Ross R, Amvrosiadis G, Carns P et al. Mochi: Composing data services for high-performance computing environments. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 35(1): 121–144 Jan. 2020. DOI 10.1007/s11390-020-9802-0

Mochi: Composing Data Services for High-Performance Computing Environments

Philip Carns, Matthieu Dorier, Kevin Harms, Robert Latham, Rob Ross, and Shane Snyder
Argonne National Laboratory

Sam Gutierrez, Bob Robey, Brad Settlemyer, and Galen Shipman
Los Alamos National Laboratory

George Amvrosiadis, Chuck Cranor, Greg Ganger, nd Qing Zheng
Carnegie Mellon University

Dana Robinson and Jerome Soumagne
The HDF Group

Garth Gibson
Vector Institute for Artificial Intelligence

Argonne NATIONAL LABORATORY

What are we trying to accomplish?

We're trying to transform HPC data services from a monoculture to an ecosystem.

- Redefining how teams design and develop distributed services for use in HPC systems.
- Providing a portable "programming model" for these services.
- Providing a set of core building blocks.
- Demonstrating the methodology and tools with DOE science use cases.

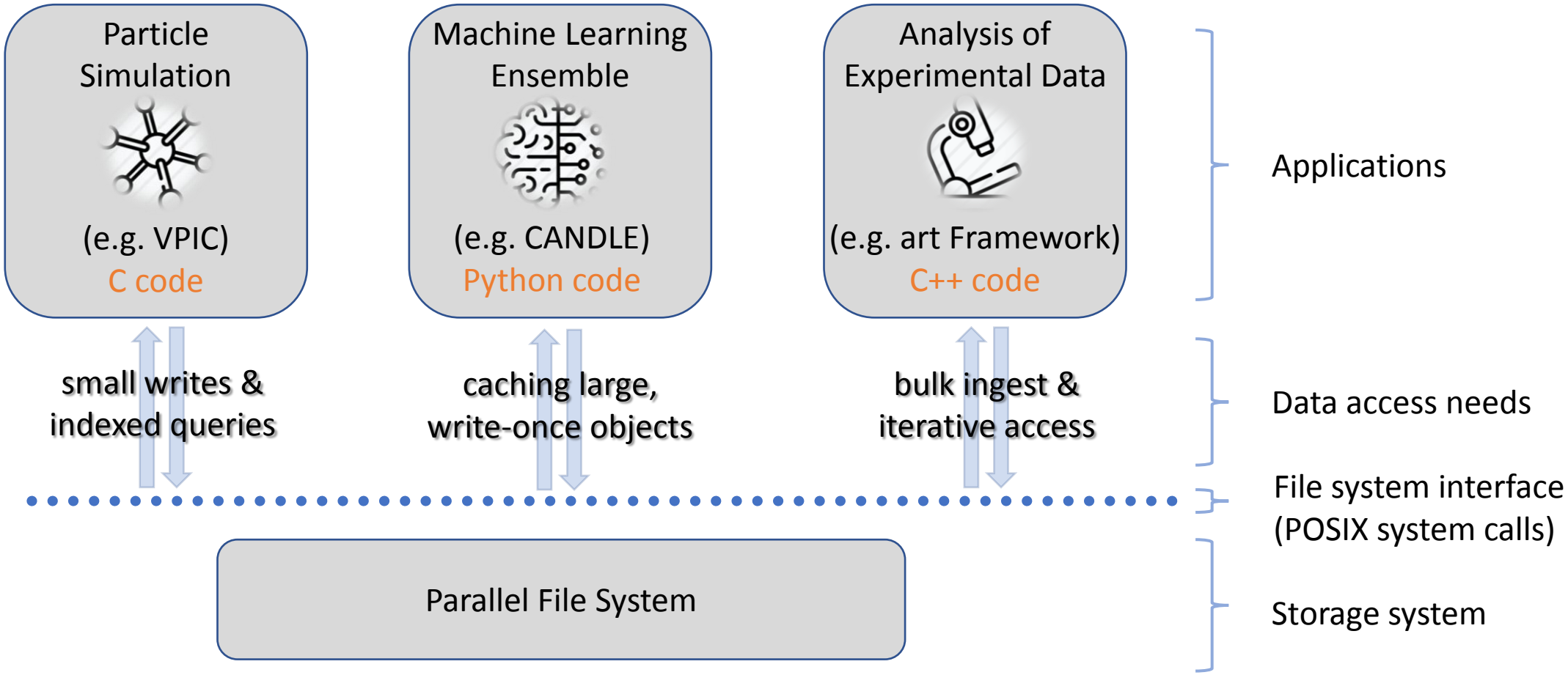
We're trying to foster a community of service developers.

- Developing a set of training materials that will help others employ the tools.
- Making all these building blocks available to the larger community.

Carnegie Mellon University
HDF
Los Alamos NATIONAL LABORATORY EST. 1943
Argonne NATIONAL LABORATORY

How is this traditionally done in HPC?

