THE MANY COLORS OF CHAMELEON: BUILDING A RECONFIGURABLE TESTBED FOR SYSTEMS RESEARCH

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

May 9, 2019
Barcelona Supercomputing Center seminar
CHAMELEON IN A NUTSHELL

- **Deeply reconfigurable:** “As close as possible to having it in your lab”
  - Deep reconfigurability (bare metal) and isolation
  - Power on/off, reboot from custom kernel, serial console access, etc.
  - But also – modest KVM cloud for ease of use

- **Combining large-scale and diversity:** “Big Data, Big Compute research”
  - **Large-scale:** ~large homogenous partition (~15,000 cores), 5 PB of storage distributed over 2 sites connected with 100G network…
  - …and **diverse:** ARMs, Atoms, FPGAs, GPUs, Corsa switches, etc.
  - **Coming soon:** more storage, more accelerators

- Blueprint for a **sustainable** production testbed: “cost-effective to deploy, operate, and enhance”
  - Powered by OpenStack with bare metal reconfiguration (Ironic)
  - Chameleon team contribution recognized as official OpenStack component

- **Open, collaborative, production** testbed for **Computer Science Research**
  - Started in 10/2014, testbed available since 07/2015, renewed in 10/2017
  - Currently 3,000+ users, 450+ projects, 100+ institutions
CHAMELEON HARDWARE

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

Core Services
3.5PB Storage System

Core Services
0.5 PB Storage System

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

SkyLake Standard Cloud Unit
32 compute Corsa switch x2

SkyLake Standard Cloud Unit
32 compute Corsa switch x2

504 x86 Compute Servers
48 Dist. Storage Servers
102 Heterogeneous Servers
16 Mgt and Storage Nodes

SCUs connect to core and fully connected to each other

To GENI and other partners

Chicago

Austin
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
  ▶ 2 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
REQUIREMENTS FOR EXPERIMENTAL WORKFLOW

- Fine-grained
- Complete
- Up-to-date
- Versioned
- Verifiable
- Advance reservations & on-demand
- Isolation
- Different resource types
- Deeply reconfigurable
- Appliance catalog
- Snapshotting
- Complex Appliances
- Network Isolation
- Hardware metrics
- Fine-grained information
- Aggregate and archive
NEWEST CAPABILITIES

Networking:

- **Multi-tenant networking** allows users to provision isolated L2 VLANs and manage their own IP address space (since Fall 2017)
- **Stitching** dynamic VLANs from Chameleon to external partners (ExoGENI, ScienceDMZs) (since Fall 2017)
- VLANs + AL2S connection between UC and TACC for **100G experiments** (since Spring 2018)
- **BYOC—Bring Your Own Controller**: isolated user controlled virtual OpenFlow switches (since Summer 2018)
- And many others: new lease management features, multi-region configuration, power consumption metrics, whole disk image boot for ARM nodes, serial console access, appliances, upgrades, usability improvements, etc.
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”
EXASCALE OPERATING SYSTEMS

- Swann Perarnau, ANL
- Research: exascale operating systems
- Testbed requirements:
  - Bare metal reconfiguration
  - Boot from custom kernel with different kernel parameters
  - Fast reconfiguration, many different images, kernels,params
  - Hardware: accurate information and control over changes, performance counters, many cores
  - Access to same infrastructure for multiple collaborators

HPPAC’16 paper: “Systemwide Power Management with Argo”
CLASSIFYING CYBERSECURITY ATTACKS

- Jessie Walker & team, University of Arkansas at Pine Bluff (UAPB)
- Research: modeling and visualizing multi-stage intrusion attacks (MAS)
- Testbed requirements:
  - Easy to use OpenStack installation
  - A selection of pre-configured images
  - Access to the same infrastructure for multiple collaborators
CREATING DYNAMIC SUPERFACILITIES

- NSF CICI SAFE, Paul Ruth, RENCI-UNC Chapel Hill
- Creating trusted facilities
  - Automating trusted facility creation
  - Virtual Software Defined Exchange (SDX)
  - Secure Authorization for Federated Environments (SAFE)
- Testbed requirements
  - Creation of dynamic VLANs and wide-area circuits
  - Support for slices and network stitching
  - Managing complex deployments
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 complex deployments
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”*
BUILDING AN ECOSYSTEM

- Helping hardware providers interact
  - Bring Your Own Hardware (BYOH)
  - CHI-in-a-Box: deploy your own Chameleon site

- Helping scientists interact
  - Leveraging the common denominator
  - Integrating tools for experiment management
  - Making reproducibility easier
  - Facilitating sharing
CHI-IN-A-BOX

- CHI-in-a-box: packaging a commodity-based testbed
- CHI-in-a-box scenarios
  - **Testbed extension**: join the Chameleon testbed: generalize and package + define operations models
  - **Part-time extension**: define and implement contribution models
  - **New testbed**: generalize policies
- Understanding the support cost model
- Available since Summer 2018
- **New Associate Site at Northwestern!**
  - Nodes with 100G network cards
REPRODUCIBILITY DILEMMA

Should I invest in making my experiments repeatable? Should I invest in more new research instead?

