www. chameleoncloud.org

LESSONS LEARNED FROM THE CHAMELEON TESTBED

Kate Keahey

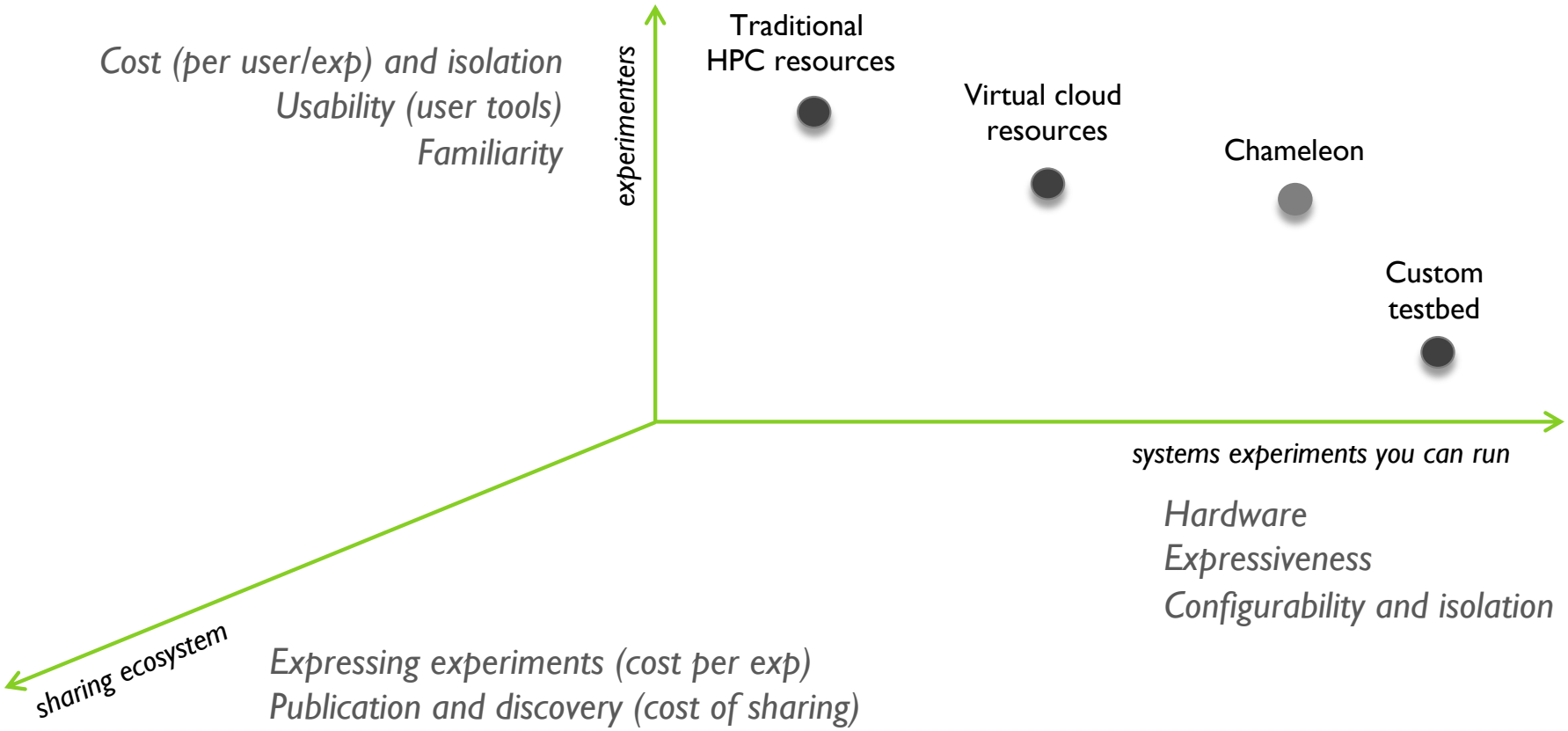
University of Chicago, Argonne National Laboratory

