



www.chameleoncloud.org

CHAMELEON: RECONFIGURABLE TESTBED FOR COMPUTER SCIENCE EXPERIMENTS

Kate Keahey

Mathematics and CS Division, Argonne National Laboratory

CASE, University of Chicago

keahey@anl.gov

September 4, 2019

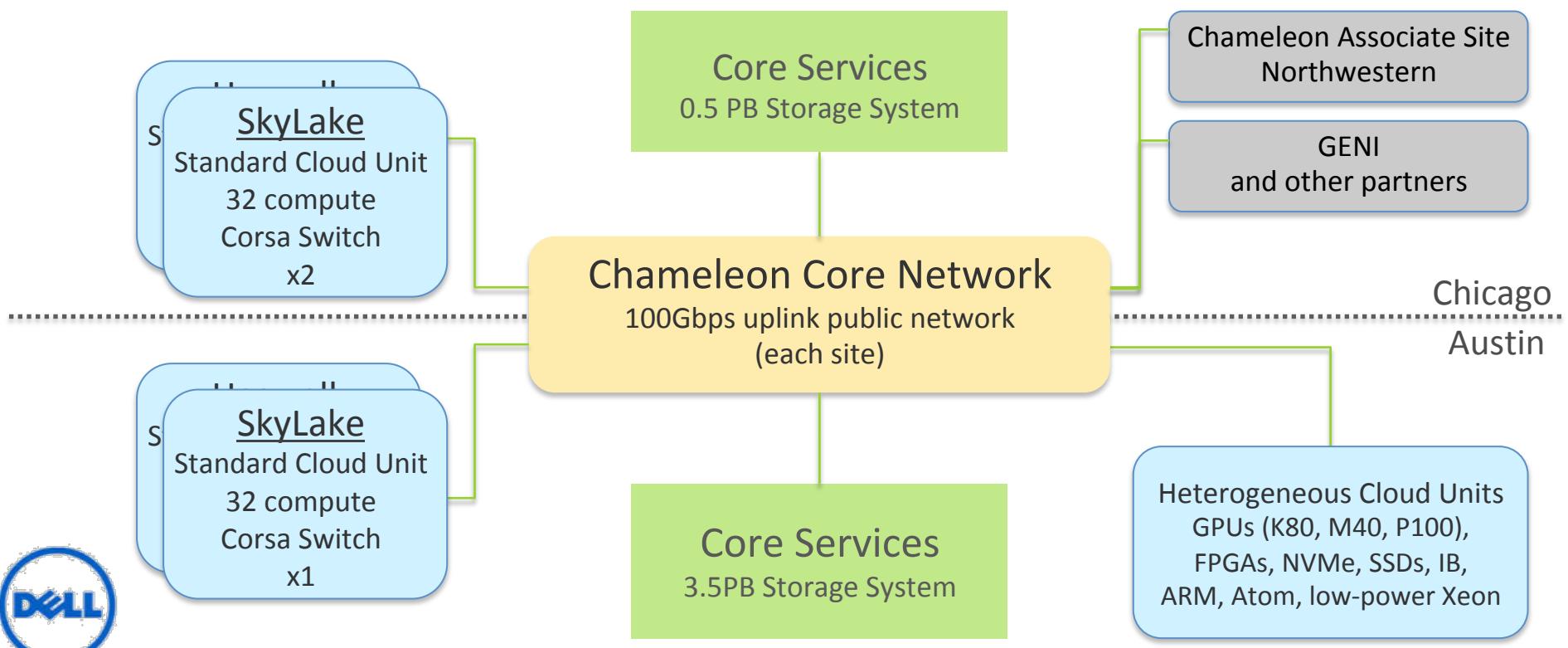
Computational Research Leadership Council meeting



