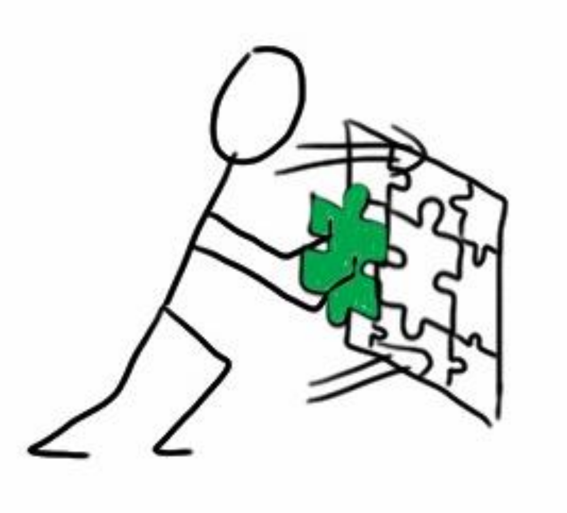
Building Custom Data Services With Mochi

Philip Carns (Argonne National Laboratory)
Philip Davis (University of Utah)
Matthieu Dorier (Argonne National Laboratory)
Chris Kelly (Brookhaven National Laboratory)
Rob Ross (Argonne National Laboratory)
Jerome Soumagne (The HDF Group)

May 12, 2022

What’s changing in HPC data services?



Application pull:

- Artificial intelligence use cases
- Use of HPC in experimental science (e.g., ATLAS/CMS)
- Streaming data

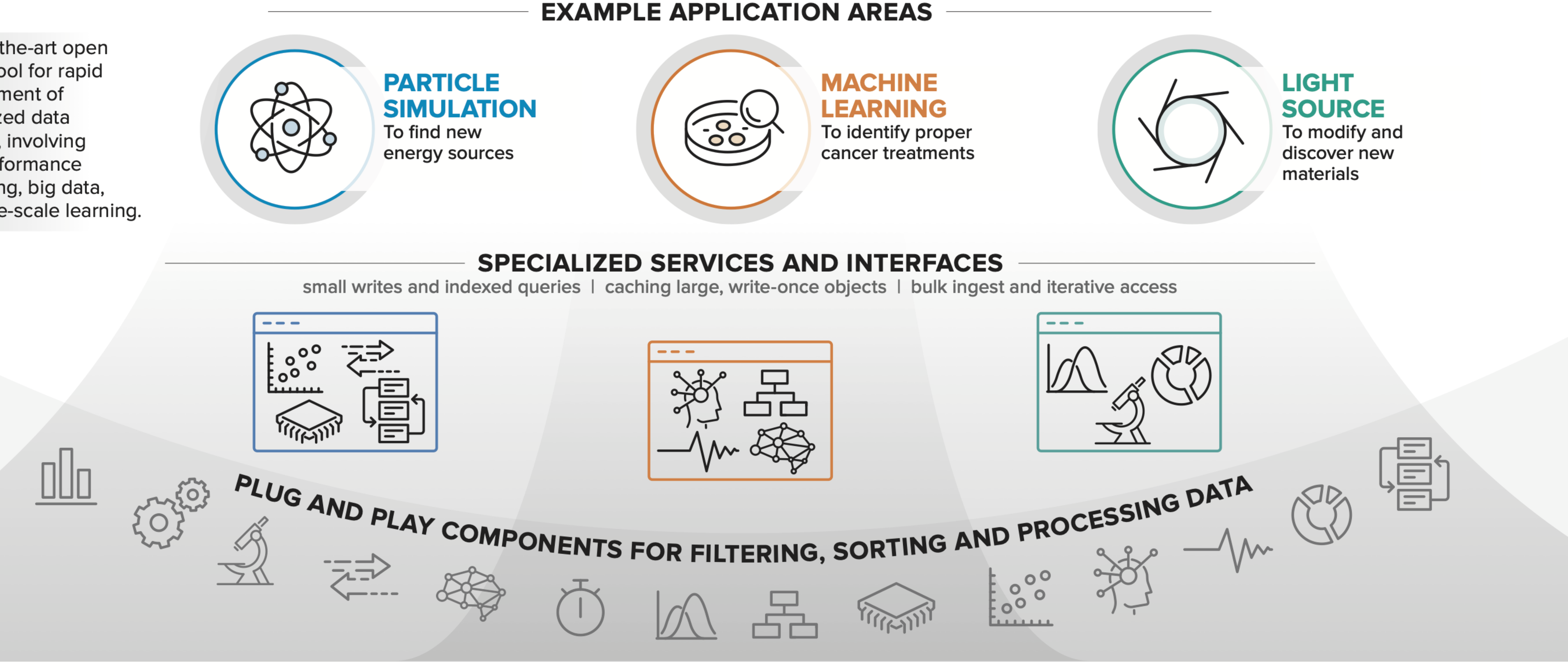
Technology push:

- New networking APIs and capabilities
- More capable storage technologies
- Compute in storage

Image from M. Geurden, “Market Opportunity Identification: Push or Pull?,”July 2012, https://newentrepreneurship.nl/2012/07/02/market-opportunity-identification-push-or-pull/

Mochi
Customized data services for DOE science

State-of-the-art open source tool for rapid development of customized data services, involving high-performance computing, big data, and large-scale learning.



