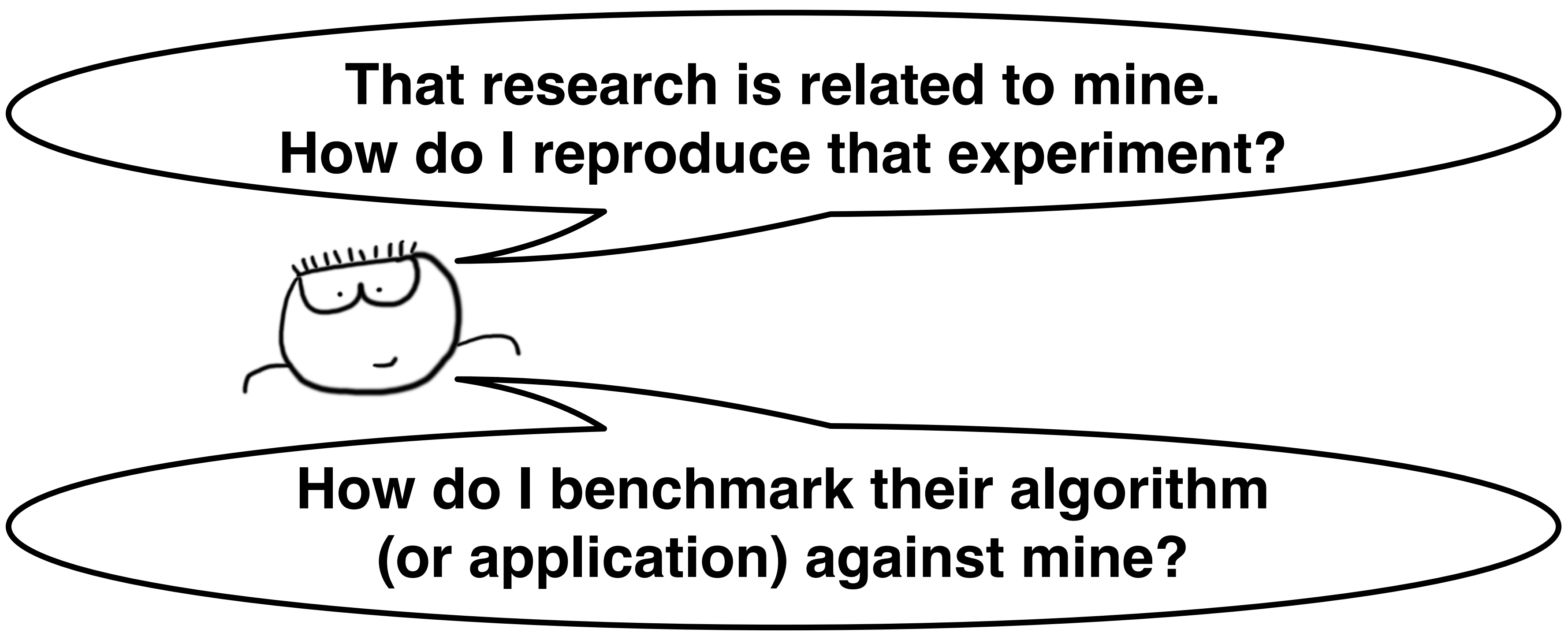
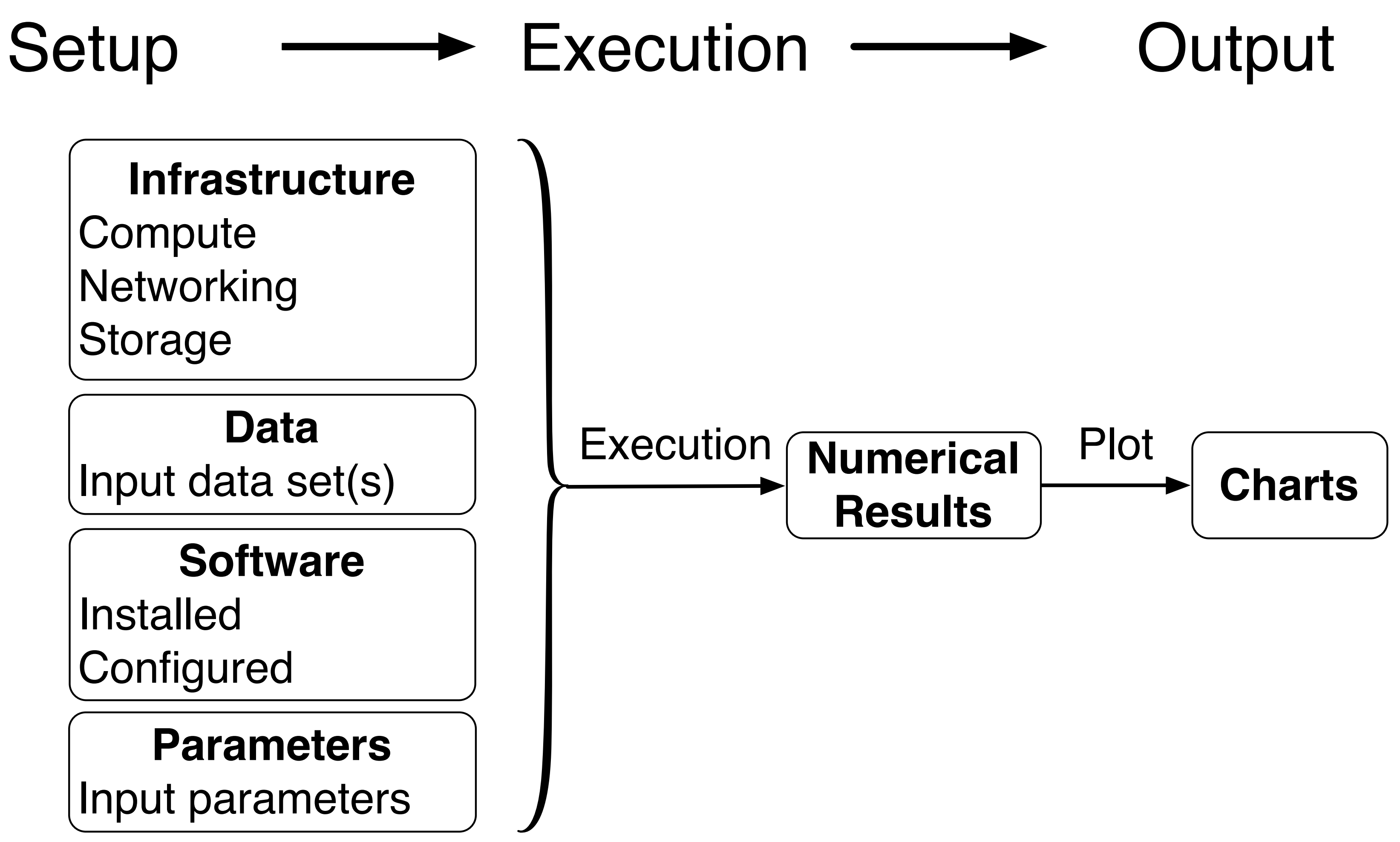
Reproducibility and Scalability in Experimentation through Cloud Computing Technologies

