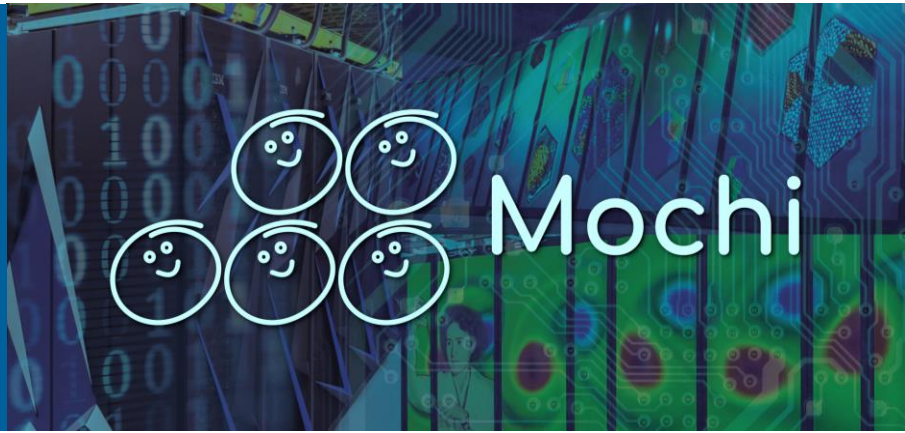
16TH JLESC WORKSHOP

Argonne NATIONAL LABORATORY

HARNESSING PROGRAMMABLE DEVICES IN THE DATA PATH WITH COMPOSABLE SERVICES



PHIL CARNS
carns@mcs.anl.gov
Mathematics and Computer Science Division
Argonne National Laboratory

April 17, 2024
RIKEN Center for Computational Science

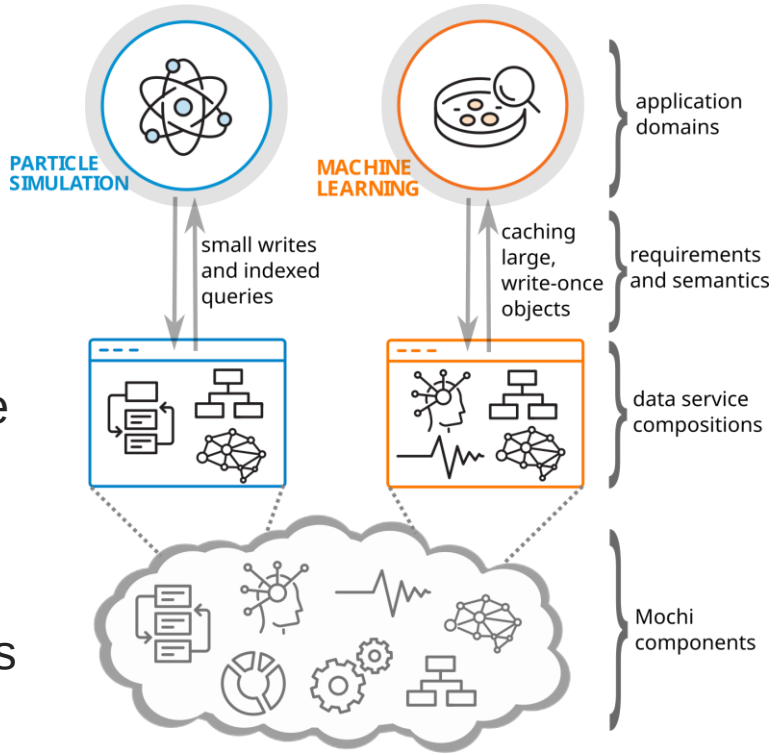
MOCHI BACKGROUND

Composable distributed services

The Mochi project provides a collection of proven **components** and **microservices** along with a **methodology** for rapid development of specialized distributed services.

- Examples of services built with Mochi include domain-specific data management, transient file systems, in-situ analysis, and performance monitoring.
- The key to Mochi is *composability*: components are combined as needed according to the needs of the science domain or platform.