Jason Anderson (UC), Zhuo Zhen (UC), Pierre Riteau (StackHPC), Paul Ruth (RENCI), Dan Stanzione (TACC), Mert Cevik (RENCI), Jacob Colleran (UC), Haryadi Gunawi (UC), Cody Hammock (TACC), Joe Mambretti (Northwestern), Alexander Barnes (TACC), François Halbach (TACC), Alex Rocha (TACC), Joe Stubbs (TACC)

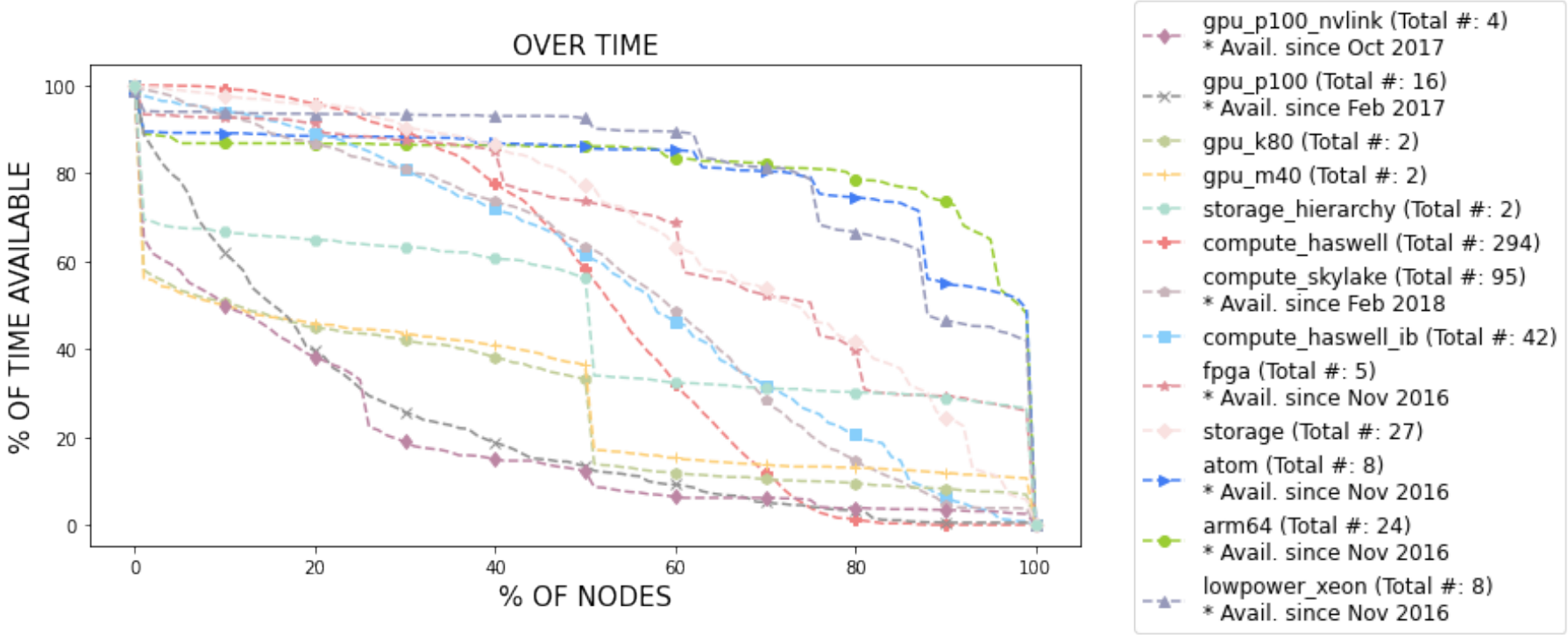
CHAMELEON IN A NUTSHELL

- 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, working on renewal now!
  - Currently 4,000+ users, 600+ projects, 100+ institutions

