CHAMELEON: AN INNOVATION PLATFORM FOR REPEATABLE COMPUTER SCIENCE RESEARCH

Kate Keahey
keahey@anl.gov
University of Chicago, Argonne National Laboratory

July 16, 2021
Innovative Computing Laboratory Lunch Talk
CHAMELEON IN A NUTSHELL

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

- Balance: large-scale versus diverse hardware
  - Large-scale: ~large homogenous partition (~15,000 cores), ~6 PB of storage distributed over 2 sites (UC, TACC) connected with 100G network
  - Diverse: ARMs, Atoms, FPGAs, GPUs, Corsa switches, etc.

- Cloud++: leveraging mainstream cloud technologies
  - Powered by OpenStack with bare metal reconfiguration (Ironic) + “special sauce”
  - Blazar contribution recognized as official OpenStack component

- We live to serve: open, production testbed for Computer Science Research
  - Started in 10/2014, available since 07/2015, renewed in 10/2017, and just now!
  - Currently 5,300+ users, 700+ projects, 100+ institutions, 300+ publications
BY THE NUMBERS

300+ Papers published

45 Countries

160+ Institutions

700+ Projects

5,500+ Users

6+ Years Old

and 3+ more years to grow!
CHAMELEON HARDWARE

Chameleon Core Network
100Gbps uplink public network (each site)

Core Services
3.5PB Storage System

Core Services
0.5 PB Storage System

Haswell
Standard Cloud Unit
32 compute
Corsa Switch
x2

SkyLake
Standard Cloud Unit
32 compute
Corsa Switch
x1

CascadeLake
Standard Cloud Unit
32 compute++
x1

Heterogeneous Cloud Units
GPUs (K80, M40, P100),
FPGAs, NVMe, SSDs, IB,
ARM, Atom, low-power Xeon

Commercial Clouds via CloudBank

Chameleon Associate Sites
(Northwestern and others)

FABRIC
and other partners

Chicago
Austin

www.chameleoncloud.org
“Start with large-scale homogenous partition”
- 12 Haswell racks, each with 42 Dell R630 compute servers with dual-socket Intel Haswell processors (24 cores) & 128GB RAM and 4 Dell FX2 storage servers with 16 2TB drives each; Force10 s6000 OpenFlow-enabled switches 10Gb to hosts, 40Gb uplinks to Chameleon core network
- 3 SkyLake racks (32 nodes each); Corsa (DP2400 & DP2200), 100Gb uplinks to core network
- CascadeLake rack (32 nodes), 100Gb ulpinks to Chameleon core network
- Allocations can be an entire rack, multiple racks, nodes within a single rack or across racks (e.g., storage servers across racks forming a Hadoop cluster)

Shared infrastructure
- 3.6 (TACC) + 0.5 (UC) PB global storage, 100Gb Internet connection between sites

“Graft on heterogeneous features”
- Infiniband with SR-IOV support, High-mem, NVMe, SSDs, P100 GPUs (total of 22 nodes), RTX GPUs (40 nodes), FPGAs (4 nodes)
- ARM microservers (24) and Atom microservers (8), low-power Xeons (8)
- Coming in Phase 3: upgrading Haswells to CascadeLake and IceLake + AMD, new GPUs and FPGAs, more IB, variety of storage options, composable hardware (LiQid), P4 networking

Edge devices – towards mixed ownership model
CHI EXPERIMENTAL WORKFLOW

- discover resources
  - Fine-grained
  - Complete
  - Up-to-date
  - Versioned
  - Verifiable

- allocate resources
  - Allocatable resources: nodes, VLANs, IPs
  - Advance reservations and on-demand
  - Expressive interface
  - Isolation

- configure and interact
  - Deeply reconfigurable
  - Appliance catalog
  - Snapshotting
  - Orchestration
  - Jupyter integration
  - Networks: stitching and BYOC

- monitor
  - Hardware metrics
  - Fine-grained data
  - Aggregate
  - Archive

Authentication via federated identity,
Interfaces via GUI, CLI and python/Jupyter
VIRTUALIZATION OR CONTAINERIZATION?