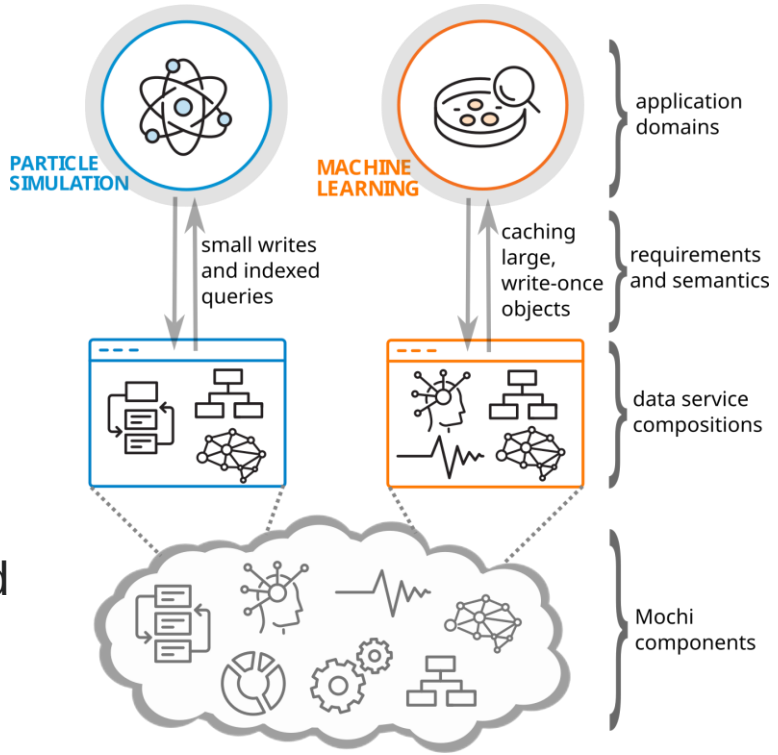
https://www.mcs.anl.gov/research/projects/mochi/



EXAMPLE SERVICES

What can you build with Mochi?

- Specialized file systems
  - **GekkoFS**: (presented earlier today) ad-hoc file system leveraging in-system storage
- Domain-specific data services
  - **HEPnOS**: an object storage service with a C++ API tailored to organizing and analyzing HEP experimental data
- Alternative data models
  - **Mofka**: an event streaming service designed for high volume scientific data
- Things that aren't "data services" exactly
  - **Colza**: an elastic in situ visualization service

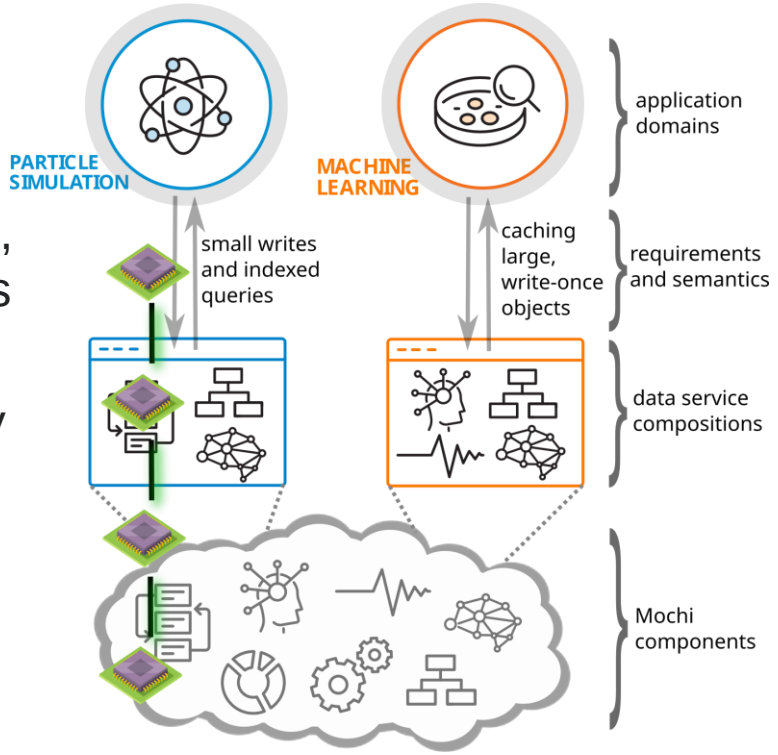


SMART DEVICES IN THE DATA PATH

New possibilities

The Mochi ecosystem also presents a unique opportunity to harness programmable devices.

- **Well-positioned**: Components that span nodes, storage devices, and networks to enable access to off-node resources.
- **Portable**: Intrinsic modularity and composability make it easy to swap in smart-device enabled microservices.
- **On demand**: Services can be instantiated as needed to cater to specific science needs.



EXAMPLE SMART DEVICE CAPABILITIES

- Smart NICs
  - Examples: DPU-enabled Infiniband or P4-enabled Ethernet cards
  - Capabilities: service offload, packet inspection, autonomous communication
- Smart SSDs
  - Examples: SSDs with embedded processors and/or structured APIs
  - Capabilities: inline computation, data restructuring, rich interfaces
- Smart network fabrics:
  - Examples: HPC and enterprise switches with embedded packet processing
  - Capabilities: accelerated collectives, traffic shaping