Jonathan Klinginsmith
Indiana University   jklingin@indiana.edu

Common Research Scenarios


That research is related to mine.
How do I reproduce that experiment?
How do I benchmark their algorithm
(or application) against mine?

Experimental reproduction needs



Setup → Execution → Output

Infrastructure
Compute
Networking
Storage

Data
Input data set(s)

Software
Installed
Configured

Parameters
Input parameters

Execution → **Numerical Results** → Plot → **Charts**

How does cloud computing help?

For reproductactability, IaaS clouds provide

Infrastructure, **Data**, and **Storage**

Machine Images   Block Storage   Object Storage

IaaS cloud artifacts can be referenced:

machine image: *ami-00001234*
block storage: *vol-00002468*
object storage: *http://object.url*

Specific instance types can be selected to meet needs:

instance types: m1.large, cc2.8xlarge, ...

Provide APIs:

Create an instance: *run-instances*
Create and **attach storage:** *create-volume; attach-volume*

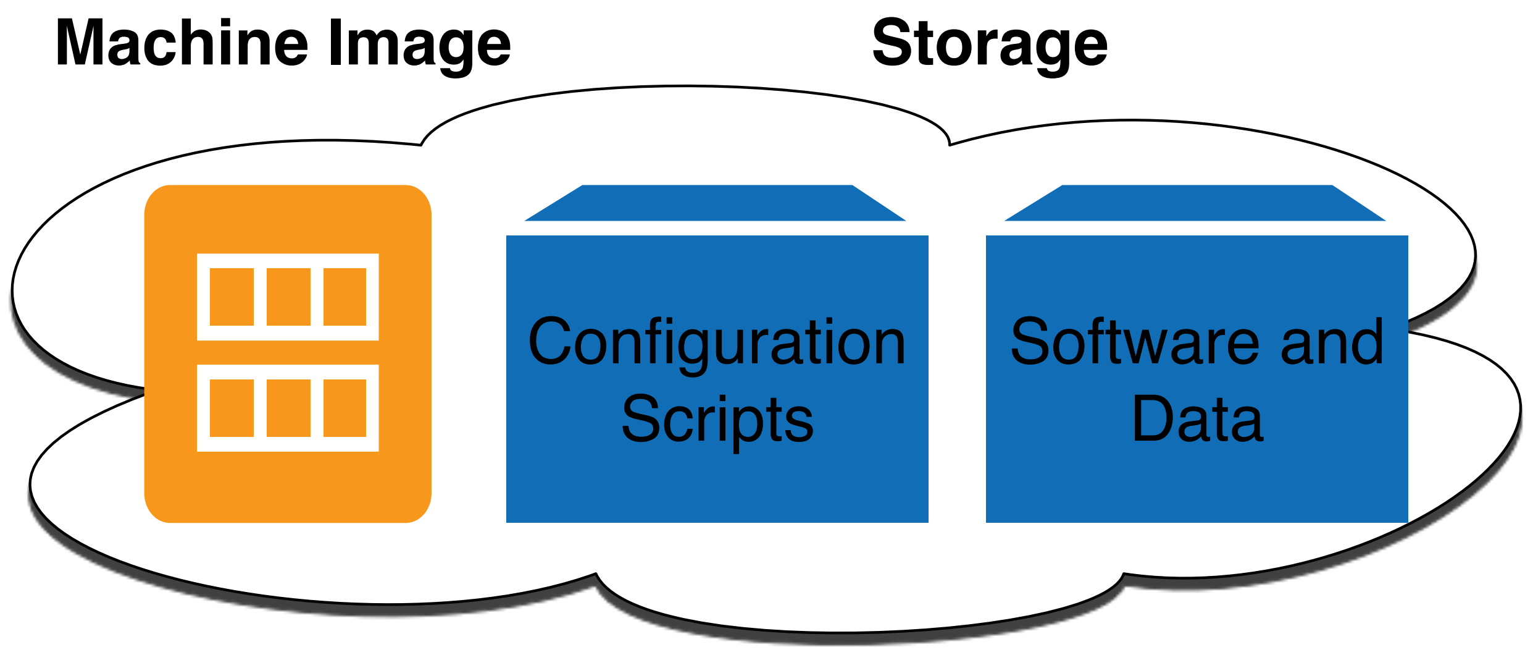
Virtual Appliances can store:

Software installed and configured
Example data set(s)
Any additional items to reproduce the experiment

