SCALABLE COMPUTING SOFTWARE LABORATORY

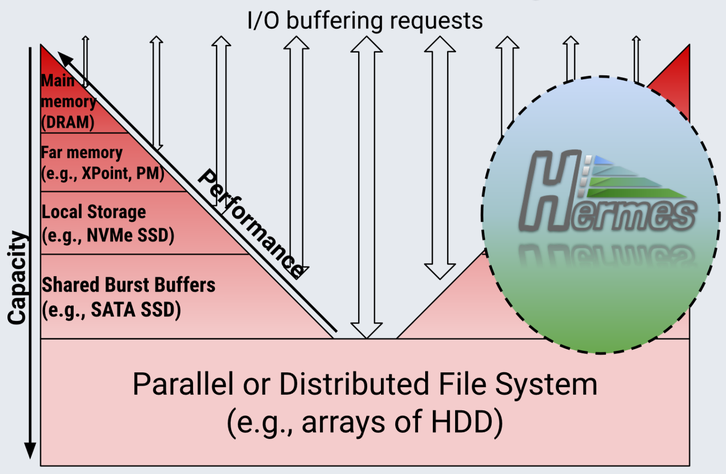
ILLINOIS INSTITUTE OF TECHNOLOGY

Hermes: Multi-Tiered I/O Buffering

Xian-He Sun, sun@iit.edu

Department of Computer Science, Illinois Institute of Technology

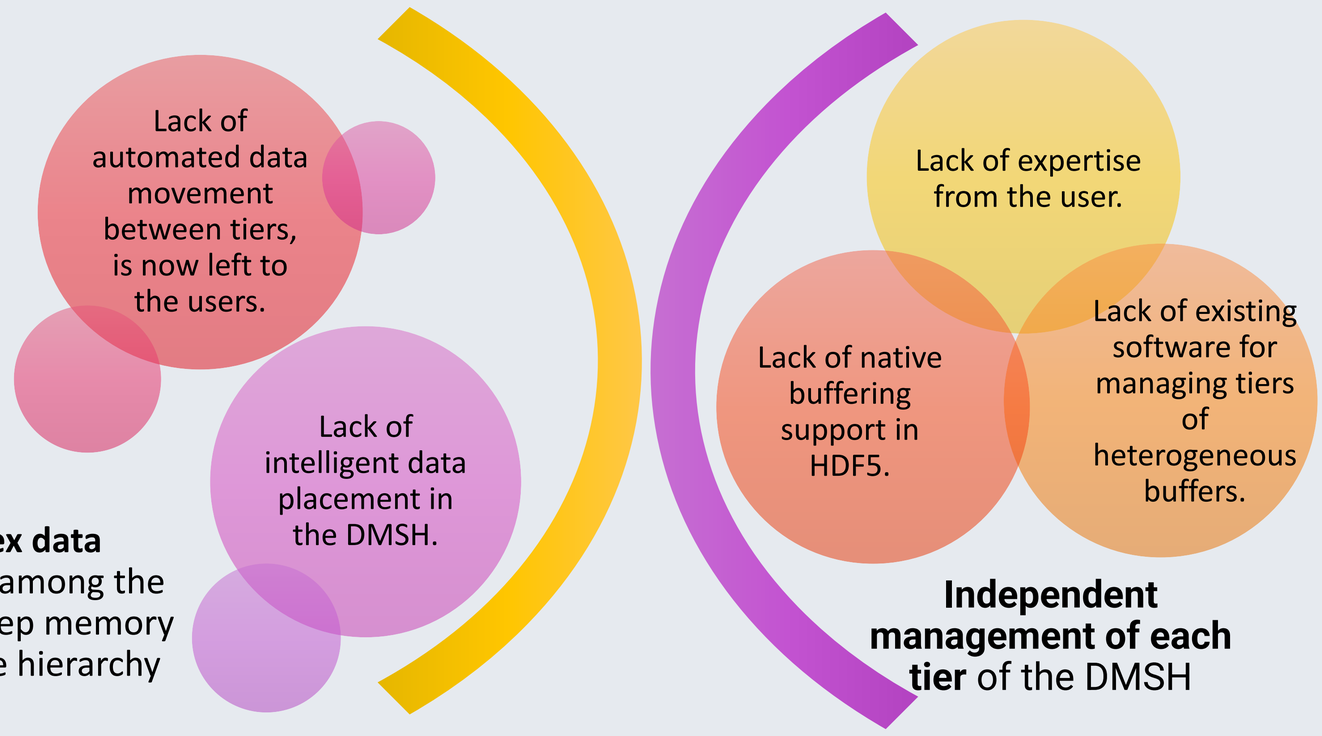
Multi-Tiered Storage



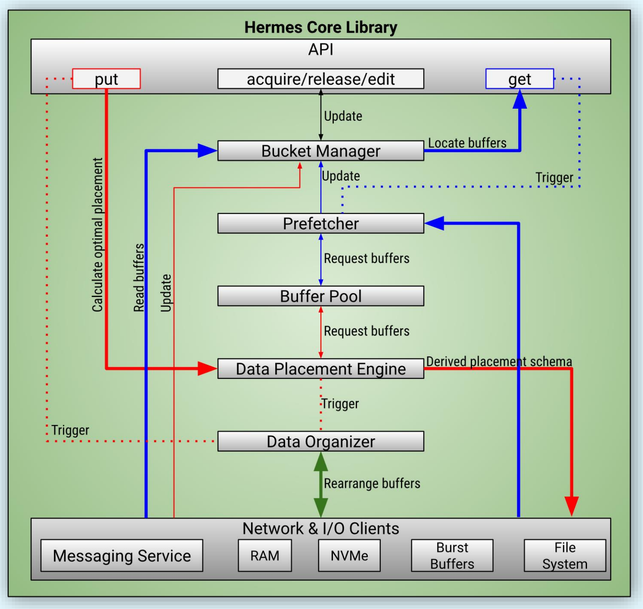
- New storage system designs incorporate non-volatile burst buffers between the main memory and the disks.
- HPC hierarchical storage systems with burst buffers (BB) have been installed at several HPC sites.
- Multiple levels of memory and storage in a hierarchy, called **DMSH**.

Synopsis

Current Situation



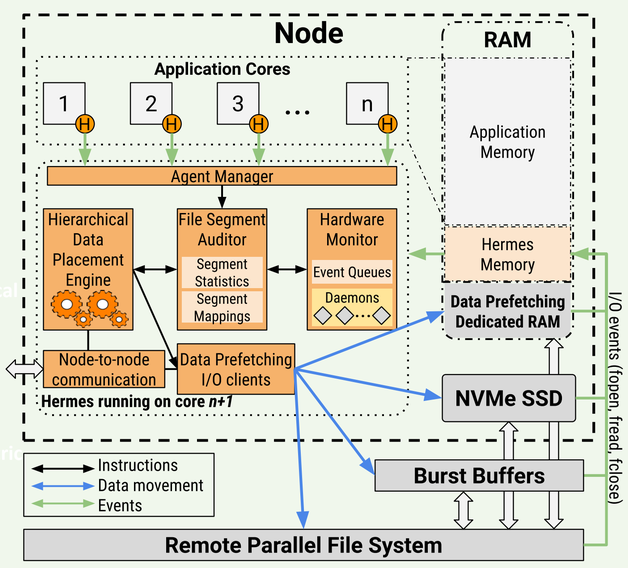
Overview



- **Hermes API:**
  - intercepts all I/O calls from the applications.
  - calculates the operations to be carried out in case of an active buffering scenario.
- **Hermes Data Placement Engine (DPE)**
  - calculates the data destination, i.e., where in the hierarchy should the data be placed.
  - uses various data placement policies.
- **Hermes Buffer Organizer**
  - event-based component
  - carries out all data movements
    - E.g., for prefetching reasons, evictions, lack of space, or hotness of data etc.
- **Metadata Manager**
  - maintains two types of metadata:
    - user's metadata operations (e.g., files, directories, permissions etc.),
    - Hermes library's internal metadata (e.g., locations of all buffered data and internal temporary files that contain user files).
- **Messaging Service**
  - enables horizontal buffering
  - provides an infrastructure to pass instructions to other nodes to perform operations on data or facilitate its movement
- **Buffer Pool Manager**
  - handles all buffers inside Hermes
  - equipped with several data replacement policies

