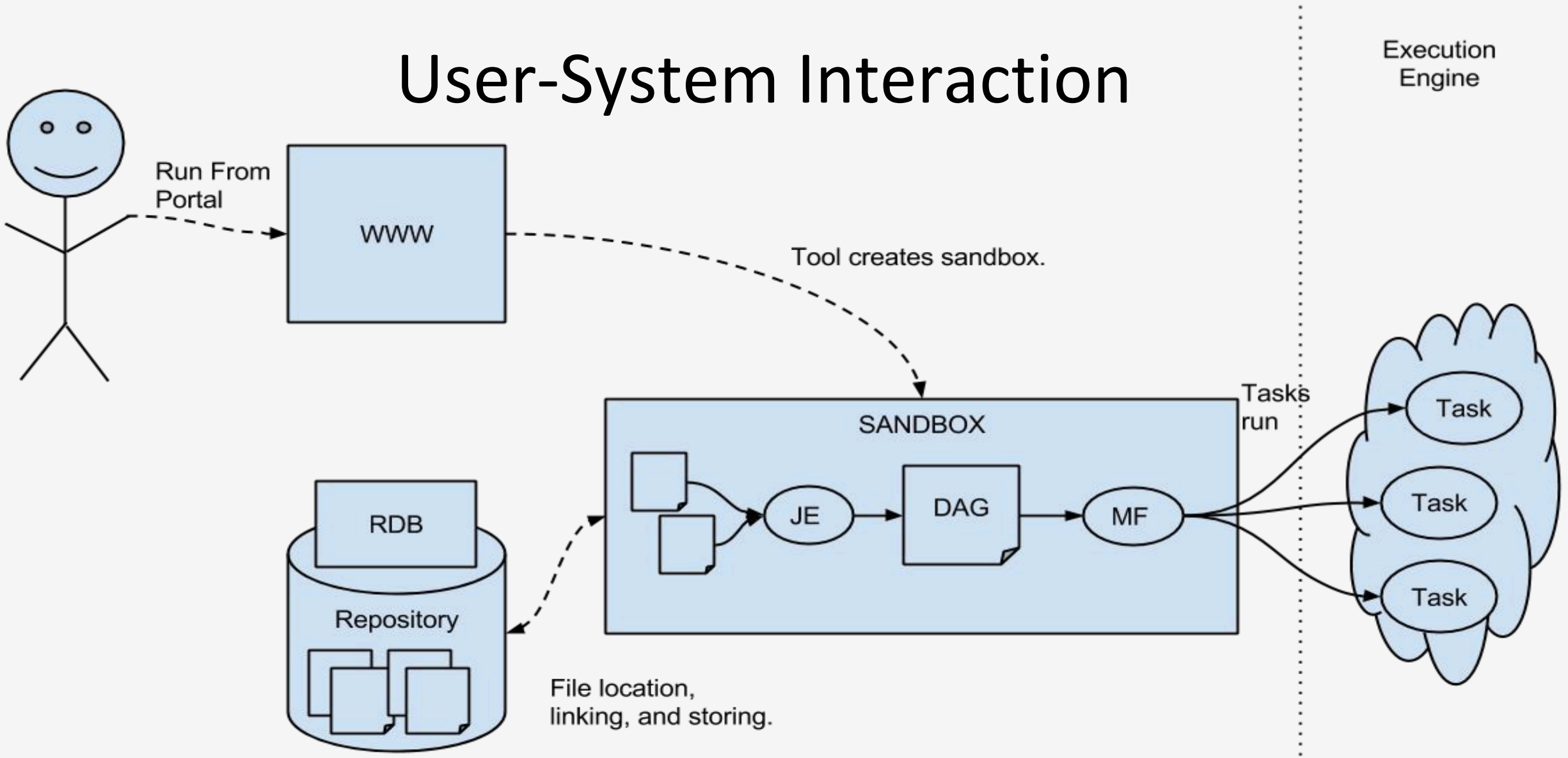
User-System Interaction



CCTools

UNIVERSITY OF NOTRE DAME

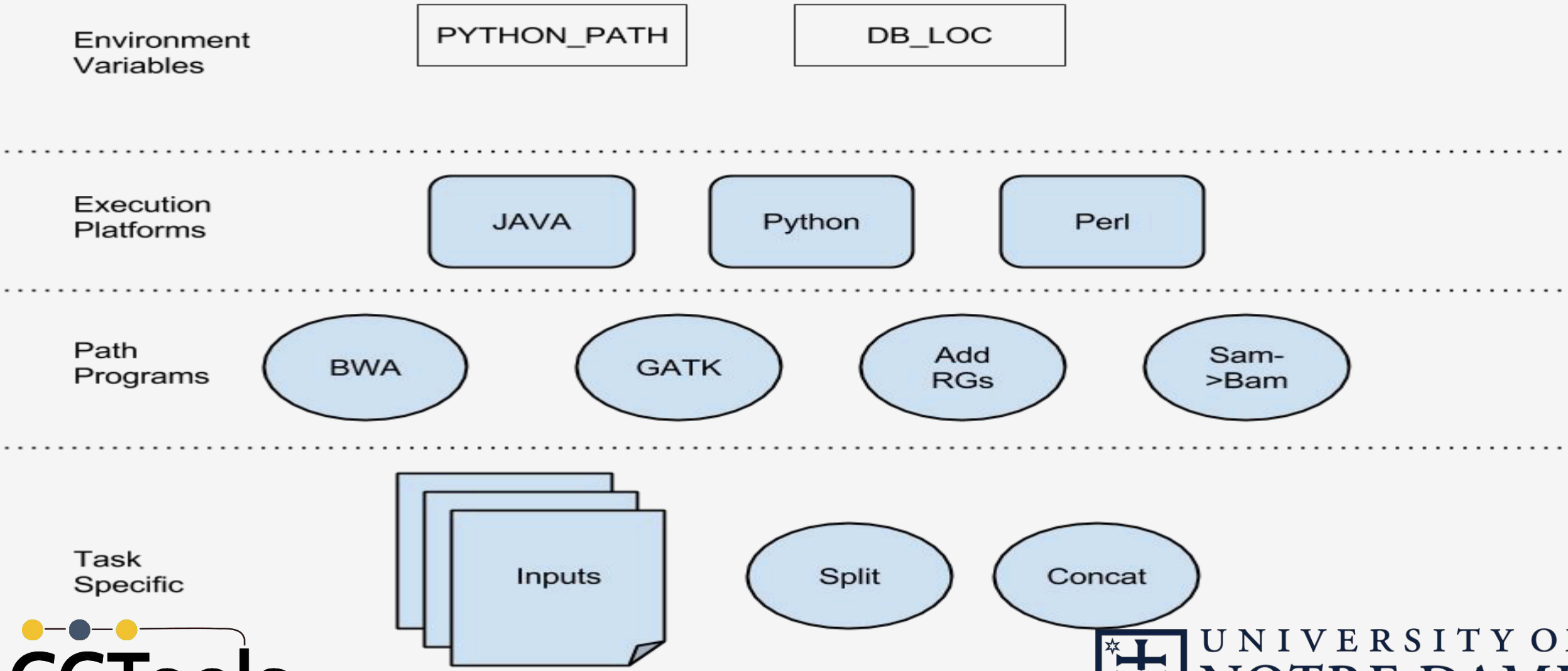
User-System Interaction



CCTools

UNIVERSITY OF NOTRE DAME

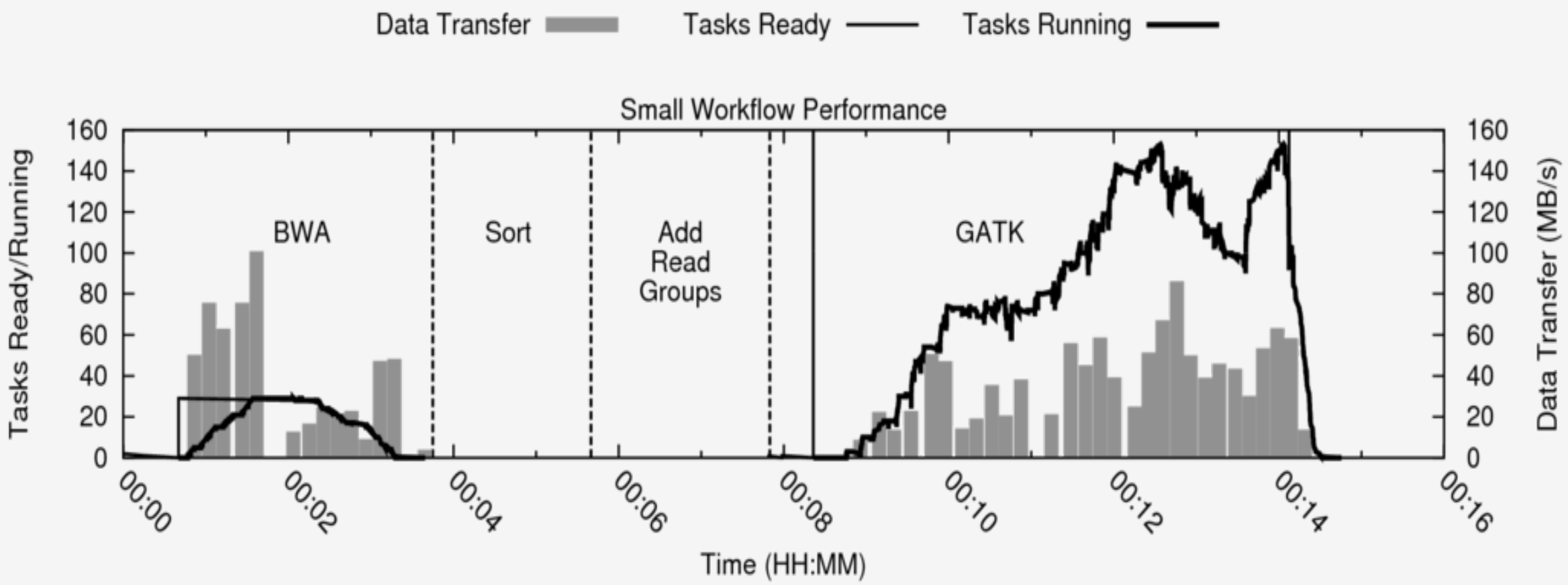
Managing Environmental Expectations



CCTools

UNIVERSITY OF NOTRE DAME

Small Scale Run

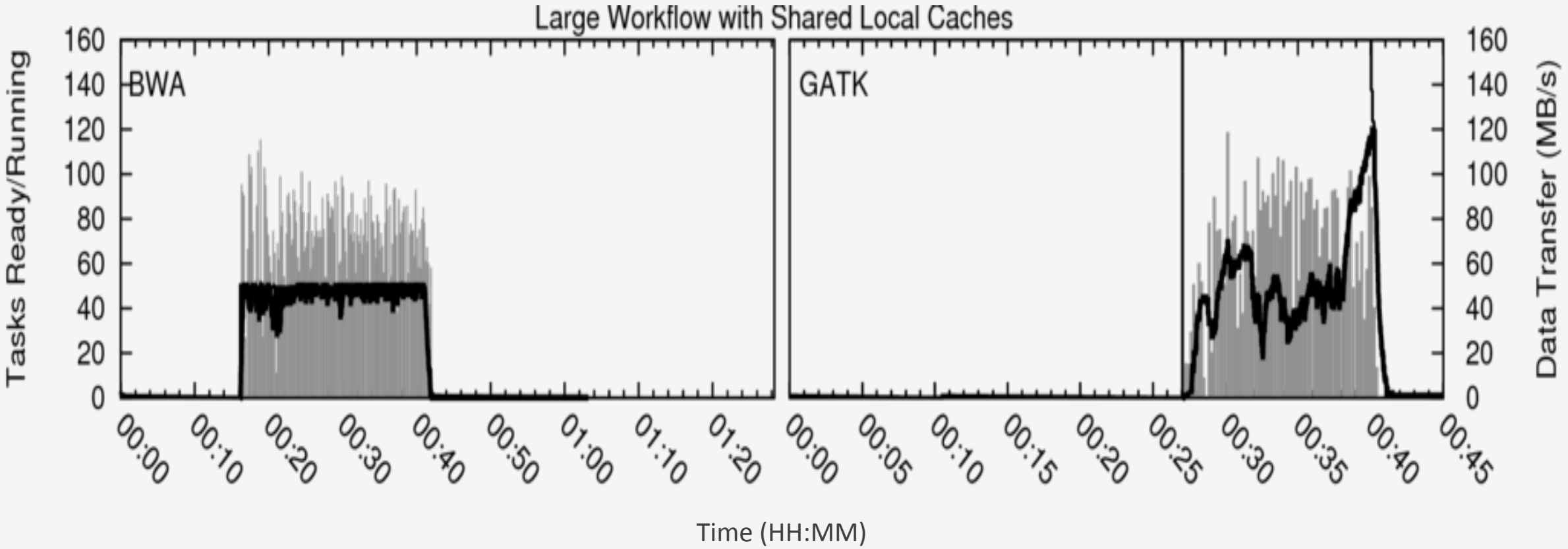


CCTools

