www. chameleoncloud.org

Infrastructure for New Ideas

Kate Keahey

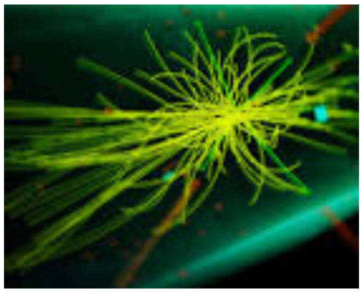
University of Chicago / Argonne National Laboratory

keahey@uchicago.edu

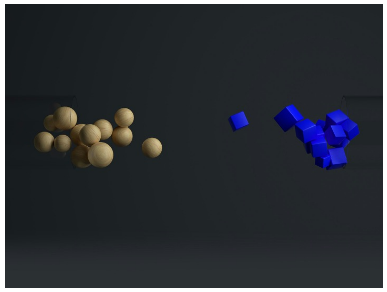
CARLA, 09/25/25

MAKING SCIENCE


Human genome sequencing


High Energy Physics


Drug discovery


Materials discovery

high performance computing data science

Artificial Intelligence machine learning

security energy efficiency performance management

storage systems virtualization networking databases


Chameleon
800+
Papers
published
1,300+
Unique
projects
13,000+
Users
10

INFRASTRUCTURE FOR NEW IDEAS

- Support a **broad** set of experiments
  - Diverse: architectures, accelerators, storage, interconnects and networks
  - From large to small: scale versus diversity trade-off
  - Wide interfaces, distributed capability, integration with other testbeds

- **Deep**ly reconfigurable
  - Capabilities: power on/off, custom kernel boot, serial console access, firmware change, etc.
  - A spectrum of reconfigurability options: bare metal, virtualization, containerization
- Supporting and innovating experimental **methodology**
  - Experiment packaging and sharing
  - Practical reproducibility
- **Cost-effective** yet **evolving**
  - Solid base in mainstream open source capabilities
  - Capable of evolution

openstack
CLOUD SOFTWARE

CHAMELEON HARDWARE

CHI@UC
Skylake, CascadeLake, IceLake, AMD nodes, and GigaIO
GPUs (A100, V100, RTX6000), FPGAs (Xilinx Alveo U280)
Storage (NVMe SSD, NVDIMM)
Network (Corsa SDN, 25G Ethernet, 200G InfiniBand)

CHI@Edge
Raspberry Pi, Jetson Nano, Jetson Xavier NX, AGX Orin
+ User-added devices

Chicago

Austin

CHI@TACC
Haswell, SkyLake, CascadeLake, Fugaku (ARM64 + HBM),
AMD nodes, and LIQID/GigaIO disaggregated hardware
GPU (K80, M40, P100, MI100, H100s), FPGAs (Altera)
Storage (NVMe, SSD, NVDIMM)
Network (Corsa SDN, 25G Ethernet, 200G InfiniBand)

Storage
~0.5 PB

Network
100Gbps public network

Storage
~3.5PB

Commercial Clouds via **CloudBank**

FABRIC, FAB and other partners

Chameleon Volunteer Sites

CHI@NCAR

NCAR

EXPERIMENT STRUCTURE

Create **experimental environment**

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

allocate resources
- Allocatable resources: nodes, VLANs, IPs
- Advance reservations and on-demand
- Non-fungible
- Isolation

configure environment
- Bare metal, KVM, containers
- Image catalog
- Snapshotting
- Network stitching
- Orchestration

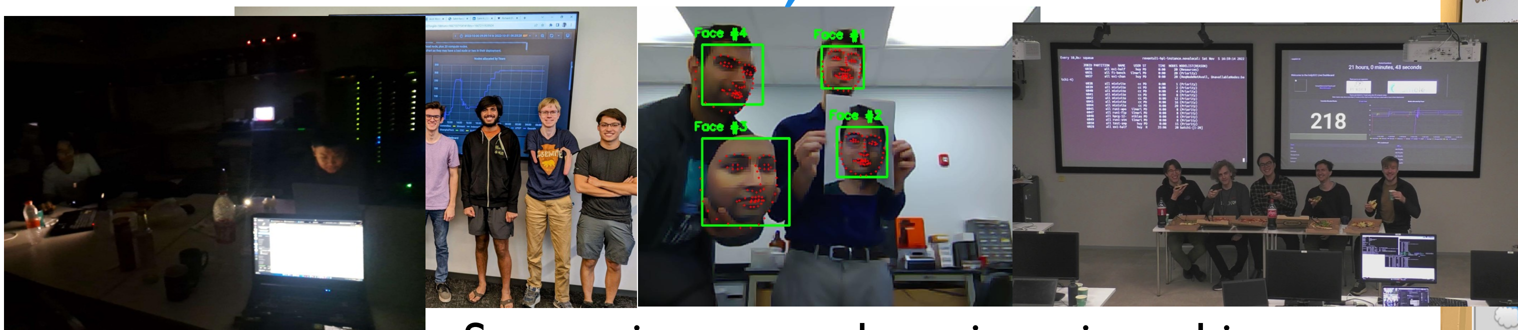
Authentication via federated identity, accessed via GUI, CLI, and python-chi

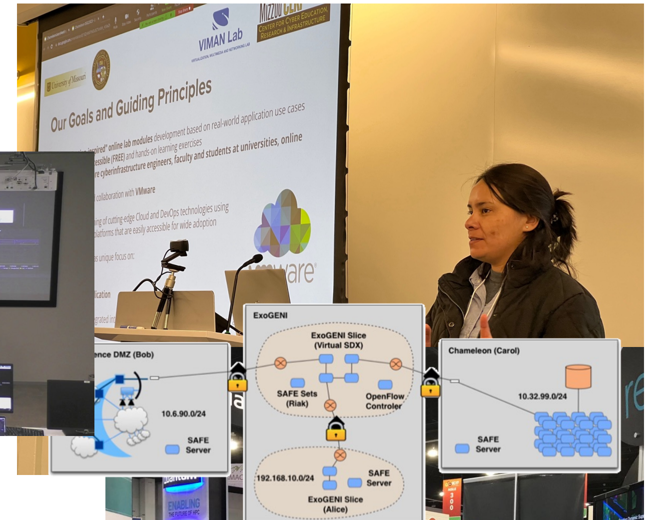
Run **experiment body**

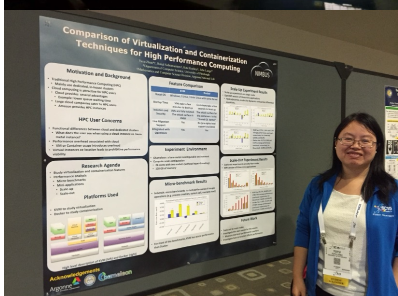
analyze and present results

Paper: "Lessons Learned from the Chameleon Testbed", USENIX ATC 2020

NOT JUST A TESTBED, A COMMUNITY





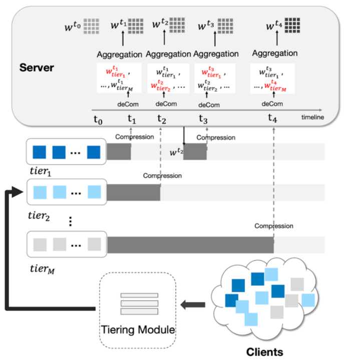


Supporting research projects in architecture, operating systems design, virtualization, power management, real-time analysis, security, storage systems, databases, networking, machine learning, neural networks, data science, and many others.