A single virtual appliances cannot provide **scalability**

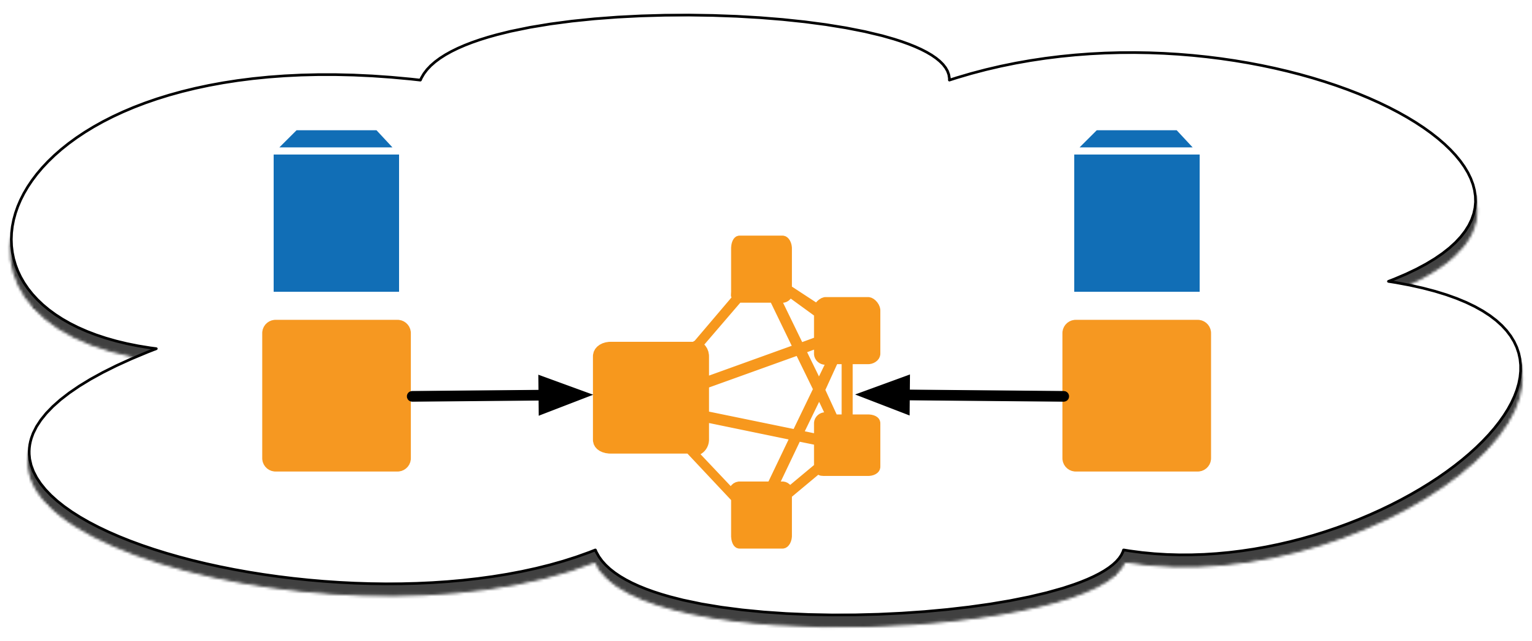
Reproducibility and scalability

Objects for reproducible, scalable applications



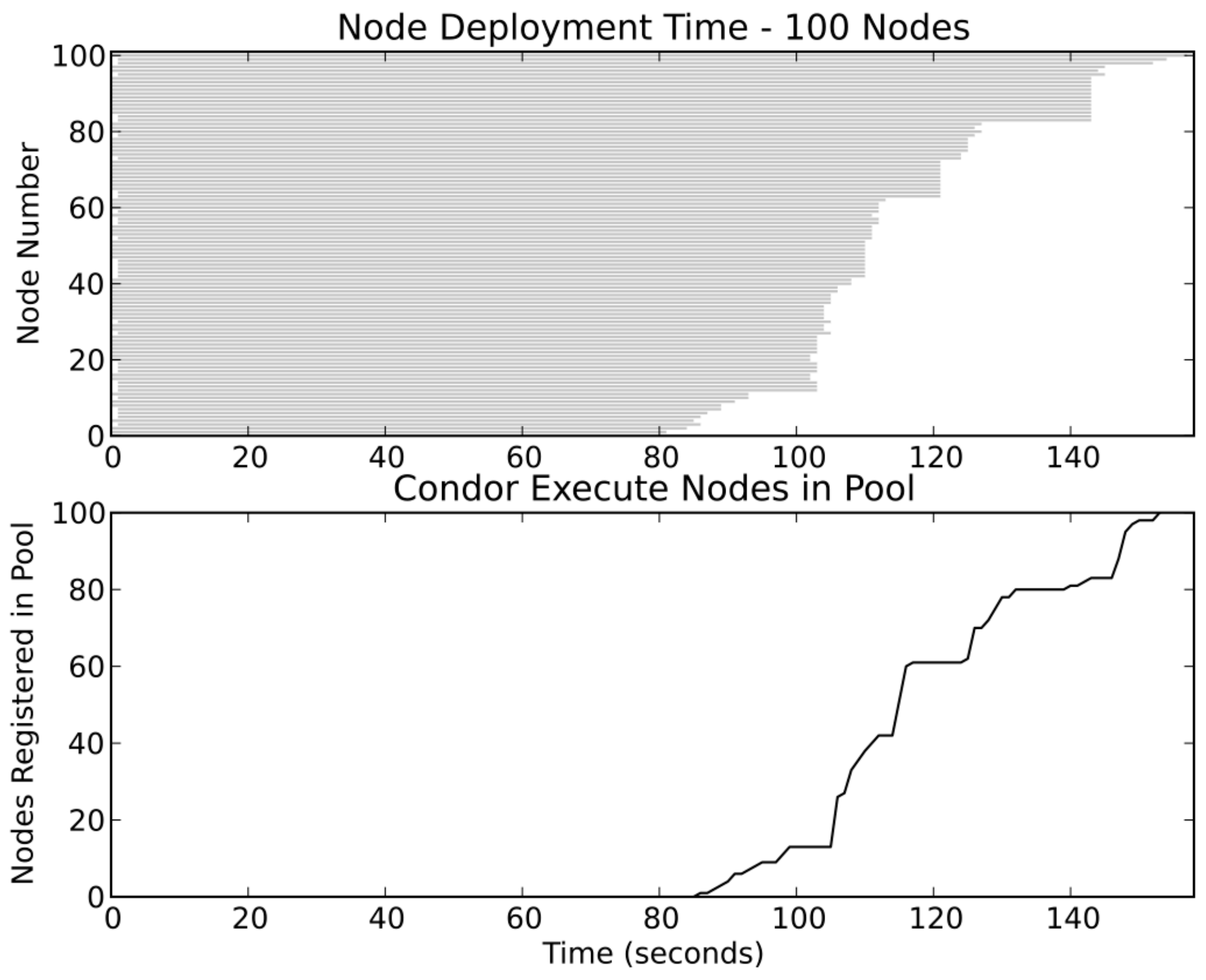