CHAMELEON IN A NUTSHELL

- ▶ We like to change: testbed that adapts itself to your experimental needs
 - ▶ Deep reconfigurability (bare metal) and isolation (CHI) – but also ease of use (KVM)
 - ▶ CHI: power on/off, reboot, custom kernel, serial console access, etc.
- ▶ We want to be all things to all people: balancing large-scale and diverse
 - ▶ Large-scale: ~large homogenous partition (~15,000 cores), 5 PB of storage distributed over 2 sites (now +1!) connected with 100G network...
 - ▶ ...and diverse: ARM^s, Atoms, FPGAs, GPUs, Corsa switches, etc.
- ▶ Sustainable operations: leveraging mainstream cloud technologies
 - ▶ Powered by OpenStack with bare metal reconfiguration (Ironic) + “special sauce”
 - ▶ Chameleon team 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
 - ▶ Currently 3,500+ users, 500+ projects, 100+ institutions

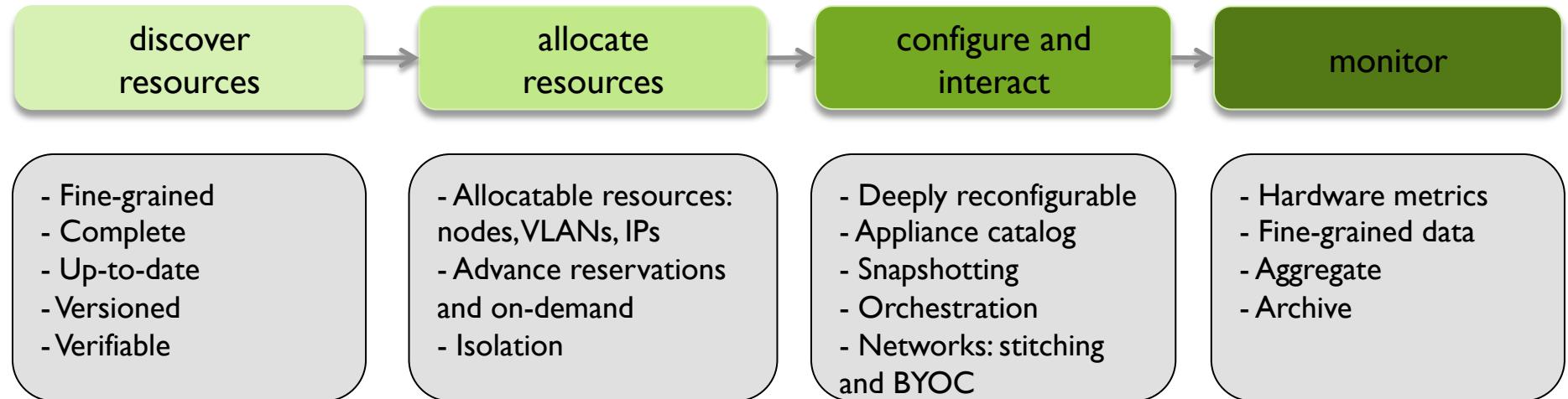
CHAMELEON HARDWARE



CHAMELEON HARDWARE (DETAILS)

- ▶ “Start with large-scale homogenous partition”
 - ▶ 12 Haswell Standard Cloud Units (48 node racks), each with 42 Dell R630 compute servers with dual-socket Intel Haswell processors (24 cores) and 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 Standard Cloud Units (32 node racks); Corsa (DP2400 & DP2200) switches, 100Gb uplinks 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 + 0.5 PB global storage, 100Gb Internet connection between sites
- ▶ “Graft on heterogeneous features”
 - ▶ Infiniband with SR-IOV support, High-mem, NVMe, SSDs, GPUs (22 nodes), FPGAs (4 nodes)
 - ▶ ARM microservers (24) and Atom microservers (8), low-power Xeons (8)
- ▶ Coming soon: more nodes (CascadeLake), and more accelerators