Query: 600MB Ref: 36MB

UNIVERSITY OF NOTRE DAME

Full Scale Run



CCTools

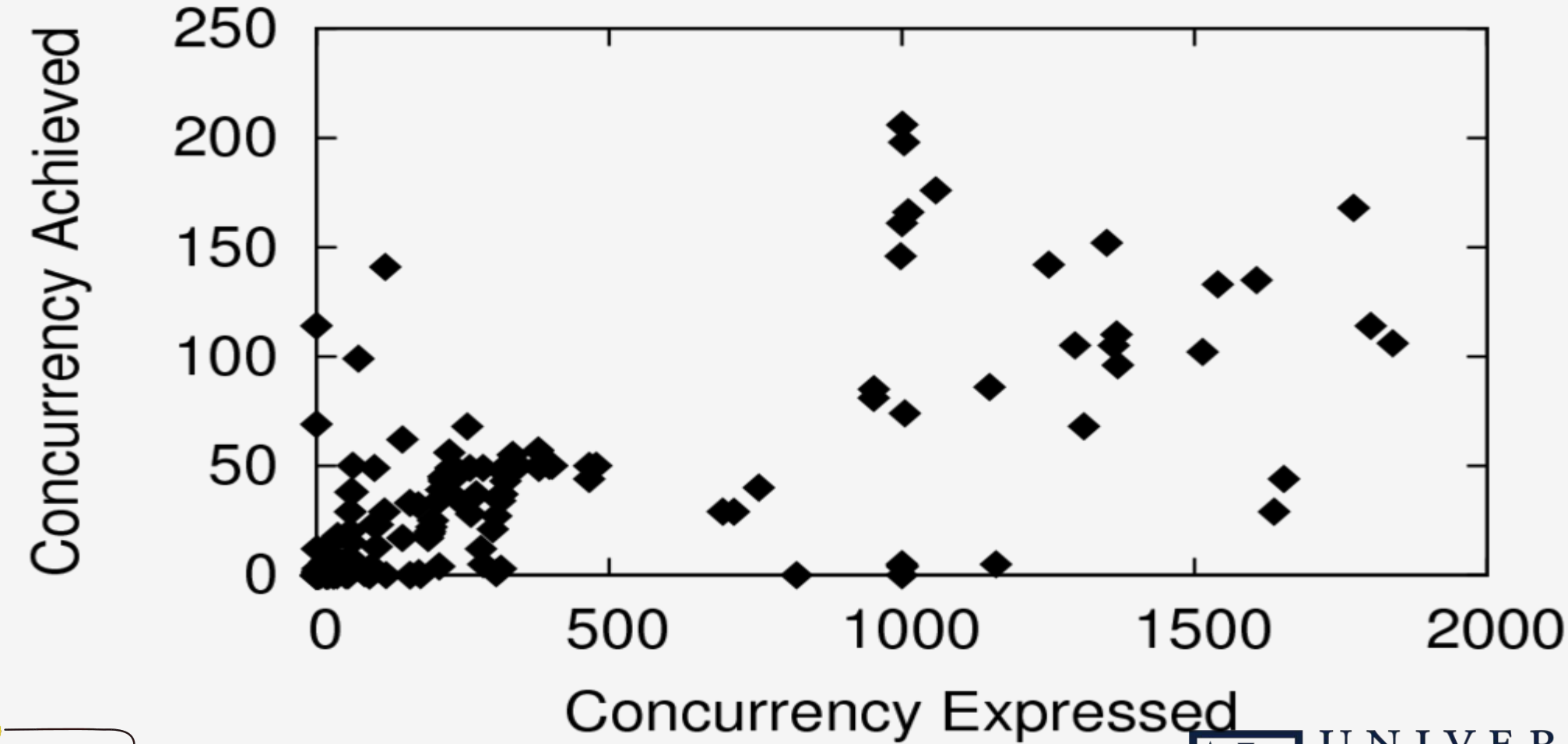
Query: 32GB Ref: 36MB

UNIVERSITY OF NOTRE DAME

Performance in Real-Life

- 100+ Different runs through Workflow
- Utilizing 500+ Cores with heavy load
- Data sets ranging from >1GB to 50GB+

Real Usage Concurrency Comparison



CCTools

UNIVERSITY OF NOTRE DAME

Conclusions

- Using Dynamic Job Expansion we were able to scale up a workflow without requiring the huge amount of time to process.
- Found viable solutions for:
  - Using Work Queue we utilized 100s of cores from a Condor Pool
  - Cleaning Sandbox using knowledge of intermediates and logging
  - Explored methods to transmit needed environments such as executables and Java
- 61.5X speed-up on 32 GB dataset utilizing these methods

CCTools

UNIVERSITY OF NOTRE DAME

Visit our website: ccl.cse.nd.edu

Follow us on Twitter: @ProfThain

Check out our blog: cclnd.blogspot.com

CCTools

UNIVERSITY OF NOTRE DAME

Join us at the

Cooperating Computing Lab

Workshop on Scalable Computing

October 19th-20th

For details visit: ccl.cse.nd.edu/workshop/2016/

CCTools

UNIVERSITY OF NOTRE DAME