Reproduce experiments with persistent objects

Use configuration scripts to setup and run experiment



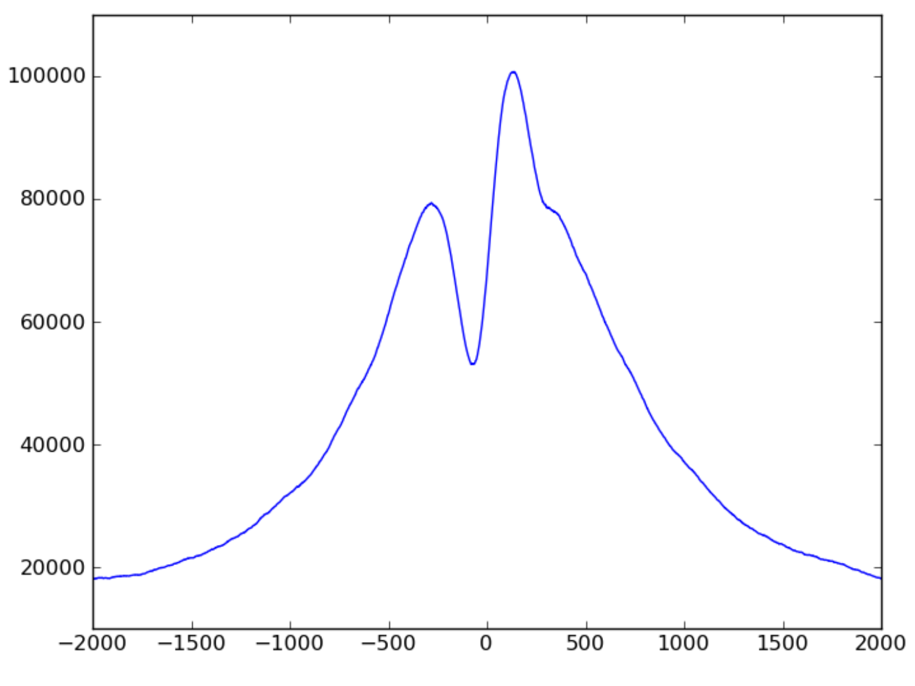
Use of persistent objects for scalability