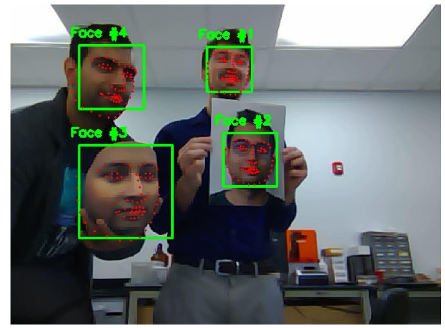
Chameleon www. chameleoncloud.org

Check out user experiment stories on our blog:
https://www.chameleoncloud.org/blog/category/user-experiments/

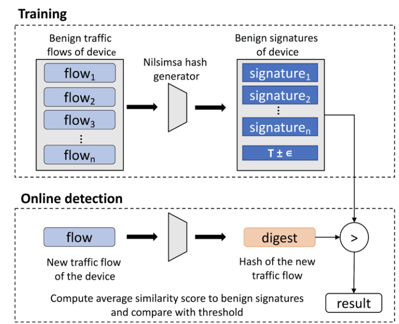
FROM CLOUD TO EDGE WITH CHAMELEON



federated learning



biometrics



network traffic fingerprinting for IoT devices

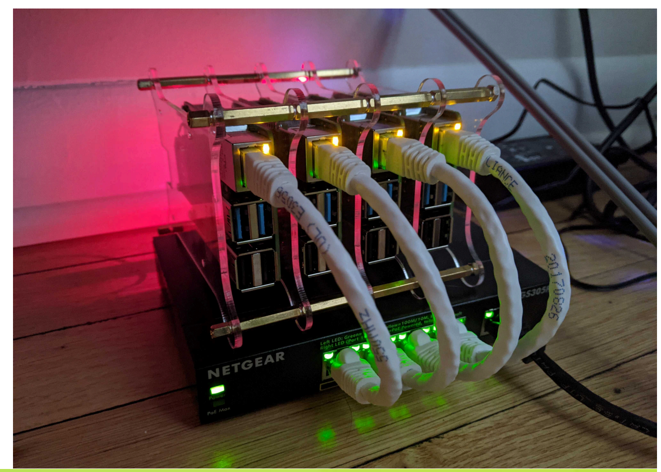
- Increasingly more Chameleon project applications working on IoT/edge
- Simulation/emulation don't always provide the answer: What are the impacts of this approach on power management on edge device? How will the performance transfer to edge? Can we measure the impact of distribution/networking for edge/cloud applications?
- **Goal: "realistic edge to cloud experiments from one Jupyter notebook"**

PROGRAMMABLE WITH CHI@EDGE

A lot like a cloud!
All the features we know and love – but for edge!
"Edge to cloud from one Jupyter notebook."

Not at all like a cloud!
Location, location, location!
IoT: cameras, actuators, SDRs!
Not server-class!
And many other challenges!

- CHI@Edge: all the features you love in CHI, plus:
  - Reconfiguration through non-prescriptive **container deployment** via OpenStack interfaces (using K3 under the covers)
  - Support for "standard" **IoT peripherals** (camera, GPIO, serial, etc.) + easy for you to add support for your own peripherals
  - **Bring Your Own Device (BYOD): Mixed ownership** model via an SDK with devices, virtual site, and **restricted sharing** – building on OpenBalena

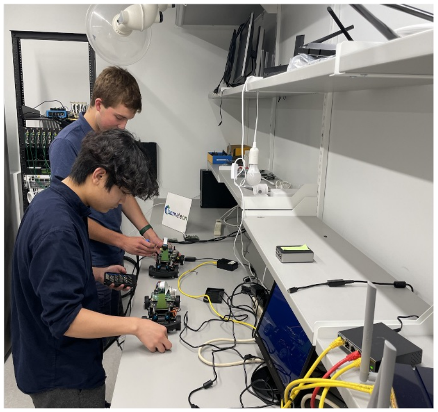


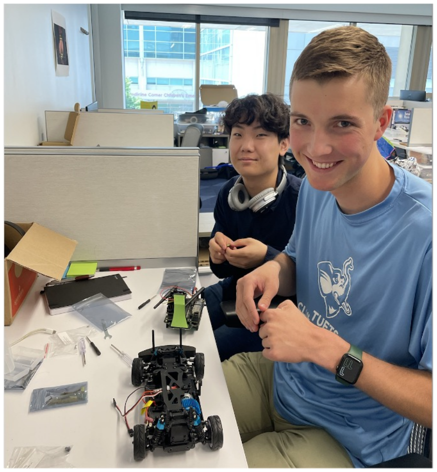
Paper: "CHI@Edge: Supporting Experimentation in the Edge to Cloud Continuum", PEARC'25

AUTOLEARN

- Chameleon notebooks based on the DonkeyCar package
- Students learn in three stages:
  - Data collection – actual/simulator – edge to cloud
  - Model training in the cloud
  - Verification via autonomous driving – actual/simulator – edge to cloud
- Supports different emphasis in teaching
  - Introduction to engineering might emphasize driving the actual car
  - Machine learning focus might use the simulator
- Individual exploration:
  - E.g., digital twin combining simulator and experimental driving

Paper: "AutoLearn: Learning in the Edge to Cloud Continuum", EduHPC'23





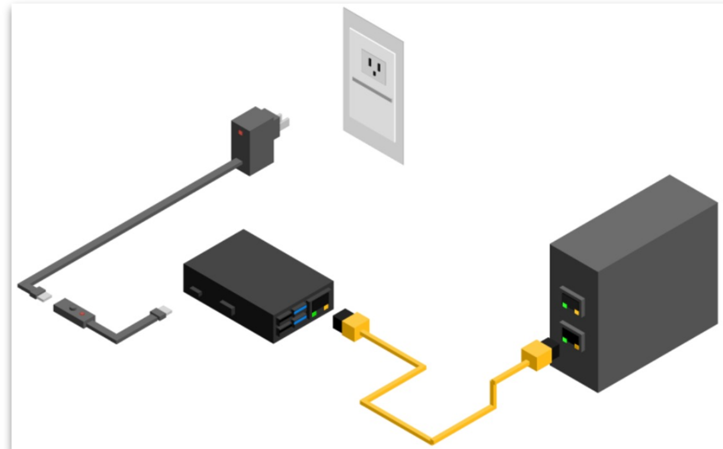
REU 2023 students working on hardware setup for autonomous vehicles

FROM EXPLORATION TO OBSERVATION: THE FLOTO PROJECT CASE STUDY

- Why broadband monitoring?
  - Technical questions: what happens in conditions of oversubscription?
  - Policy questions: can we characterize the "digital divide" in our society?
  - Modeling questions: what assumptions about broadband are realistic?
- Measuring broadband – different approaches/applications depending on context, objective, use case, etc.
  - Netrics: open-source library of standard network diagnostic tools (ndt7, speedtest, ping, traceroute, etc.) for continuous, longitudinal network measurement
  - Others: e.g., residential versus rural broadband and other use cases
- **Approach:** connect a "measurement box" to the router and run tests
- **Can we use CHI@Edge as a large observatory instrument for broadband monitoring?**
- Collaboration with Nick Feamster & his UChicago team