THE MOST EXPERIMENTS FOR THE MOST USERS

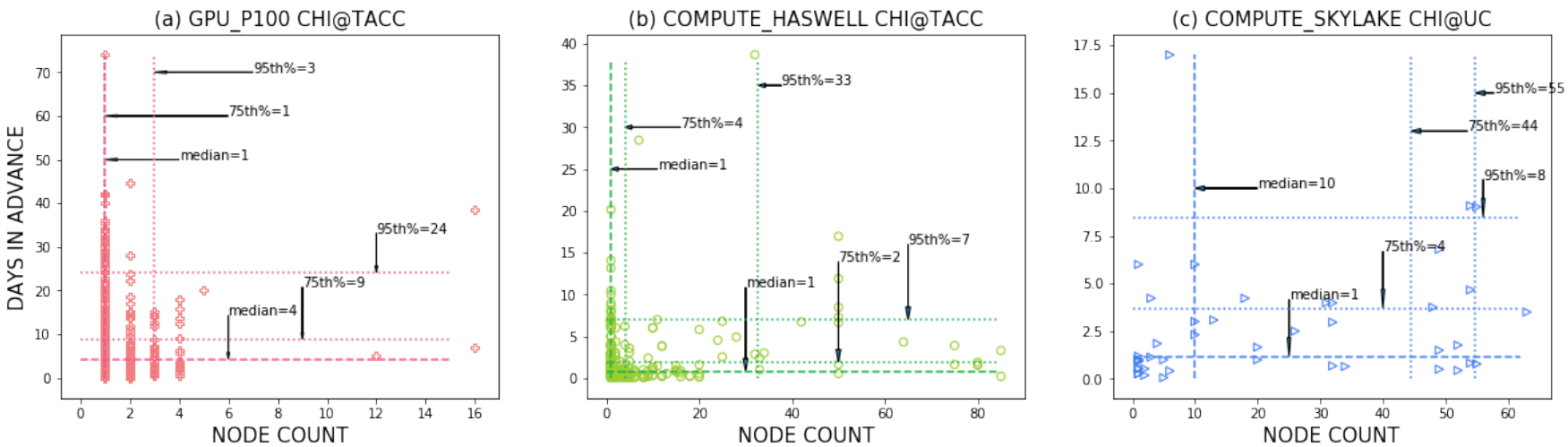


EXPERIMENTS: HARDWARE



- Largest lease: 120
- 67% single node, 5% exceed 10 nodes (11% on Haswell)

EXPERIMENTS: ALLOCATABLE RESOURCES



- Allocatable: managed in time (advance reservations, extensions) and space
- Advance reservations are critical to provide access to resources in demand
- Extensions: 5.4% usage across leases