Example demonstrating creation an 100 node Condor pool

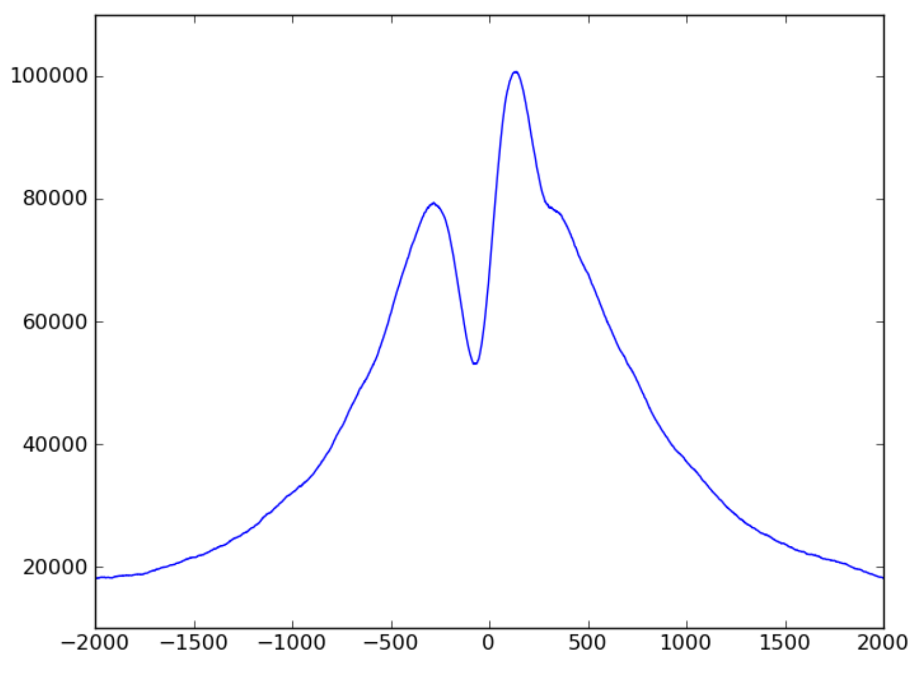


Use of persistent objects for reproducibility

Example of plots created in different clouds



TSS plot produced in AWS



TSS plot produced in FutureGrid Eucalyptus

References

S. Anders. A detailed use case: TSS plots – HTSeq v0.5.3p6 documentation. http://www.huber.embl.de/users/anders/HTSeq/doc/tss.html

B. Howe. Virtual appliances, cloud computing, and reproducible research. Computing in Science and Engineering, 14:36–41, 2012.

J. Klinginsmith, et al. Towards reproducible escience in the cloud. In Cloud Computing Technology and Science (CloudCom), pages 582–586, 2011.

D. Nurmi, et al. The eucalyptus open-source cloud-computing system. In Proc. of the 2009 9th IEEE/ ACM Int. Symp. on Cluster Computing and the Grid, pages 124–131, 2009.

T. Tannenbaum, et al. Condor – a distributed job scheduler. In Beowulf Cluster Computing with Linux. MIT Press, 2001.

Futuregrid: An experimental, high-performance grid test-bed. https://portal.futuregrid.org/

INDIANA UNIVERSITY