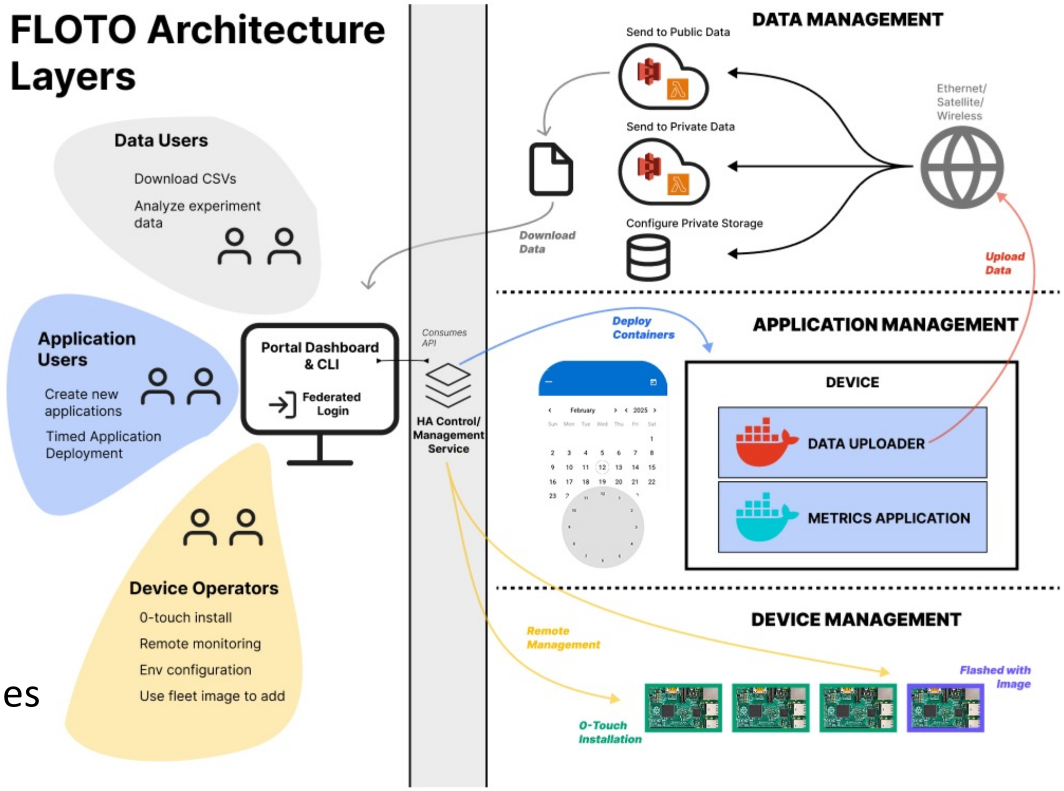
THE DEVICES

- Raspberry Pi 4 (8GB)
- Additional Components
  - MicroSD Cards (32GB)
  - CAT 6 Ethernet Cable
  - Power Cord
- Optional: PoE+ HATs to enable deployment in locations with scarce power sources
- Inventory: 1,000 devices
- Allocations via a device request form: https://floto.cs.uchicago.edu/deployment/apply-for-floto-devices/



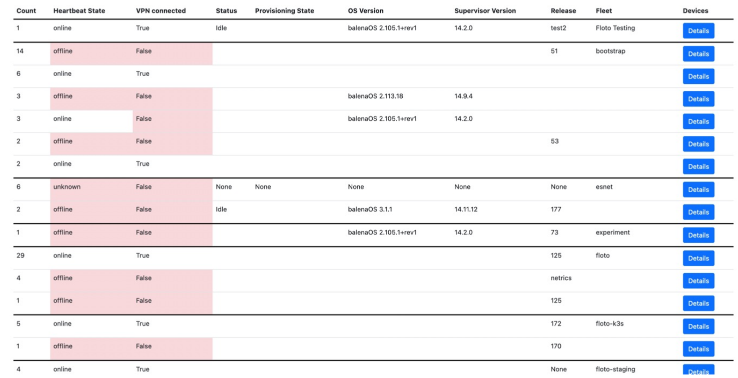
IBIS: A SENSING SUPERCOMPUTER

- Device operators
  - Ease of use vs control trade-off
  - User operator vs centralized
- Application users
  - Applications composed of several functions
  - Application configuration
- Data users
  - Sharing versus privacy trade-off
  - Established community data pipelines versus new sharing methods

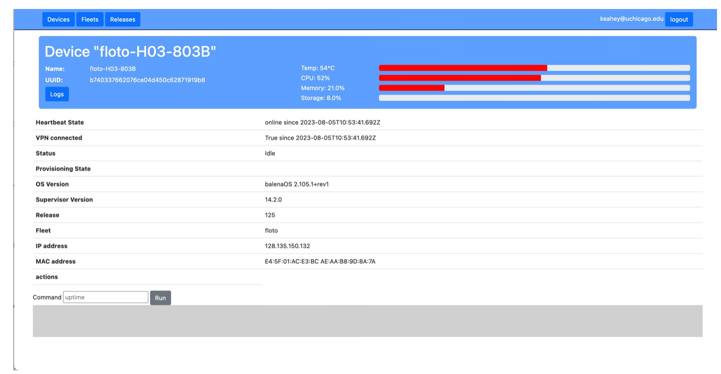


DEVICE MANAGEMENT LAYER

- Onboard, offboard, and repurpose devices
- Devices self-enroll
  - 0 touch device enrollment (after imaging)
  - Alternatively, flash with our image to enroll your own device
- Configuration management
  - Update and deploy without physical access, stateless, vetted images, includes software and device configuration, can be pinned to releases
  - OpenBalena + “special sauce”
- Robust remote management features
  - View status and statistics, create and manage deployments, trigger appropriate actions (e.g., send mail), dashboard and CLI interfaces
- HA control plane, federated identity login, etc.