Also see: "Managing Allocatable Resources" , CLOUD'19

EXPERIMENTS: EXPRESSIVENESS

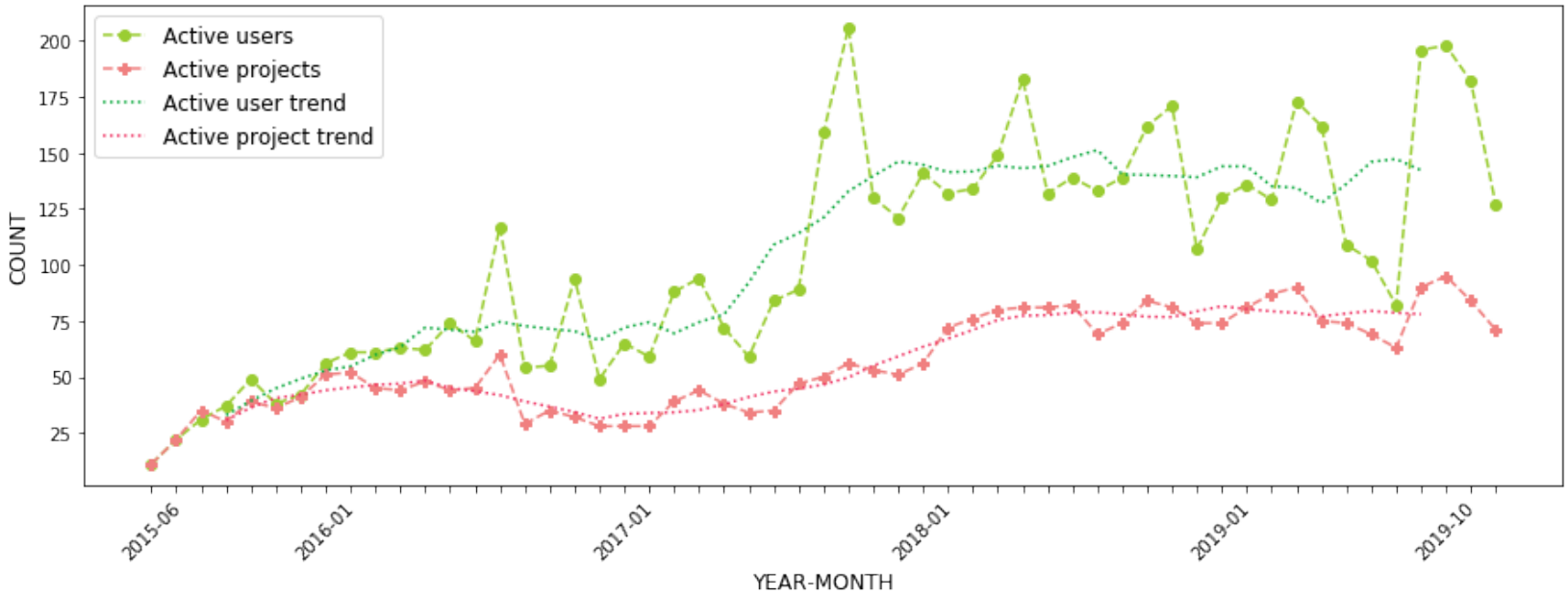
- Resources can be specified at different levels
  - Model/constraint-based: none (9.5%), single (89.24%), multiple (1.26%)
  - Hardware type (single constraint): 90.18%
  - Node UID (single constraint): 3.38% (18.45% for leases made 7 days in advance)
- Separation of allocation and configuration
  - 20.07% allocations had more than 1 instance deployed (max of 12)
- Network stitching (ExoGENI): 22 (8%) projects created 920 stitched links
- Bring Your Own Controller (BYOC): 11 (4%) projects
- Orchestration (Heat): 94 (2017), 155 (2018), and 405 (2019) deployments
- Automated deployment: surprisingly little use