LIMITATIONS AND OPPORTUNITIES

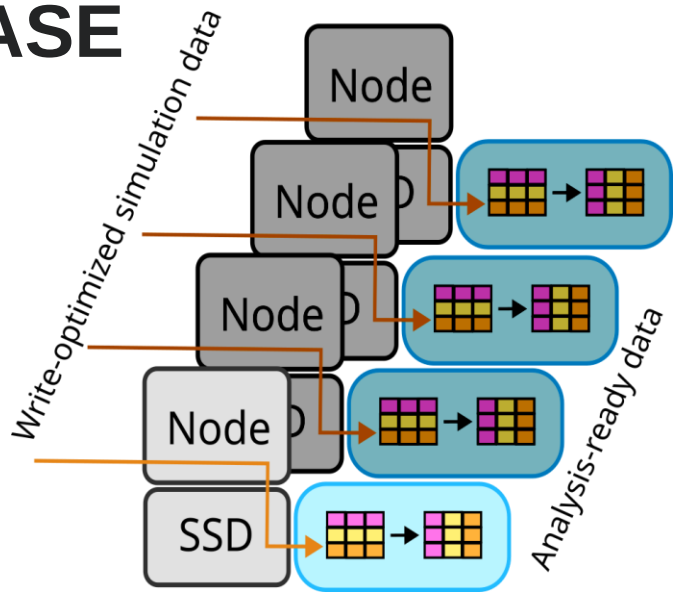
What to keep in mind about smart devices

- *All such devices have limited processing power.*
- Conventional CPUs and GPUs can perform raw calculations far more efficiently.
- There are many other opportunities, however:
  - To reduce data movement
  - To reduce power consumption
  - To reduce resource contention
  - To decouple control plane features
- Smart devices can be thought of as a way to strategically move functionality to “the other side of the PCI bus”.

EXAMPLE SMART SDD USE CASE

Data layout transformations

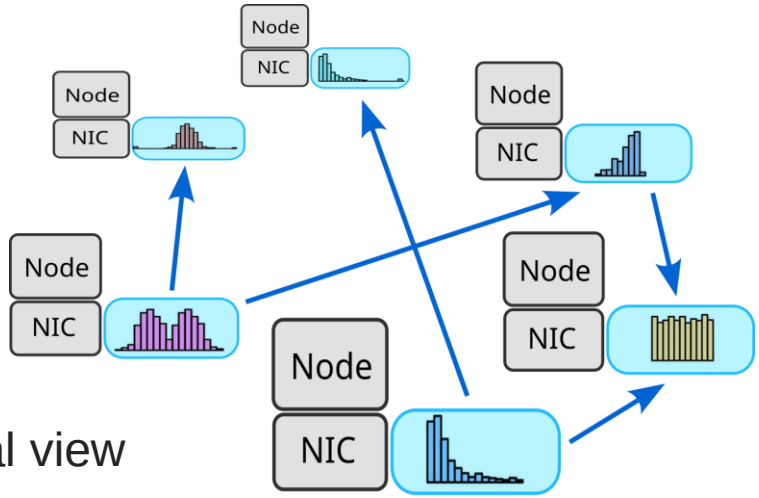
- Smart SSDs can manipulate data, for example to translate between write-optimized and analysis optimized formats or perform log compaction.
- Why are smart SSDs advantageous?
  - They can rewrite data asynchronously.
  - Minimal need for computational power in this use case



- Why not conventional host processes?
  - Locality: Host-based transformation requires moving the data across the PCI and memory bus (twice).
  - Runtime efficiency: Work can be shifted off of the runtime critical path by decoupling the data production rate and data translation rate.

EXAMPLE SMART NIC USE CASE

Aggregate telemetry and traffic shaping



- Smart NICs can observe host traffic, gather lightweight metrics over sliding windows, and use those metrics to inform transparent load balancing and shedding policies.
- Why are smart NICs advantageous?
  - Visibility over all network traffic
  - Ability to exchange data to construct a global view
- Why not conventional host processes?
  - Smart NICs can also encompass clients in a manner that is agnostic to applications, tasks, virtual machines.
  - Smart NICS can exchange data at high frequency without perturbing host PCI bus or interrupting processes.

COLLABORATIVE OPPORTUNITIES

- Find me this week if you have questions, suggestions, or ideas (on this topic or Mochi in general)!
- This talk highlighted a few potential scenarios, but much more is possible.
- Of particular interest:
  - Specific scientific computing use cases that would benefit from specialized data services and/or smart devices in the data path
  - Expertise with smart devices in other contexts that could be translated to distributed / HPC services
- I'm also happy to chat with people about the Darshan I/O characterization tool (not covered in this talk).

THANK YOU!

THIS WORK WAS SUPPORTED BY THE U.S. DEPARTMENT OF ENERGY, OFFICE OF SCIENCE, ADVANCED SCIENTIFIC COMPUTING RESEARCH, UNDER CONTRACT DE-AC02-06CH11357.