EXAMPLE APPLICATION AREAS

PARTICLE SIMULATION
To find new energy sources

MACHINE LEARNING
To identify proper cancer treatments

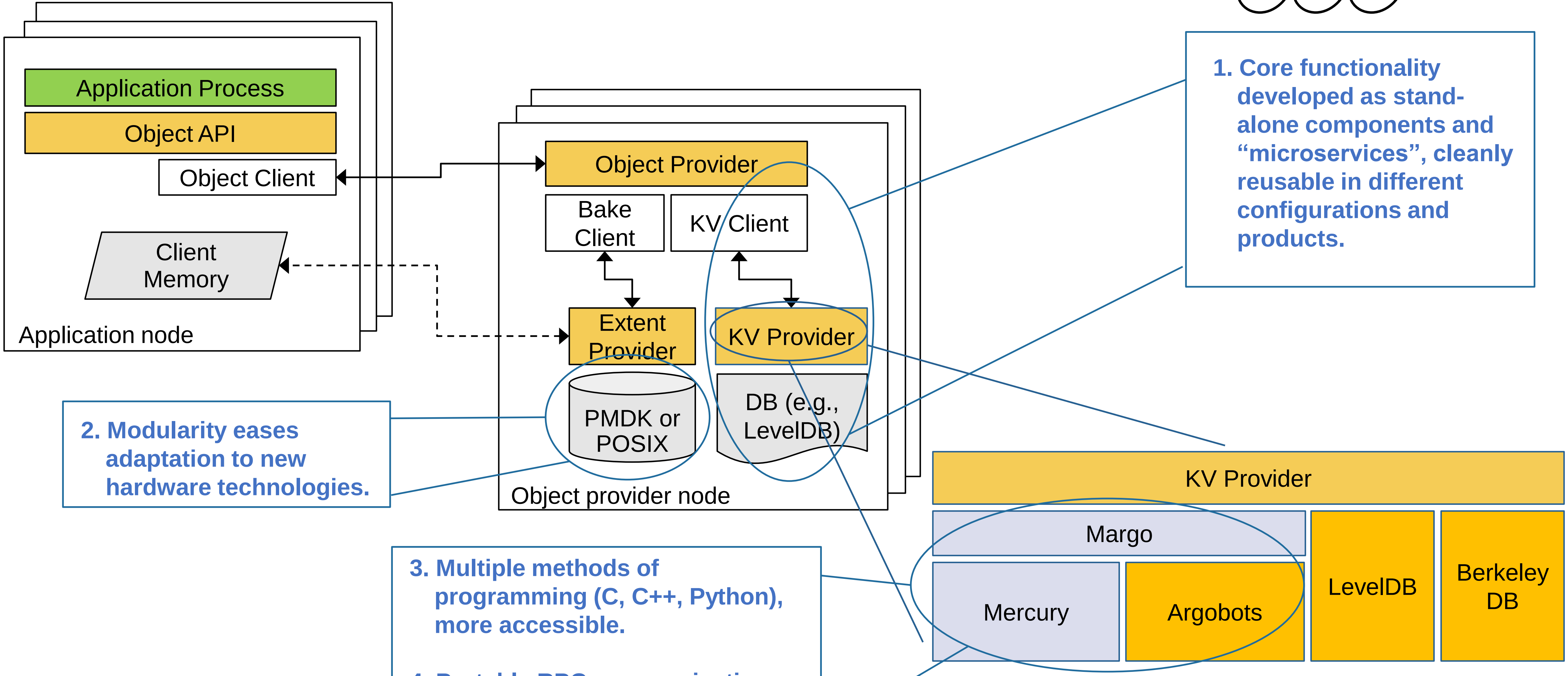
LIGHT SOURCE
To modify and discover new materials

SPECIALIZED SERVICES AND INTERFACES

small writes and indexed queries | caching large, write-once objects | bulk ingest and iterative access

PLUG AND PLAY COMPONENTS FOR FILTERING, SORTING AND PROCESSING DATA

What’s new in the Mochi approach?



1. Core functionality developed as stand-alone components and “microservices”, cleanly reusable in different configurations and products.

2. Modularity eases adaptation to new hardware technologies.

3. Multiple methods of programming (C, C++, Python), more accessible.

4. Portable RPC communication library designed for multi-service environments

	Component	Summary
Core		
	Argobots	Argobots provides user-level thread capabilities for managing concurrency.
	Mercury	Mercury is a library implementing remote procedure calls (RPCs).
	Margo	Margo is a C library using Argobots to simplify building RPC-based services.
	Thallium	Thallium allows development of Mochi services using modern C++.
	SSG	SSG provides tools for managing groups of providers in Mochi.
Utilities		
	ABT-IO	ABT-IO enables POSIX file access with the Mochi framework.
	Bedrock	Bedrock is a bootstrapping and configuration system for Mochi components.
	ch_placement	ch-placement is a library implementing multiple hashing algorithms.
	Shuffle	Shuffle provides a scalable all-to-all data shuffling service.
Microservices		
	BAKE	Bake enables remote storage and retrieval of named blobs of data.
	POESIE	Poesie embeds language interpreters in Mochi services.
	REMI	REMI is a microservice that handles migrating sets of files between nodes.
	Sonata	Sonata is a Mochi service for JSON document storage based on UnQLite.
	Yokan	Yokan enables RPC-based access to multiple key-value backends.

Agenda

Times in EST

Mochi

11:00 – 11:15	Welcome and Introductions	Rob Ross
11:15 –11:25	Getting Started	Phil Carns
11:25 – 11:40	New Components: Bedrock and Yokan	Matthieu Dorier
11:40 – 11:55	Mercury Updates	Jerome Soumagne
11:55 – 12:10	Case Study 1: DataSpaces	Philip Davis
12:10 – 12:25	Case Study 2: Chimbuko	Chris Kelly
12:25 – 12:30	Wrap-up	Rob and Phil