EXPERIMENTERS: COST

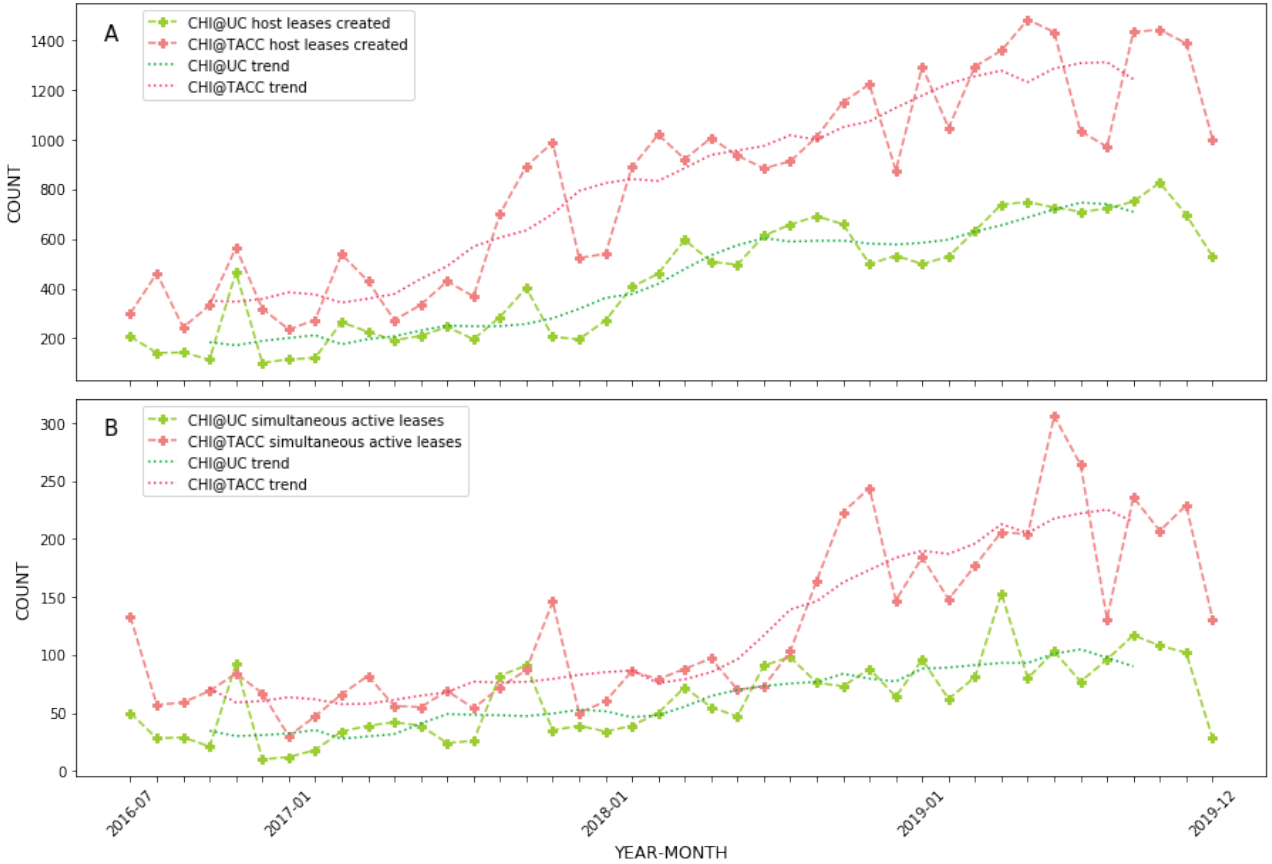
- Support cost
  - Average of 13 help desk tickets per week, less than one ticket per user
  - Heavily leveraging smoke tests, live monitoring, and automated remediation
- Working with mainstream open source project (OpenStack)
  - Familiar interfaces: 858 deployments across 441 organizations in 63 countries
  - Transferable skills
  - Working with large community (~8,400 total contributors, ~6,000 reviewing code)
  - New features: whole disk image boot, support for non x86, multi-tenant networking
  - Access to existing documentation and support systems
  - Opportunity to contribute (though at a cost): Blazar as OpenStack component

Chameleon expresses capabilities needed for CS research in terms of a mainstream cloud functionality (CHI-in-a-Box)