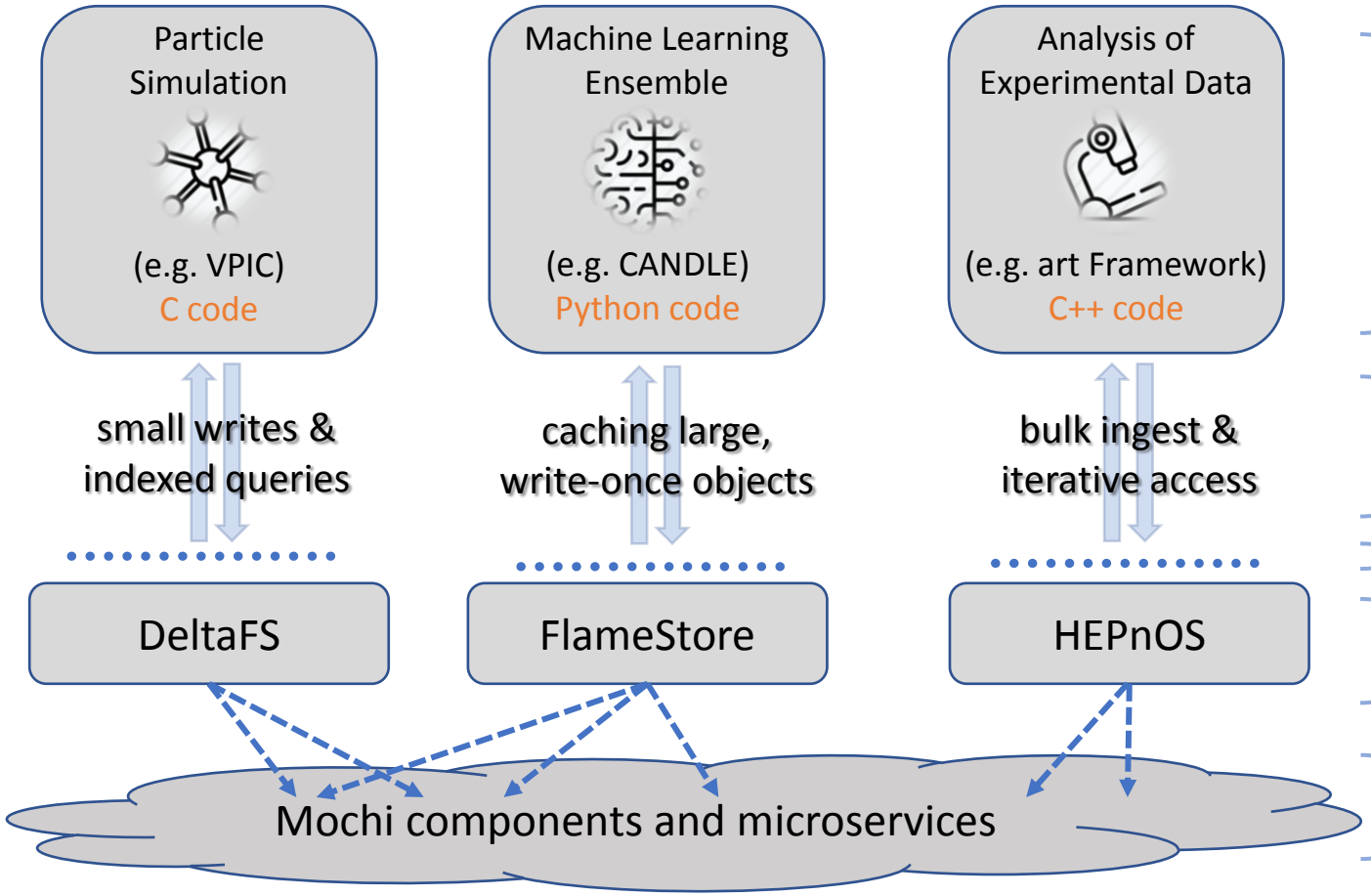
File system monoculture for data (dis)service



All applications use the same "one size fits all" file system interfaces, semantics, and policies for data access.

What's new in the Mochi approach?