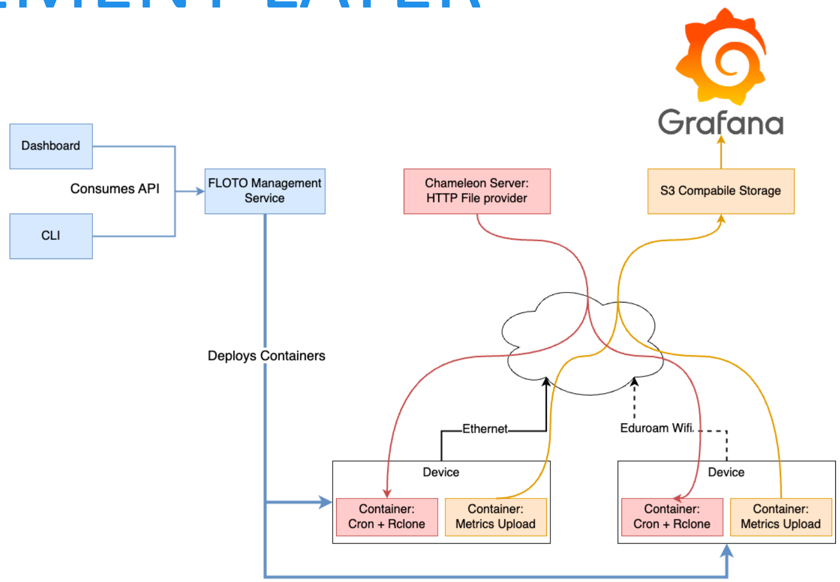


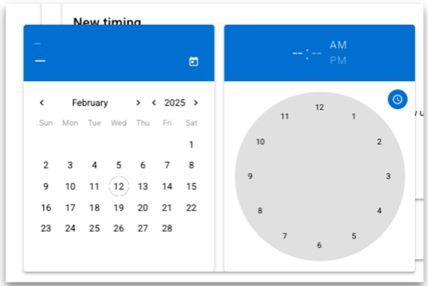
Count	Heartbeat State	VPN connected	Status	Provisioning State	OS Version	Supervisor Version	Release	Fleet	Devices
1	online	True	Idle		balenaOS 2.105.1+rev1	14.2.0	test2	Floto Testing	Details
14	offline	False					51	bootstrap	Details
6	online	True							Details
3	offline	False			balenaOS 2.113.18	14.9.4			Details
3	online	False			balenaOS 2.105.1+rev1	14.2.0			Details
2	offline	False					53		Details
2	online	True							Details
6	unknown	False	None	None	None	None	None	esnet	Details
2	offline	False	Idle		balenaOS 3.1.1	14.11.12	177		Details
1	offline	False			balenaOS 2.105.1+rev1	14.2.0	73	experiment	Details
29	online	True					125	floto	Details
4	offline	False					netrics		Details
1	offline	False					125		Details
5	online	True					172	floto-k3s	Details
1	offline	False					170		Details
4	online	True					None	floto-staging	Details



APPLICATION MANAGEMENT LAYER

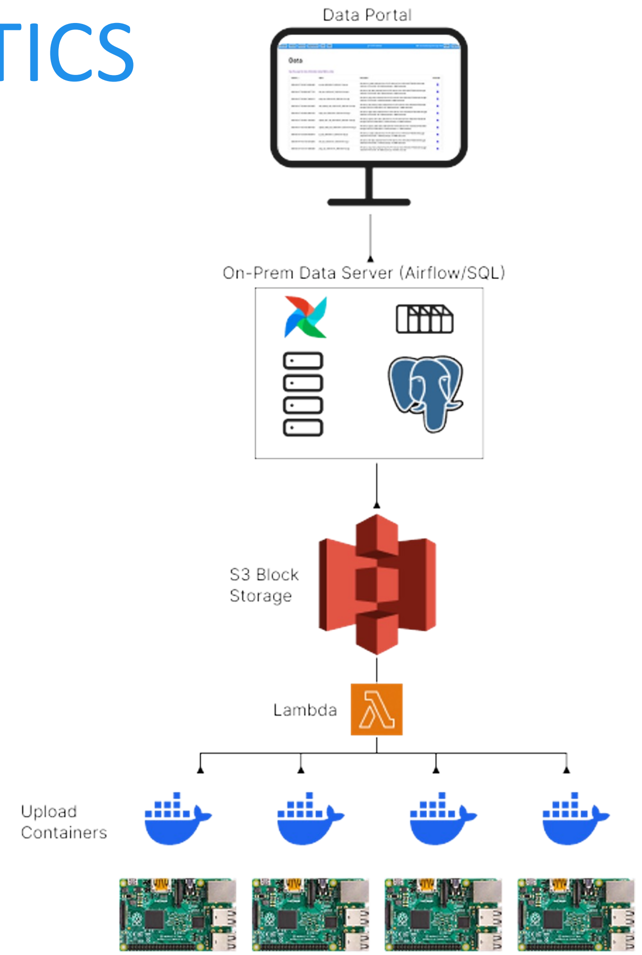
- Supports deployment of applications on device fleets via a system container
- Applications are packaged as Docker containers and reviewed
- Users can reserve overlapping or non-overlapping timeslots for application deployment so as not to conflict with other deployments
- Generic data streaming implemented as a “system application”
- Multi-container applications deployed via docker-compose syntax





DATA COLLECTION AND ANALYTICS

- Applications gather data locally on device
- Upload containers send it to cloud/central storage
  - Default processing pipeline (Netrics Data): Netrics Uploader → AWS Lambda & S3 → on-prem Airflow & PostgreSQL → FLOTO Data Portal
  - User-defined pipeline (new broadband test): FLOTO Uploader/Custom → Chameleon Block Storage → Compute processing → Shared as artifact on Chameleon
- Filter and share securely
  - Filtering: Remove sensitive data before publication
  - Public portal dataset (default): download CSV files
  - Private storage: FLOTO-managed or user-controlled



Paper: "Discovery Testbed: An Observational Instrument for Broadband Research", eScience'23

INSTRUMENT ADAPTABILITY

What knobs can I turn on this instrument?

- Deployment scope: deploy the devices in a different area
- Application: adapting "sensing abilities" programmatically
- Hardware: combine devices with different IoT gadgets (e.g., GPS)
- Data aggregation: different methods for different applications
- Data: ask different questions of the data

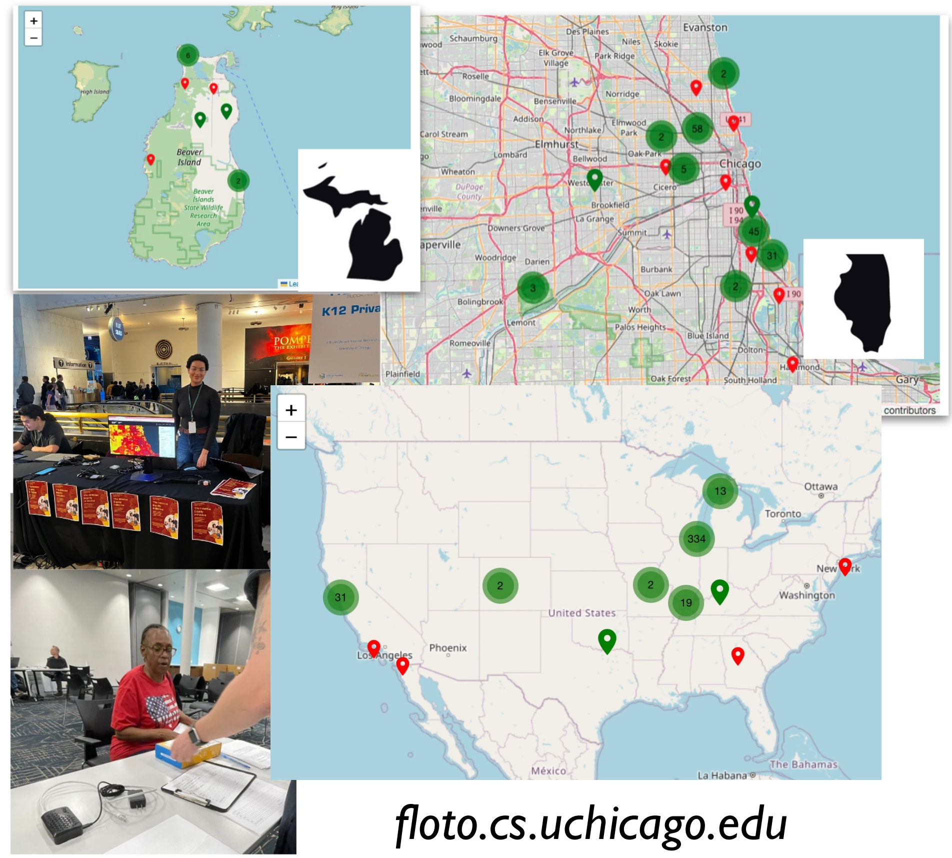
FLOTO: DEPLOY DEVICES IN DIFFERENT AREAS

~500 devices deployed across multiple states
Notable deployments:

- Chicago (180+ devices)
- Milwaukee (200+ devices)
- Marion County, IL; Beaver Island, MI -- and others

As a distributed community, we rely on trust and deep partnerships to bring infrastructure where it is needed most

- Building trust with communities
- Managing devices remotely (with many participants)
- Coordinating large-scale distribution



floto.cs.uchicago.edu

FLOTO: RUN A DIFFERENT APPLICATION

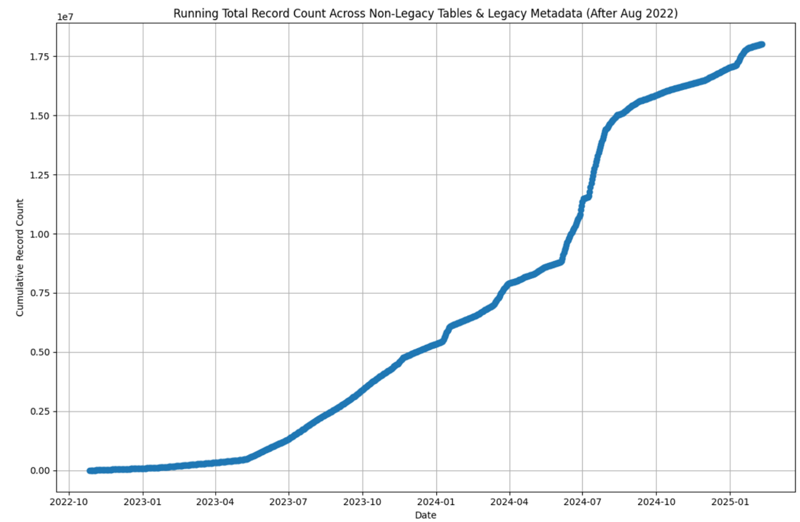
Applications Deployed on FLOTO to Date:

Each application provides different methods for broadband measurement depending on research interest

- **Netrics**: Broadband performance measurements to study access networks
- **RADAR Toolkit**: QoE measurements for telehealth applications
- **NetUnicorn**: Data pipeline experiments
- **Georgia Tech**: IPv6 Performance Studies
- **M-Lab**: Measurement Swiss Army Knife (MSAK) integration
- **ARA**: Monitoring 5G wireless performance in rural areas

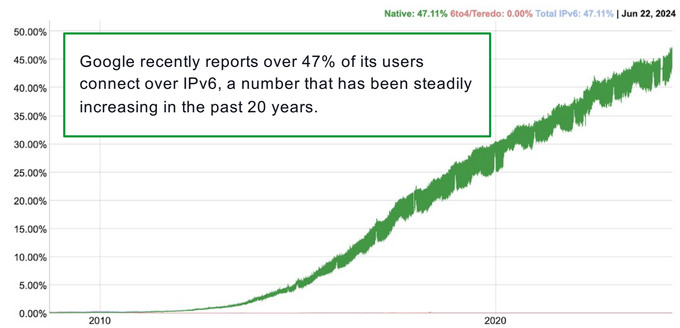
FLOTO: MINE THE DATASET

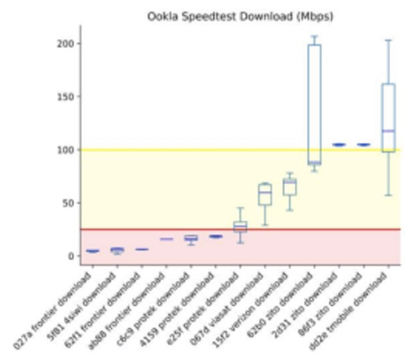
- ~25M million measurements collected since Oct. 2022
- What Measurements? Time series speed tests, latency, DNS performance, network paths on fixed connection (no WiFi bias)
- Spans 19 different network providers
- Multiple access technologies (fiber, cable, satellite, fixed wireless)
- Data is publicly available via project website
- Proposed as NAIRR dataset for working with projects like e.g., anomaly detection

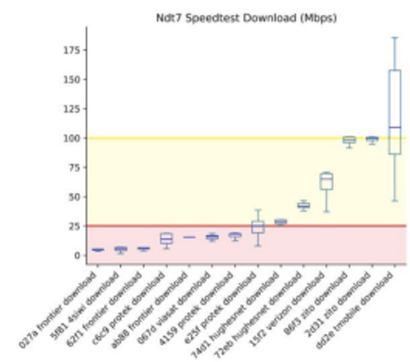


FLOTO: CASE STUDIES

- Computer Science questions: IPv4 versus IPv6
  - Objective: Understand how Internet speed varies between IPv4 and IPv6'
  - Method: sequential speed tests comparing IPv4 and IPv6 results under similar conditions
  - Early Findings: IPv4 and IPv6 speeds degrade differently under various conditions, influenced by the ISP (SIGMOD paper in preparation)
- Policy questions: Marion County
  - Objective: Improve internet infrastructure and performance in Marion County, Illinois
  - Method: Deploy FLOTO devices to collect and analyze broadband performance data
  - Finding: 32% of sampled households below the federal threshold -- data used to support grant applications for fiber broadband expansion

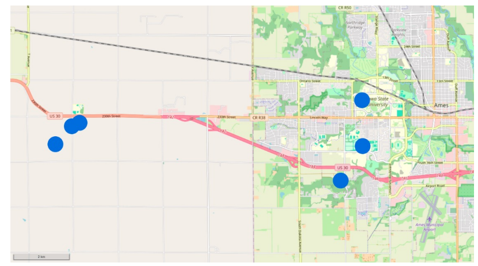


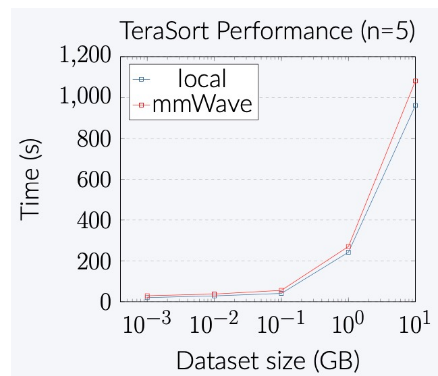


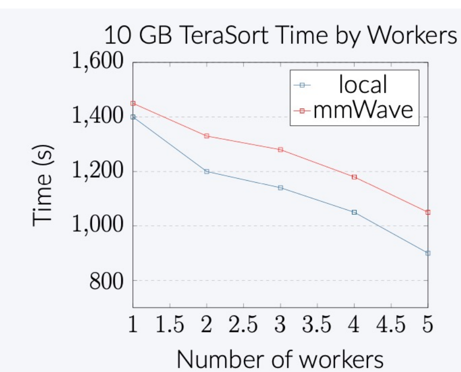


MEASURING RURAL WIRELESS

- Collaboration with ARA project
- Assessing the quality of rural 5G networks
  - Measuring device to device latency
  - Clock synchronization
  - Comparing over different network fabrics
- Deployed 6 Raspberry Pi devices with 5G connectivity in rural Iowa
- Latency measurements: GPS-based time synchronization for precise measurements (4000x more precise than NTP over 5G)
- Tested using Hadoop
- Hey presto: 5G networks can support distributed computing with performance comparable to wired connections!