- Yuyu Zhou, University of Pittsburgh
- Research: lightweight virtualization
- Testbed requirements:
  - Bare metal reconfiguration, isolation, and serial console access
  - The ability to “save your work”
  - Support for large scale experiments
  - Up-to-date hardware

*SC15 Poster: “Comparison of Virtualization and Containerization Techniques for HPC”*
DATA SCIENCE RESEARCH

- ACM Student Research Competition semi-finalists:
  - Blue Keleher, University of Maryland
  - Emily Herron, Mercer University
- Searching and image extraction in research repositories
- Testbed requirements:
  - Access to distributed storage in various configurations
  - State of the art GPUs
  - Easy to use appliances and orchestration
ADAPTIVE BITRATE VIDEO STREAMING

- Divyashri Bhat, UMass Amherst
- Research: application header based traffic engineering using P4
- Testbed requirements:
  - Distributed testbed facility
  - BYOC – the ability to write an SDN controller specific to the experiment
  - Multiple connections between distributed sites
- https://vimeo.com/297210055

*LCN’18: “Application-based QoS support with P4 and OpenFlow”*
FEDERATED LEARNING

- Zheng Chai and Yue Cheng, George Mason University
- Research: federated learning
- Testbed requirements:
  - Bare metal, ability to record network traffic precisely
  - Support for large-scale and diverse hardware
  - Powerful nodes with large memory

PRACTICAL REPRODUCIBILITY

- Towards a world where experiments are as sharable as papers today
- Goals
  - Complete packaging of an experiment – for reproducibility in the long run
  - Easy to repeat packaging – for repeatability in the short run
- Introducing variation
  - Extending impact: making it easier for others to build on your research (and cite it!)
  - Extending lifespan: making it easier to adapt for future environments (newer/different OS, updated hardware)
- Creating a market for experiments
PRACTICAL REPRODUCIBILITY

- Reproducibility baseline: sharing hardware via instruments held in common
- Clouds: sharing experimental environments
  - Disk images, orchestration templates, and other artifacts
- What is missing?
  - Telling the whole story: hardware + experimental container + experiment workflow + data analysis + story – literate programming
  - The easy button: it has to be easy to package, easy to repeat, easy to find, easy to get credit for, easy to reference, etc.
  - Nits and optimizations: declarative versus imperative, transactional versus transparent

*Paper: “The Silver Lining”, IEEE Internet Computing 2020*
REPRODUCIBILITY BUILDING BLOCKS

- Hardware: the baseline
  - >105 hardware versions over 5 years
  - Expressive allocation

- Clouds: images and orchestration
  - >130,000 images, >35,000 orchestration templates and counting
  - Portability and federation

- Packaging and repeating: integration with JupyterLab

- Share, find, publish and cite: Trovi and Zenodo
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)
  - Especially for highly distributed experiments (CHI@Edge) notebook as terminal multiplexer

Paper: “A Case for Integrating Experimental Containers with Notebooks”, CloudCom 2019
TROVI: CHAMELEON'S EXPERIMENT PORTAL

Create a new packaged experiment out of any directory of files in your Jupyter server. It is private to you unless shared. Supports sharing similar to Google Drive.

Any user with a Chameleon allocation can find and "replay" the packaged experiment.
SHARING EXPERIMENTS: PUBLICATION

Familiar research sharing ecosystem

Trovi: a digital sharing platform
- Make your experiments sharable within a community of your choice with one click
- A library of reproduced experiments from foundational papers for research and education (see e.g., Brunkan et al., “Future-Proof Your Research”, SC20 poster)

Integration with Zenodo: make your experimental artifacts citable via Digital Object Identifiers (DOIs) (export/import)

Coming soon: the Chameleon daypass!

Digital research sharing ecosystem
PARTING THOUGHTS

- Time to reproduce is critical:
  - Packaging experiments for repeatability/reproducibility matters
  - Repeating them matters even more!

- We need to create a “marketplace” for repeating research
  - Repeatability and reproducibility can be thought of as the same thing at different “price points”
  - Recognition for published digital artifacts (software, data, experiments, etc.)
  - Starting early: education is an unappreciated tool for fostering reproducible research

- Use what you have: leveraging testbeds, existing digital artifacts, frameworks, patterns, etc. has the potential to lower the ”price” of reproducibility and make it affordable

- Coming soon: Chameleon daypass and repeatability hackathon!
We’re here to change

www.chameleoncloud.org

keahey@anl.gov