EXPERIMENTERS: ACTIVE USERS

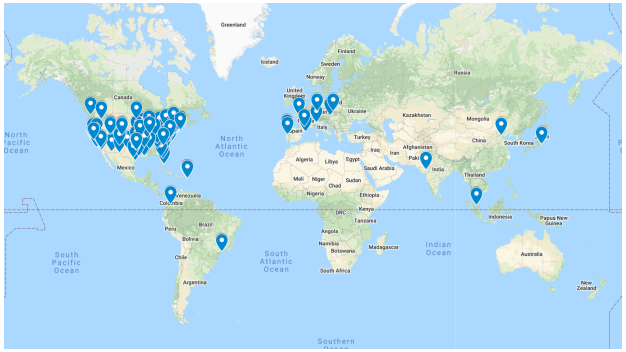


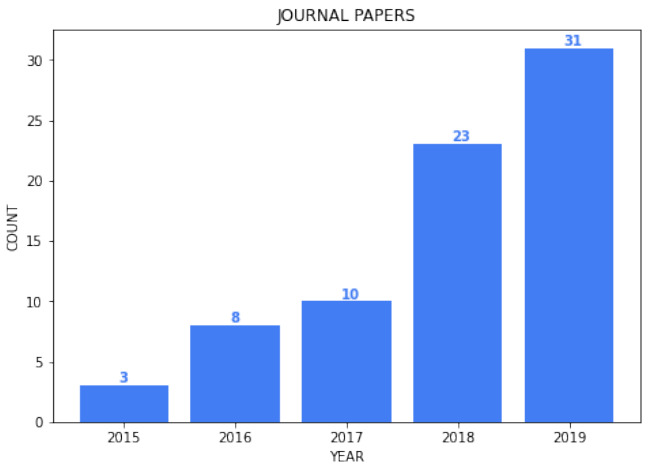
EXPERIMENTERS: ACTIVE LEASES



EXPERIMENTERS: COMMUNITY

- Institutions: 168 (11 MSI, 19 EPSCOR)
- Geography (US): 40 states + Puerto Rico
- Funding source: NSF (also DOE, DARPA, others)
- Research versus education
  - Education: 45/513 projects use ~9% of total time
  - Research: similar average usage
- Publications: 275/75 overall /journal
- Field of science
  - 12% (non CS), 10% (security), 17% (ML), 8% (Edge)
- Renewals: ~75% of eligible projects sought renewal, 33 renewed > 5 times



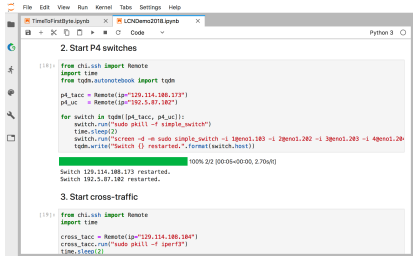


SHARING EXPERIMENTS

- Testbeds/clouds lead to the creation of compatible digital artifacts that package an experiment
  - In Chameleon: ~120,000 images and ~31,000 orchestration templates
- Elements of reproducibility support in Chameleon
  - Testbed versioning
  - Image versioning
  - Orchestration
  - Experiment Précis (Linux history analogue)
- How do we tie them all together?

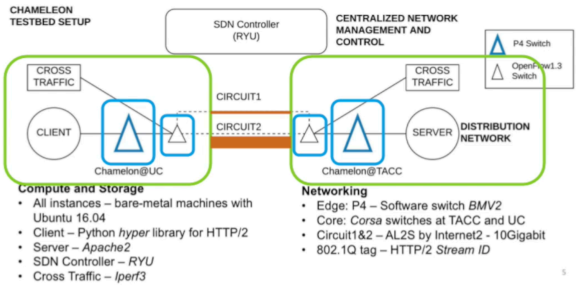
SHARING EXPERIMENTS: PACKAGING

Jupyter Notebooks



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

+



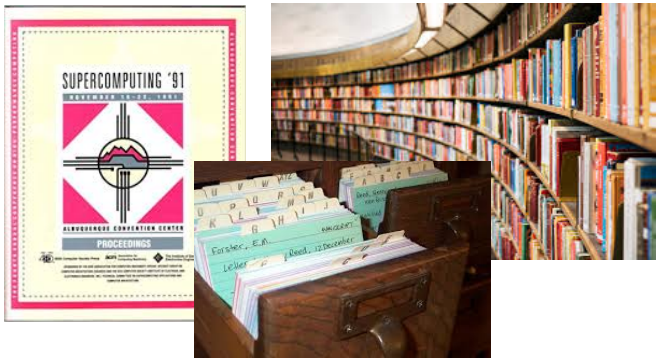
Complex Experimental containers

- 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)
  - Named containers: your experimental process goes here

Also see: "A Case for Integrating Experimental Containers with Notebooks", CloudCom 2019

SHARING EXPERIMENTS: PUBLICATION

Familiar research sharing ecosystem



Digital research sharing ecosystem

Chameleon jupyterhub

?

- Digital publishing with Zenodo: make your experimental artifacts citable via Digital Object Identifiers (DOIs)
- Integration with Zenodo
  - Export: make your research citable and discoverable
  - Import: access a wealth of digital research artifacts already published
- Towards making research findable: the digital sharing platform

zenodo

PARTING THOUGHTS

- Chameleon expresses capabilities needed for CS research in terms of a mainstream cloud functionality -- OpenStack
  - Our paper discusses the extensions and augmentations to support our use case
  - Practical delivery: CHI-in-a-Box – packaging of the CHameleon Infrastructure
- Experimental testbeds: opportunity for sharing
  - The most experiments for the most experimenters
  - Opportunity for the support of efficient sharing of experiments
- Chasing the research frontier: the functionality of any scientific instrument has to follow the emergent opportunities in the science they serve – development-driven operations

We're here to change

www.chameleoncloud.org

keahey@anl.gov