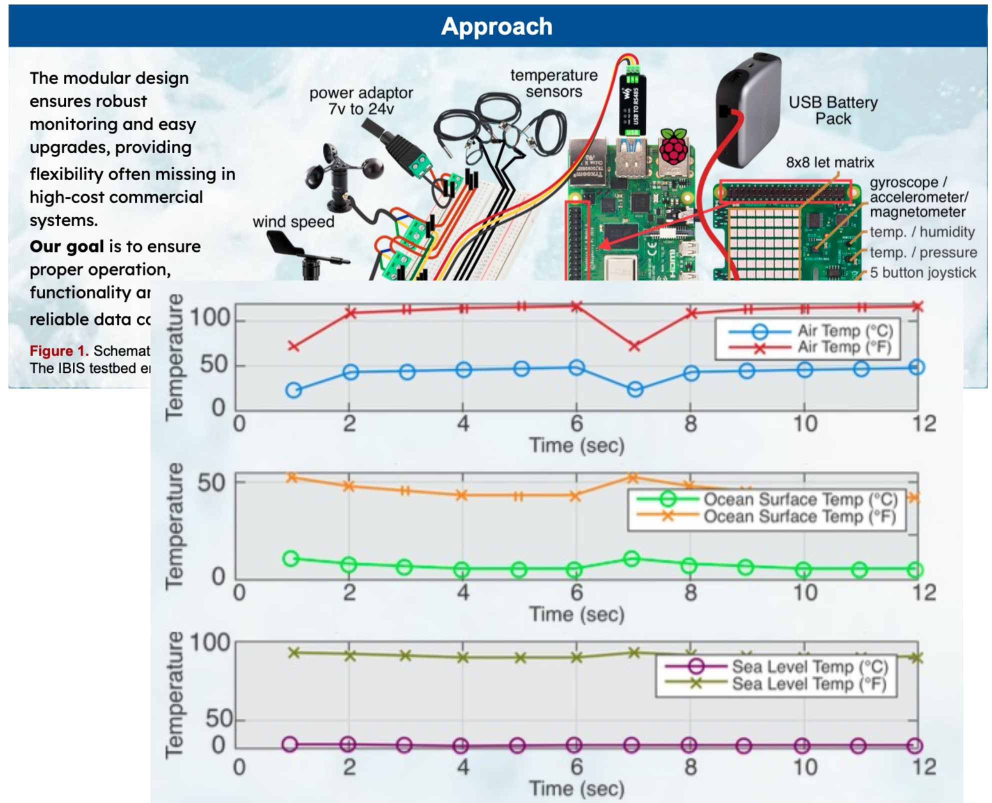




Zack Murry, University of Missouri

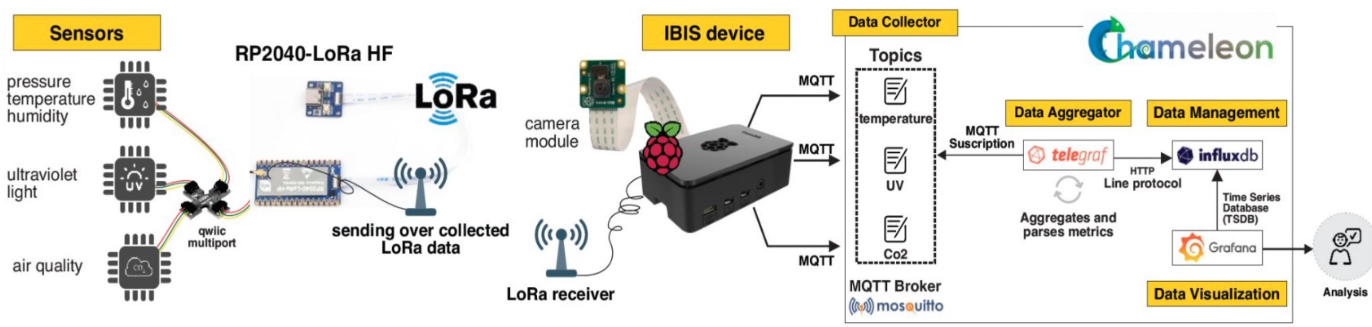
SENSOR STATIONS FOR MARINE AND COASTAL ECOSYSTEMS

- Smart buoy system: sensor stations for oceanic data collection (water quality, water movement, water levels, etc.)
- Collaboration with FIU
- Integrated multiple environmental sensors with IBIS infrastructure
- Demo deployment with real and simulated data
- Implemented cloud-based data visualization system
- Collaboration with FIU



NCAR WEATHER SENSING STATIONS

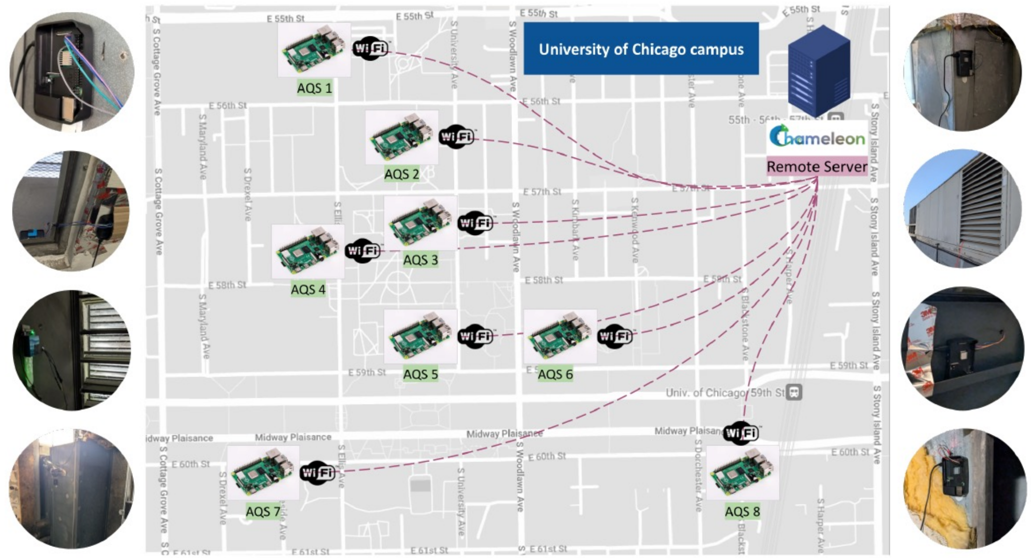
- openIoTwx: NCAR 3D printed weather stations
- Richer continuum: IBIS SBCs connecting to openIoTws via LoRa
  - Exploring power (4x factor), connectivity (cellular vs aggregation via LoRa), sensing (additional camera sensors), and processing (to e.g., reduce size of data) trade-offs
- Future challenges
  - Image-based weather prediction methods, scaling up to create dense, high-resolution weather monitoring networks, and assessing long-term reliability in diverse outdoor environments



William Fowler, Tufts University

AIR-QUALITY FORECASTING WITH FEDERATED LEARNING

- Compare simulation, emulation, and real-world deployments for Federated Learning
- Deployed 8 Raspberry Pis with air quality sensors on UChicago campus
- Simulating FL training on a single compute node can accurately reproduce model performance metrics (accuracy), but presents limitations for reproducing system metrics (training time, CPU usage, and communication latency)



Paper: "On Reproducibility Challenges of Federated Learning: Investigating the Gap between Simulation, Emulation and Real-World Deployments", **Cédric Prigent**, *Inria, France*