EXPERIMENTAL WORKFLOW



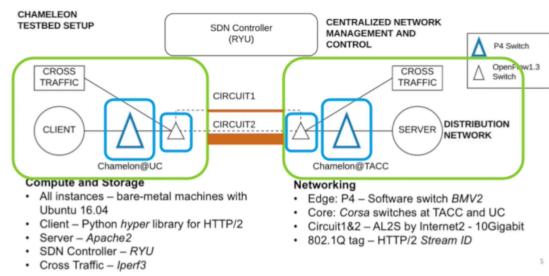
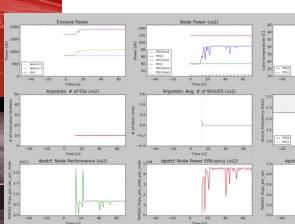
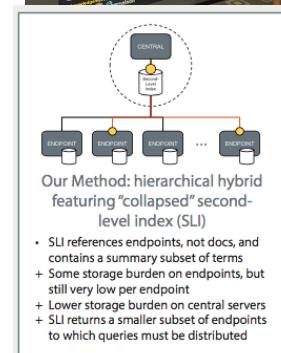
$$\text{CHI} = 65\% \text{*OpenStack} + 10\% \text{*G5K} + 25\% \text{"special sauce"}$$

LEAVING NO EXPERIMENT BEHIND...

The collage includes:

- Performance Comparison:** A bar chart titled "minFE(95% Confidence Interval, lower is better)" showing runtime in seconds versus the number of physical machines (1, 2, 4, 8, 16, 32, 64) for Bare-metal, Docker, and KVM.
- Poster Session:** Two researchers presenting at a poster session, one pointing to a chart on the wall.
- Network Routing:** A diagram of a network path from Source (Src) to Destination. "DeepRoute decides which path to allocate on" (Path 1, Path 2, or Path 3).
- Poster Abstract:** A detailed poster abstract titled "Comparison of Virtualization and Containerization Techniques for High Performance Computing" with sections on Motivation and Related Work, System Components, Evaluation Metrics, and Conclusion.
- Poster Abstract (Continued):** A continuation of the poster abstract, focusing on Evaluation Metrics and Conclusion.
- Networking Infrastructure:** A diagram of the Chameleon testbed setup, showing a SDN Controller (RYU), P4 Switches, and a Distribution Network.
- ExoGENI Slices:** A diagram showing three slices (Alice, Bob, Carol) connected via a central OpenFlow Controller, with SAFE Servers and Chameleon nodes.
- SC17 Conference:** Two researchers standing in front of a large SC17 conference banner.
- Performance Monitoring:** A grid of performance monitoring plots showing metrics like Envelope Power, Node Drawn, Node Temperature, and Application Performance over time.
- Researcher Interaction:** Researchers sitting around a table, engaged in discussion.
- Chameleon Cloud Logo:** The Chameleon logo with the URL www.chameleoncloud.org.

Supporting research projects in architecture, operating systems design, virtualization, power management, real-time analysis, security, storage systems, databases, networking, machine learning, neural networks, data science, and many others.



BEYOND THE PLATFORM: BUILDING AN ECOSYSTEM

- ▶ Interacting with hardware providers
 - ▶ Bring Your Own Hardware (BYOH)
 - ▶ CHI-in-a-Box: deploy your own Chameleon site
- ▶ Helping our user interact – with us but primarily with each other
 - ▶ Facilitating contributions of appliances, tools, and other artifacts: appliance catalog, blog as a publishing platform, and eventually notebooks
 - ▶ Integrating tools for experiment management
 - ▶ Making reproducibility easier

CHI-IN-A-BOX

- ▶ CHI-in-a-box: packaging a commodity-based testbed
 - ▶ First released in summer 2018, continuously improving
- ▶ CHI-in-a-box scenarios
 - ▶ Independent testbed: package assumes independent account/project management, portal, and support
 - ▶ Chameleon extension: join the Chameleon testbed (currently serving only selected users), and includes both user and operations support
 - ▶ Part-time extension: define and implement contribution models
 - ▶ Part-time Chameleon extension: like Chameleon extension but with the option to take the testbed offline for certain time periods (support is limited)
- ▶ Adoption
 - ▶ New Chameleon Associate Site at Northwestern since fall 2018 – new networking!
 - ▶ Two organizations working on independent testbed configuration



REPRODUCIBILITY DILEMMA

Should I invest in making my experiments repeatable?



