PRACTICAL REPRODUCIBILITY WITH CHAMELEON

Kate Keahey
Mathematics and CS Division, Argonne National Laboratory
CASE, University of Chicago
keahey@anl.gov

October 14, 2021
6th Annual CROSS Research Symposium
PRACTICAL REPRODUCIBILITY

- Can experiments be as sharable as papers are today?
- Could it be as easy to provide conditions for reviewers to repeat experiments or data analysis in a paper as it is to organize a PC meeting?
- Can I simply integrate somebody’s model into my research instead of reinventing the wheel?
- Can I have so much fun playing with somebody’s experiment that discover a new result?
- Can I develop exercises for my class based on most recent research results?

The existence of powerful open testbeds is a fundamental requirement for practical reproducibility

www.chameleoncloud.org
CHAMELEON IN A NUTSHELL

- We like to change: a testbed that adapts itself to your experimental needs
  - Deep reconfigurability (bare metal) and isolation – but also a small KVM cloud
  - power on/off, reboot, custom kernel, serial console access, etc.

- Balance: large-scale versus diverse hardware
  - Large-scale: ~15,000 cores, ~6 PB of storage originally distributed over 2 sites (UC, TACC) connected with 100G network
  - Diverse: ARMs, Atoms, FPGAs, GPUs, Corsa switches, etc.
  - CHI-in-a-Box sites at Northwestern, NCAR, IIT, and other places

- Cloud++: CHameleon Infrastructure (CHI) via mainstream cloud tech
  - Powered by OpenStack with bare metal reconfiguration (Ironic) + “special sauce”
  - Blazar contribution recognized as official OpenStack component

- Reproducibility, repeatability and sharing
  - Packaging (via Jupyter), sharing, discovering, and publishing experiments
OPEN TESTBED – BY THE NUMBERS

- 400+ Papers published
- 750+ Projects
- Over 6,000 Users
- 6+ Years Old
- 160+ Institutions
- 45 Countries

and 3 more years to grow!
TESTBED AS SHARING PLATFORM

- **Instruments held in common** are a reproducibility imperative
  - Hardware and hardware versions: >105 versions over 5 years
  - Expressive allocation

- **Sharing via cloud pattern**
  - Disk images, orchestration templates, and other artifacts
  - Chameleon >130,000 images, >35,000 orchestration templates and counting

- Testbed as “player” for environments

*Paper: “The Silver Lining”, IEEE Internet Computing 2020*
WHAT IS MISSING?

- Packaging: complete, imperative, non-transactional, integrated (literate programming)
- Get access for reproducibility
- Discover/find experiments through various channels
- Package experiment in a way that is cost-effective but also user-friendly
- Give access for reproducibility
- Share work in progress; publish and advertise completed work
PACKAGING SHARABLE EXPERIMENTS

Literate Programming with Jupyter

Experimental storytelling:
ideas/text, process/code, results

Complex Experimental containers

- Repeatability by default: Jupyter notebooks + Chameleon experimental containers
  - JupyterLab for our users: use jupyter.chameleoncloud.org with Chameleon credentials
  - Interface to the testbed in Python/bash + examples (see LCN’18: https://vimeo.com/297210055)

Paper: “A Case for Integrating Experimental Containers with Notebooks”, CloudCom 2019
Authors create a subproject with multiple short-term leases that are long enough to reproduce the experiment.

Readers click through data of a published experiment, request a daypass, and reproduce either the experiment or data analysis.
SHARING AND FINDING EXPERIMENTS

Digital publishing with Zenodo: make your experiments citable via Digital Object Identifiers (DOIs)

Trovi: sharing work in progress
- BINs to collect all the artifacts, fine-grained sharing, versioning
- Portal to browse, filter, and find interesting experiments
- Integrated with Jupyter/Chameleon, Swift, Zenodo, and github (in progress)
PARTING THOUGHTS

- Testbeds as a reproducibility fundamental
  - Public resource: instrument held in common
  - Instruments and methods enabling practical reproducibility
  - Packaging, access, and sharing

- Practical reproducibility == making reproducibility affordable
  - Time to package is important but time to repeat is critical!
  - Reproducibility marketplace

- Potential
  - Integration of research and teaching
  - Integration of Computer Science research in emergent applications
Think Big!
(with the help of a small reptile)

www.chameleoncloud.org