SOUNDSCAPING

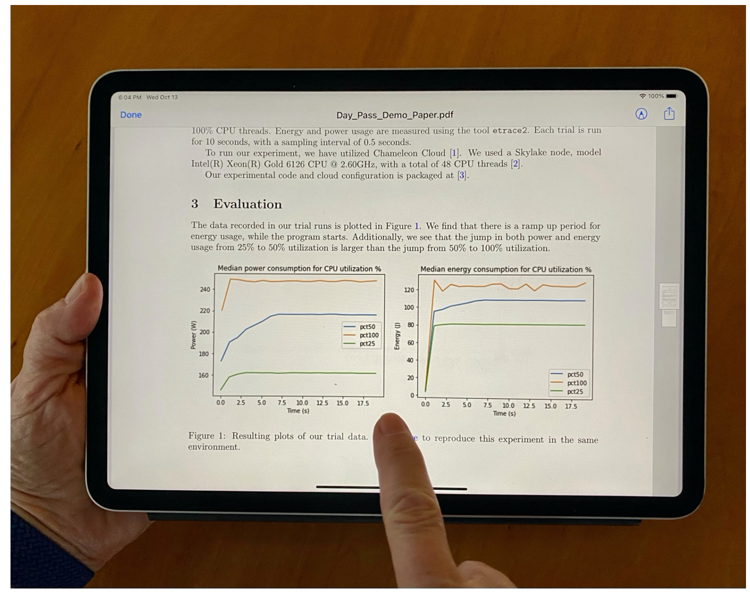
- Using acoustics for biodiversity conservation: tracking wildlife, protecting crops
- Scaling challenge
  - Expensive hardware (~$1,000 per device)
  - Requires manual data collection and servicing
- How can we
  - Reliably stream and analyze audio in **real-time**
  - From **thousands of Listeners**, not dozens
  - While minimizing hardware and operating **costs** for years-long studies
  - In an environments integrating deployment, visualization, storage, and management
- Architecture: custom low cost/power Listeners and Aggregators combine needs-based sensing with network access with an integrated data analytics framework

Hudson Reynolds, Boston University
Alex Tuecke, Worcester Polytechnic Institute

BUILDING AND REPRODUCING EXPERIMENTS

Practical reproducibility == feasible enough to be a mainstream method of scientific exploration

- Can digital experiments be as sharable as papers are today?
- Is there a library I can go to and find experiments to play with?
- Can I simply integrate somebody's model into my research instead of reinventing the wheel and get to a new result faster?
- Can I discover something new through playing with somebody else's experiment?
- Can I develop exercises for my class based on most recent research results?



https://repeto.cs.uchicago.edu

WHAT DO WE HAVE?

- **Open platforms** are essential for sharing – especially in computer science
  - Open, version-controlled hardware
  - Non-fungible resources
- **Experimental environment setup**
  - Disk images, orchestration templates, and other artifacts
  - Thousands of images, orchestration templates, digital artifacts of various kinds
- Are we there yet? If not, what is missing?

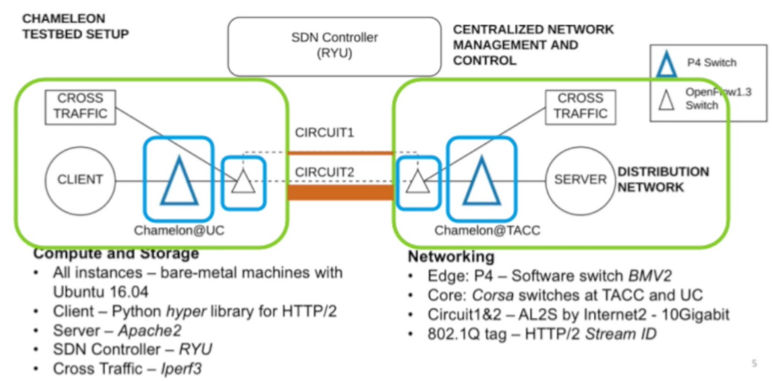
A car without a road

Paper: "The Silver Lining", IEEE Internet Computing 2020

PACKAGING EXPERIMENTAL ENVIRONMENTS

- **Wide versus narrow interfaces**
- **Support for declarative methods**
  - Heat, Terraform, and other mainstream orchestration tools
  - Hard to introspect and could be tricky for reproducibility
- **Support for imperative methods**
  - CLI, python-chi and/or scripts
  - Potentially via Jupyter integration
  - Can be re-played incrementally, troubleshooting and making changes as you go

Package this!

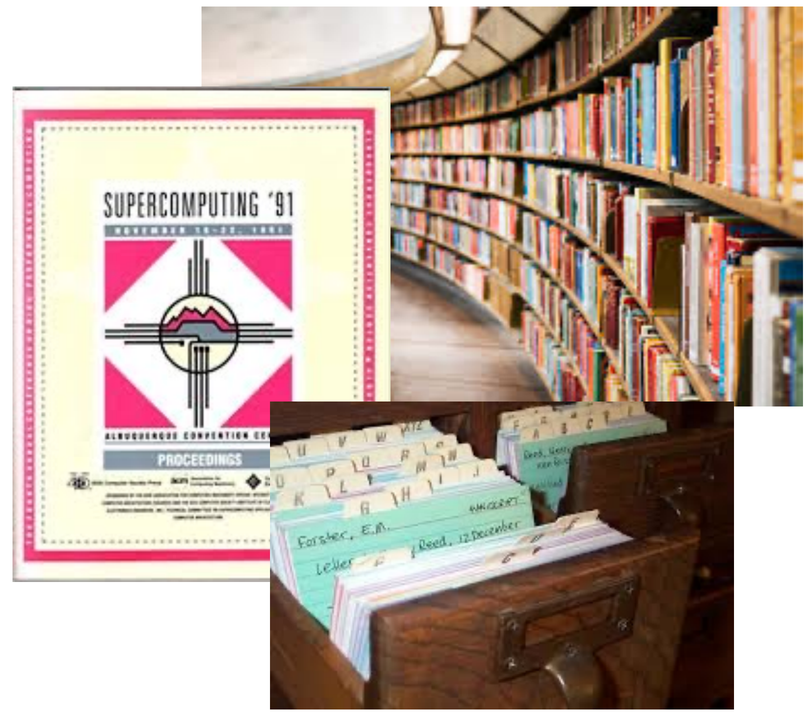


Complex Experimental containers via programmable interfaces

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

TROVI: SHARING, FINDING, AND REPRODUCING

- A testbed-integrated open experiment sharing repository integrated with **multiple testbeds**
- Trovi artifacts
  - Collection of information about all the experiment
  - **Connected to the testbed** such that the experimental environment is easy to deploy
  - Artifacts provide **metrics** about usage – interesting to both authors and reviewers
- Portal to present, browse, filter, and find interesting experiments
- Open APIs: can be integrated with any testbed



Paper: "Three Pillars of Reproducibility", ReWords'23

MPICH CLUSTER

Easy to launch MPI clusters on Chameleon:
standardized, flexible, and ready to run your HPC experiments.

Create MPI Cluster

Pick Orchestration → **Select Base Image** → **Cluster Config** → **SSH into Master Node** → **Run MPI Examples and Experiments**

Jupyter + Ansible
- Launch & configure cluster nodes directly from Jupyter Notebooks
- Uses python-chi and ansible-runner
- An example is shown here.

Heat Orchestration Template
- Provision an MPI+Spack HPC cluster via OpenStack Heat Template
- Bootstrap node automatically configures master & worker nodes
- The complex appliance "MPI+Spack Bare Metal Cluster" is available.