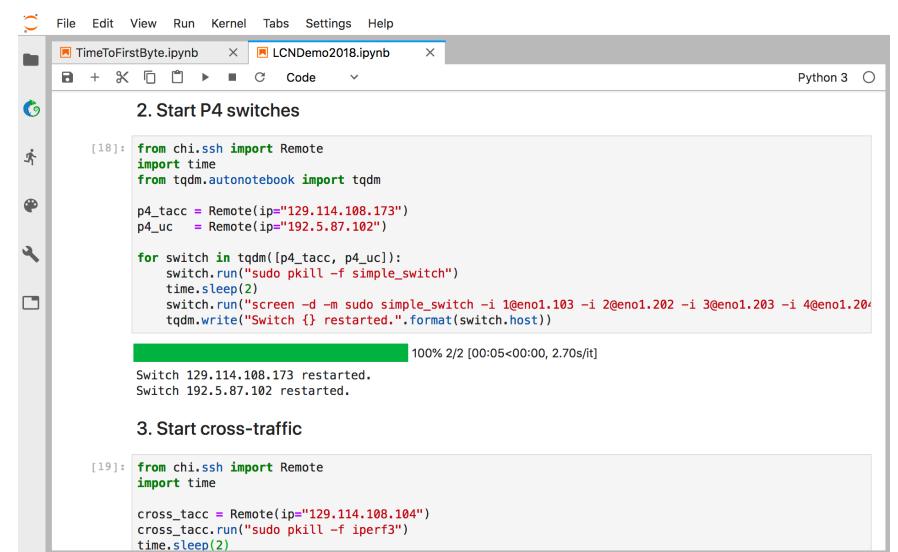
Should I invest in more new research instead?

- ▶ Reproducibility as side-effect
 - ▶ Example: Linux “history” command
- ▶ Reproducibility by default: documenting the process via interactive papers

CHAMELEON JUPYTER INTEGRATION

- ▶ Combining the ease of notebooks and the power of a shared platform
 - ▶ Storytelling with Jupyter: ideas/text, process/code, results
 - ▶ Chameleon shared experimental platform
- ▶ JupyterLab server for our users
 - ▶ Just go to jupyter.chameleoncloud.org and log in with your Chameleon credentials
- ▶ Chameleon/Jupyter integration
 - ▶ Interfaces: python and bash for all the main testbed functions
- ▶ Templates of existing experiments

Screencast of a complex experiment: <https://vimeo.com/297210055>



```
[18]: from chi.ssh import Remote
import time
from tqdm.autonotebook import tqdm

p4_tacc = Remote(ip="129.114.108.173")
p4_uc  = Remote(ip="192.5.87.102")

for switch in tqdm([p4_tacc, p4_uc]):
    switch.run("sudo pkill -f simple_switch")
    time.sleep(2)
    switch.run("screen -d -m sudo simple_switch -i 1@eno1.103 -i 2@eno1.202 -i 3@eno1.203 -i 4@eno1.204")
    tqdm.write("Switch {} restarted.".format(switch.host))

Switch 129.114.108.173 restarted.
Switch 192.5.87.102 restarted.

[19]: from chi.ssh import Remote
import time

cross_tacc = Remote(ip="129.114.108.104")
cross_tacc.run("sudo pkill -f iperf3")
time.sleep(2)
```

SHARING, EXPERIMENTING, LEVERAGING

- ▶ Sharing Jupyter notebooks in Chameleon
 - ▶ Sharing with your project members via Chameleon object storage
 - ▶ Publish to github for versioning and sharing in wider circle
 - ▶ Informally: send via email
 - ▶ Challenges: more flexible sharing policy implementation, better integration with github to support more publishing and sharing, finding relevant research, publishing
- ▶ Integration with Zenodo
 - ▶ Import from GitHub
 - ▶ Publish to Zenodo
 - ▶ Sharing platform: thin layer of discovery, indexing, filtering, etc.

PARTING THOUGHTS

- ▶ Chameleon is a rapidly evolving experimental platform
 - ▶ Originally: “Adapts to the needs of your experiment”
 - ▶ Now also: “Adapts to the needs of its community and the changing research frontier”
- ▶ Towards an Ecosystem: a meeting place of users and providers sharing resources and research
 - ▶ Testbeds are more than just experimental platforms
 - ▶ Common/shared platform is a “common denominator” that can eliminate much complexity that goes into systematic experimentation, sharing, and reproducibility
 - ▶ A critical element in building a sustainable repeatability and replicability platform