An ecosystem of services co-existing and reusing functionality

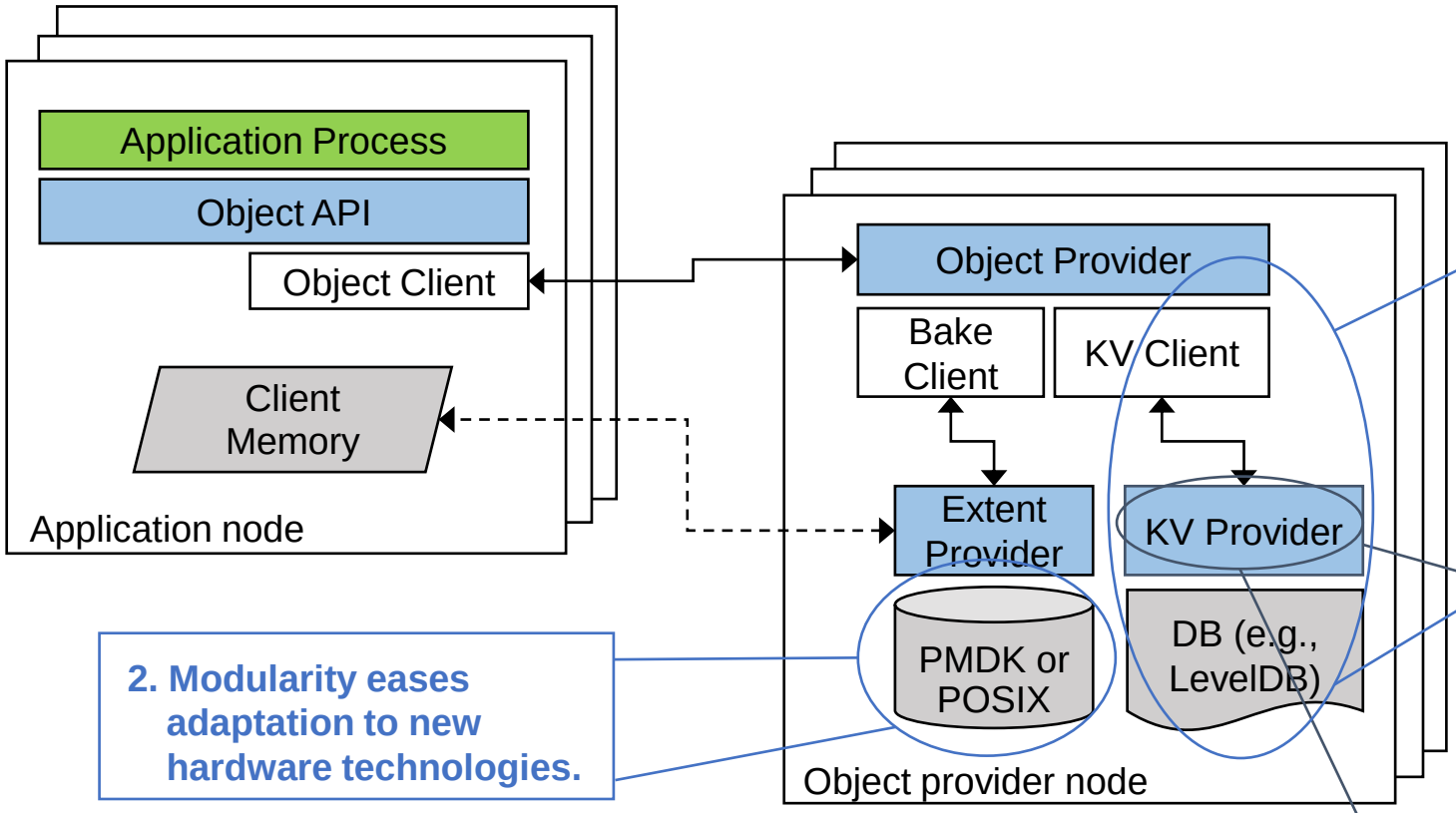


This approach has allowed us to simultaneously pursue multiple specialized service implementations.

Instead of "one size fits all", Mochi data services present tailored interfaces, semantics, and policies for data access while still leveraging robust building blocks.

Carnegie Mellon University · HDF · Los Alamos National Laboratory · Argonne National Laboratory

What's new in the Mochi approach?

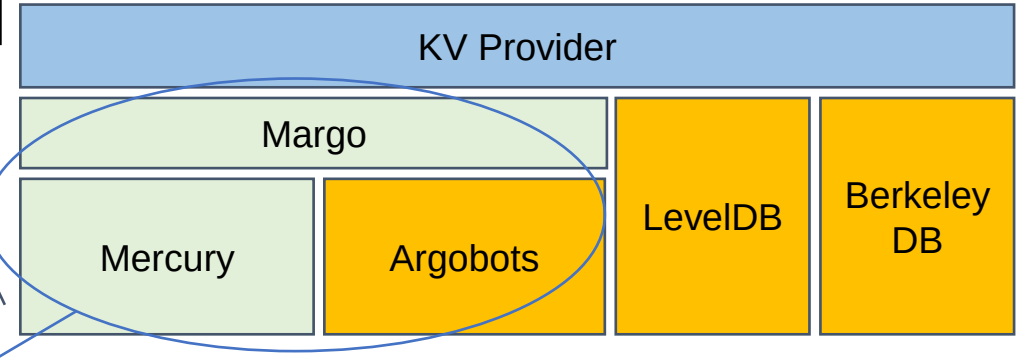


1. Core functionality developed as stand-alone components and "microservices", cleanly reusable in different configurations and products.

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



Carnegie Mellon University · HDF · Los Alamos National Laboratory · Argonne National Laboratory

Thanks!

This work is in part supported by the Director, Office of Advanced Scientific Computing Research, Office of Science, of the U.S. Department of Energy under Contract No. DE-AC02-06CH11357; in part supported by the Exascale Computing Project (17-SC-20-SC), a joint project of the U.S. Department of Energy's Office of Science and National Nuclear Security Administration, responsible for delivering a capable exascale ecosystem, including software, applications, and hardware technology, to support the nation's exascale computing imperative; and in part supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, Scientific Discovery through Advanced Computing (SciDAC) program.

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