Prebuilt MPI + Spack Images
- All nodes launch with the same configuration.
- Based on Ubuntu 22.04 with MPI + Spack + Lmod.
- OpenMPI and MPICH are installed using Spack.
- Available variants:
  - CPU Only
  - CUDA GPU
  - ROCm GPU

Cluster Setup with Ansible
- Roles defined for master and worker nodes.
- Fully configures nodes; extendable to install packages via Spack.
- Code: pmodels/cc-hpc-appliances

Trovi Artifact: MPI and Spack Based HPC Cluster Experiment

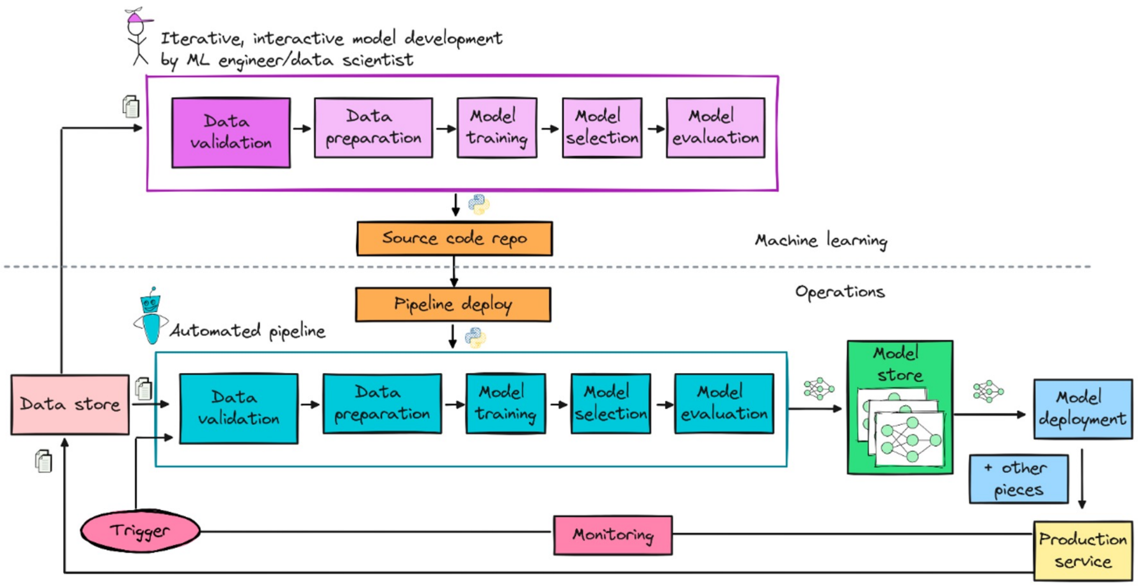
Rohan Babbar and Ken Raffenetti, ANL

TEACHING OPERATIONAL ML AT SCALE

NYU Spring 2025 • ~200 Students • Hosted on Chameleon Cloud

Teaching operational ML is different than teaching ML

- Complete ML operational pipelines
- CI/CD for production ML systems
- Distributed training optimization
- Model serving
- Edge deployment
- Production monitoring at scale



WHAT INFRASTRUCTURE DO WE NEED FOR THIS TYPE OF EDUCATION?

*Students reserved **VM** (m1.small to m1.xxlarge), **bare metal** (GPUs P100, mi100, V100), **network** (floating IPs), and **edge resources** (Raspberry Pi 5) driving exceptional utilization across CHI (even when working in 3-4 person teams to optimize resource usage)*

186K	VM Hours
18K	Bare Metal Hours
11TB	Total Storage
1500+	Leases Created

Why Chameleon?

Reconfigurable
- Kernel mods for K8s
- GPU latency benchmarks/testing

Edge Computing
- Real edge ML serving

Reservations
- Creating VM clusters for storage/training
- Trovi lesson modules

Scan to find the coursework on Trovi!

Paper: "The Cost of Teaching Operational ML", EduHPC/SC25

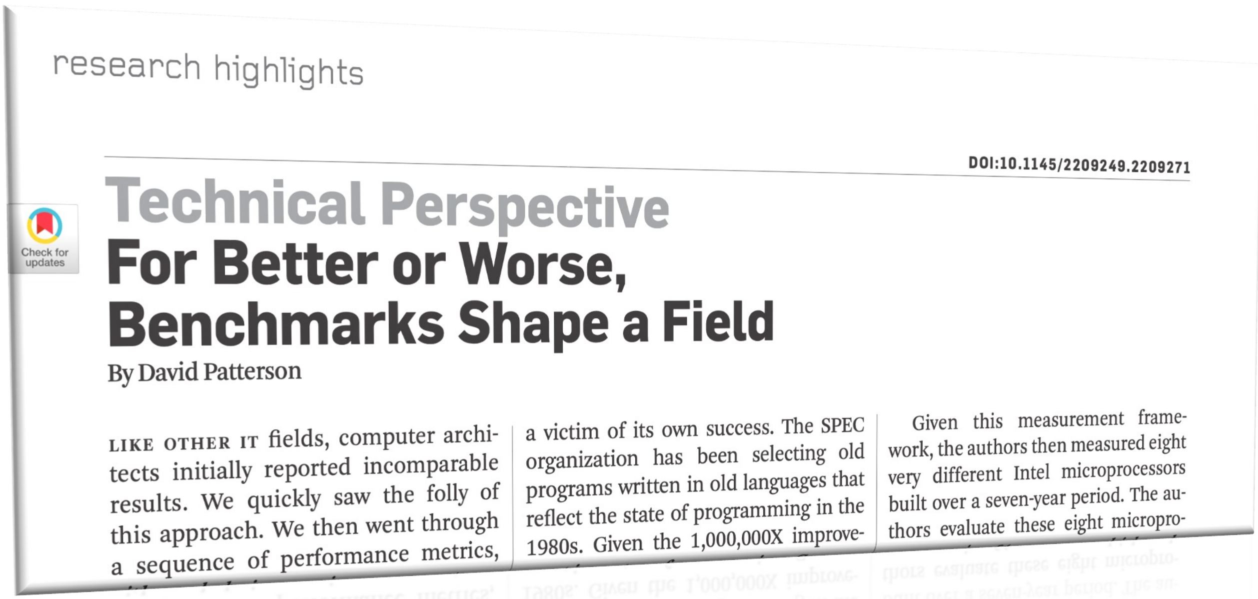
COMMUNITY WORKSHOP ON REPRODUCIBILITY FOR HIGH PERFORMANCE COMPUTING (HPC)

Attended by **reproducibility practitioners**
Experiment **packaging checklists** (GitHub)
Recommendations for authors and reviewers
Community recommendations: new tools and capabilities that the community needs
AE organization

Download the Report

https://reproduciblehpc.org

FOR BETTER OR WORSE,
SCIENTIFIC INSTRUMENTS SHAPE A FIELD


research highlights

DOI:10.1145/2209249.2209271

Technical Perspective
For Better or Worse, Benchmarks Shape a Field

By David Patterson

LIKE OTHER IT fields, computer architects initially reported incomparable results. We quickly saw the folly of this approach. We then went through a sequence of performance metrics,

a victim of its own success. The SPEC organization has been selecting old programs written in old languages that reflect the state of programming in the 1980s. Given the 1,000,000X improve-

Given this measurement framework, the authors then measured eight very different Intel microprocessors built over a seven-year period. The authors evaluate these eight micropro-

We're all snow plough drivers now!

We're here to change

www.chameleoncloud.org