- **Reproducibility as side-effect**: lowering the cost of repeatable research
  - Example: Linux “history” command
  - From a meandering scientific process to a recipe
- **Reproducibility by default**: documenting the process via interactive papers
REPEATABILITY MECHANISMS IN CHAMELEON

- Testbed versioning (collaboration with Grid’5000)
  - Based on representations and tools developed by G5K
  - >50 versions since public availability – and counting
  - Still working on: better firmware version management

- Appliance management
  - Configuration, versioning, publication
  - Appliance meta-data via the appliance catalog
  - Orchestration via OpenStack Heat

- Monitoring and logging

- However... the user still has to keep track of this information
KEEPING TRACK OF EXPERIMENTS

► Everything in a testbed is a recorded event
  ► The resources you used
  ► The appliance/image you deployed
  ► The monitoring information your experiment generated
  ► Plus any information you choose to share with us: e.g., “start power_exp_23” and “stop power_exp_23”

► *Experiment précis:* information about your experiment made available in a “consumable” form
REPEATABILITY: EXPERIMENT PRÉCIS

- OpenStack services
- Instance monitoring
- Infrastructure monitoring
- User events
- Experiment précis
- Store and share
- Orchestrator (Heat)
EXPERIMENT PRÉCIS IMPLEMENTATION

Come see our SC18 poster: “Reproducibility as Side-Effect”
EXPERIMENT PRÉCIS: A CASE STUDY

Experiment 1 (Default Setting)

OOME

Emit

Chameleon Events (Experiment Setup)

ReGen

Experiment Précis (Default Setting)

Experiment 2 (SmartConf)

Chameleon Events

Generate a similar experiment

Combine Results

Modified by User

Experiment Précis (SmartConf)

Based on Wang et al., Understanding and Auto-Adjusting Performance-Sensitive Configurations. ASPLOS, 2018

www.chameleoncloud.org
REPEATABILITY: EXPERIMENT PRÉCIS

- OpenStack services
- Instance monitoring
- Infrastructure monitoring
- User events

Experiment précis

www.chameleoncloud.org
ACTIVE PAPERS: WHAT DOES IT MEAN TO DOCUMENT A PROCESS?

- **Requirements**
  - Easy to work with: human readable/modifiable format
  - Integrates well with ALL aspects of experiment management
  - Bit by bit replay – allows for bit by bit modification (and introspection) as well – element of interactivity
  - Support story telling: allows you to explain your experiment design and methodology choices
  - Has a direct relationship to the actual paper that gets written
  - Can be version controlled
  - Sustainable, a popular open source choice

- **Implementation options**
  - Orchestrators: Heat, the dashboard, and OpenStack Flame
  - Notebooks: Jupyter, Nextjournal
COMBINING THE EASE OF NOTEBOOKS AND THE POWER OF A SHARED PLATFORM

- Combining Jupyter with Chameleon
  - Storytelling with Jupyter: ideas/text, process/code, results
  - Chameleon shared experimental platform
- Chameleon/Jupyter integration
  - Alternative interface
  - All the main testbed functions
  - “Hello World” template
  - Save&share via object store and github integration
- Screencast of a complex experiment
  - https://vimeo.com/297210055
JUPYTER ON CHAMELEON

Log in to Jupyter Notebook server

Generate output suitable for publishing

www.chameleoncloud.org
JUPYTER ON CHAMELEON

Username
********

Re-run experiment with modifications

Log in to Jupyter Notebook server

Compare results

What if...
PARTING THOUGHTS

- Physical environment: Chameleon is a rapidly evolving experimental platform
  - Originally: “Adapts to the needs of your experiment”
  - But also: “Adapts to the changing research frontier”

- Ecosystem: a meeting place of users 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

- Get engaged: www.chameleooncloud.org
Questions?

www.chameleoncloud.org
keahey@anl.gov