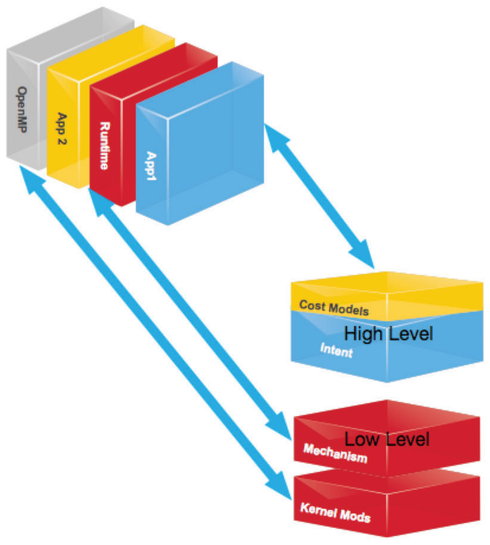
Simplified Interface to Complex Memory

EXASCALE COMPUTING PROJECT

Los Alamos NATIONAL LABORATORY · Lawrence Livermore National Laboratory · OAK RIDGE National Laboratory · Sandia National Laboratories · Georgia Tech · THE UNIVERSITY OF TENNESSEE KNOXVILLE · NC STATE

Overview



Interface for complex memory that is

- abstract
- portable
- easy-to-use
- powerful for power users
- extensible to future hardware

Low-level Interface

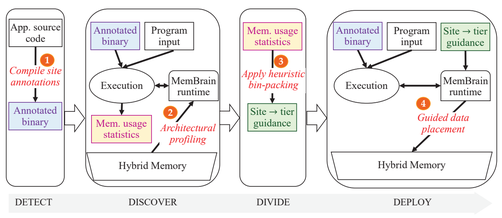
Mechanism focused low-level interface that reins in heterogeneity

High-level Interface

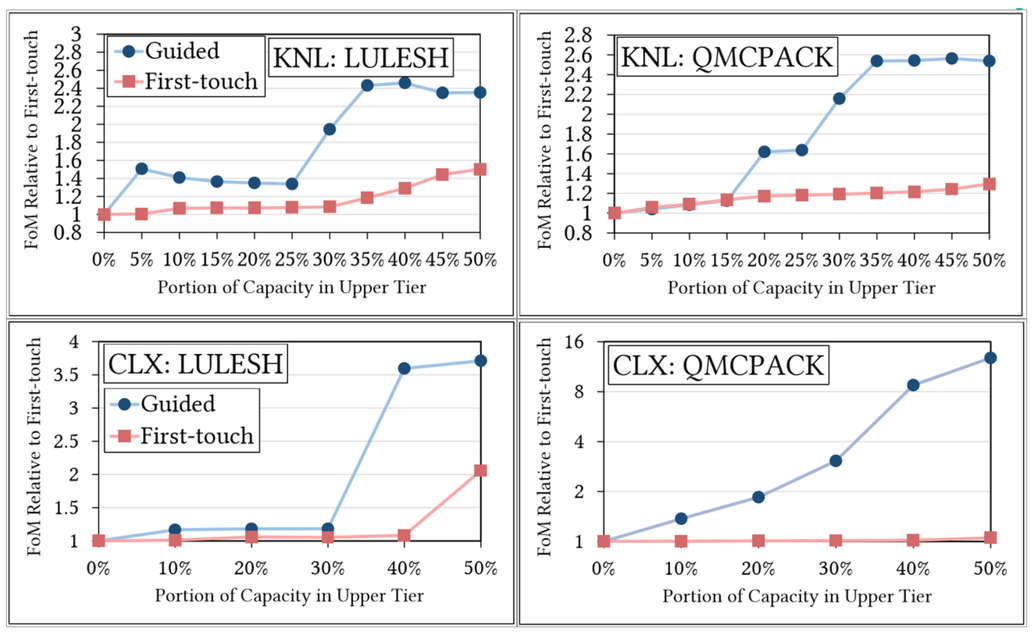
High-level interface that makes reasonable decisions for applications

Portable Application Guidance for Complex Memory Systems

Goal: Place Highly Accessed Data in the Fastest Tier Available



- Toolchain that can discover memory hotspots.
- Make use of compiler allocation context to improve hotspot isolation.
- Efficiently fit data in appropriate memory tiers.



- Benefits observed regardless of upper tier capacity.
- Benefits more pronounced with more upper tier capacity.

Evaluating the Effectiveness of Program Data Features for Guiding Memory Management

Feature	Distinguishes objects ...
application	in the same application
size	allocated with exactly the same size
size bucket	allocated with similar, but not necessarily identical sizes
type	with the same data type
allocation phase	allocated during the same phase
phase signature	alive during the same set of phases
access signature	accessed by the same set of instructions
allocation site	allocated from the same instruction
allocation context	Allocated from the same calling context

- Even simple program features are effective at improving performance.
- Allocation context is most effective for predicting memory behavior.
- Still effective when different inputs are used.

LA-UR-20-20721

Low-level Interface

Current solutions was fine for two memory types, but not really extendable to more.

- Best path forward is current NUMA distance + extensions.
- Kernel work to allow orderings of NUMA nodes, modifications to memory policies.
- Currently targeting Intel KNL, IBM Power9+Volta, and Intel Optane DC PMM.
- Continuing discussions with vendors.

Using the idea of arena-based heap management to reduce the complexity of the data placement decisions.

- Data structures that are used together belong to the same arena.
- Arenas are migrated between memory types as a whole.
- Needed for arbitration and introspection.

Implemented as userspace library on top of jemalloc.

SICM Interactions with other ECP Projects

OpenMP

- Initial support for OpenMP *target* directives under development.

Umpire

- Implemented resource, allocator, and strategy.
- Driving development of external resource interface.

Kokkos

- Added SICM MemorySpace as alternative to default HostSpace.

OMPI-X

- Extended Open MPI to specify memory type for allocations.
- Support for data placement in the context of MPI communications.

Global Arrays

- Initial integration with SICM is complete.
- Developing support for shared memory.

Using Non Volatile Memories to build energy and cost efficient HPC clusters

- Intel's Optane DC PMM provide multiple node's worth of capacity on a single node.
- Clusters with larger capacity can be built at lower scaling of other costs.

Metall (Meta Allocator for persistent memory)

- Persistently store large data structures
- Access and update in-memory data stores in local NVMe

See Metall poster for details.

In development. Contact Mike Lang for details - mlang@lanl.gov