Current Progress

Prefetching



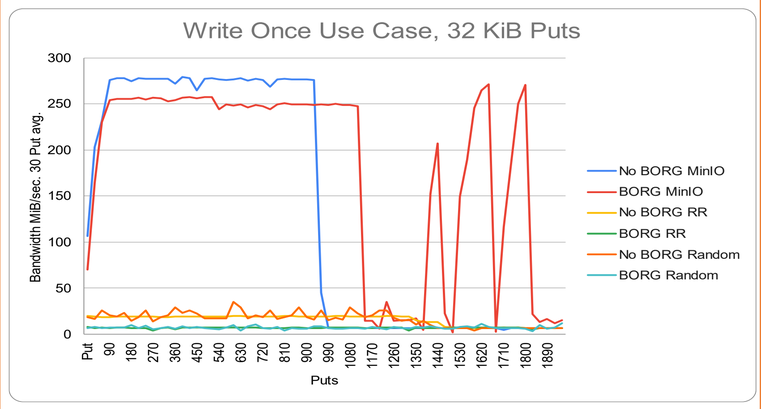
- **Server-Push**
  - Event are captured by kernel's inotify utility
  - Prefetched data is push to the hierarchy
- **Data Centric (Score Incorporates)**
  - Recency, Frequency, and Sequencing
- **Hierarchical Placement**
  - The engine calculates placement of prefetch data based on multi-tiered storage and data characteristics.

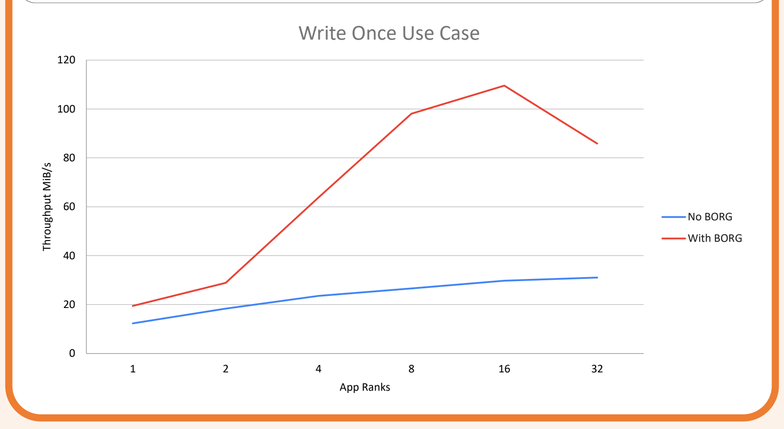
Buffer Organizer

- **Decoupled architecture**
  - Borg attempts to correct sub-optimal DPE placements by moving data among buffers.
- **Objectives**
  - Management of hierarchical buffering space
  - Data flushing
  - Read acceleration
  - Manage data life cycle, or journey
- **Blob Scoring System**
  - Blob Size
  - Blob Name
  - Recency of Blob access
  - Frequency of Blob access
  - Reference count
  - Blob links
  - User-supplied priority
- **Operators**
  - MOVE(BufferID, TargetID)
  - COPY(BufferID, TargetID)
  - DELETE(BufferID)

Results

BORG





Ongoing

Hermes and Friends

Working closely with application domain scientists:

- Storm tracking workflows from PNNL
- Integrate Hermes in their workflow
- Accelerate I/O operations via buffering
- Test Hermes code base at scale
- Simplify deployment and usage
- Identify optimization opportunities
- Enhance legacy API support

Contact

Xian-He Sun, PI
sun@iit.edu
www.cs.iit.edu/~scs

Anthony Kougkas, Lead
akougkas@iit.edu

Find more