Understanding Data Motion in the Modern HPC Data Center
Scientific computing is more than compute!
Goal: Understand data motion everywhere
Our simplified model for data motion

Storage Systems

External Facilities

Compute Systems

Storage-External

Storage-Storage

Compute-External

Compute-Compute

Storage-Storage
Mapping this model to NERSC

- External Facilities
  - DESI: 100 GB/night
  - NCEM: 45 GB/s

- Storage Systems
  - 1.8 PB, 1.5 TB/s
  - 30 PB, 700 GB/s
  - 12 PB, 130 GB/s
  - 160 PB, 50 GB/s
  - 12,076 nodes, 1.4 PB DRAM

- Compute Systems
Relevant logs kicking around at NERSC

**Globus logs**
- no remote storage
- system info missing

**HPSS logs**
- some remote storage
- system info missing

**Storage Systems**

**External Facilities**

**Darshan**
- data volumes come with caveats

**Compute Systems**
## Normalizing data transfer records

### Storage System → Compute-Storage → Compute System

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Source</strong> site, host, storage system</td>
<td>NERSC, Cori, System Memory</td>
</tr>
<tr>
<td><strong>Destination</strong> site, host, storage system</td>
<td>NERSC, Cori, cscratch1 (Lustre)</td>
</tr>
<tr>
<td><strong>Time</strong> of transfer start and finish</td>
<td>June 4 @ 12:28 – June 4 @ 12:32</td>
</tr>
<tr>
<td><strong>Volume</strong> of data transferred</td>
<td>34,359,738,368 bytes</td>
</tr>
<tr>
<td><strong>Tool</strong> that logged transfer</td>
<td>Darshan, POSIX I/O module</td>
</tr>
<tr>
<td><strong>Owner</strong> of data transferred</td>
<td>uname=glock, uid=69615</td>
</tr>
</tbody>
</table>
What is possible with this approach?

May 1 – August 1, 2019
• 194 million transfers
• 78.6 PiB data moved

- 8 TiB/day ≤ 8 TiB/day
- 8 TiB/day > 8 TiB/day
- 16 TiB/day > 16 TiB/day
- 32 TiB/day > 32 TiB/day
- 64 TiB/day > 64 TiB/day
- 128 TiB/day > 128 TiB/day
Visualizing data motion as a graph

- Job I/O is most voluminous
- Home file system usage is least voluminous
- Burst buffer is read-heavy
- Users prefer to access archive directly from Cori than use DTNs
Mapping this data to our model
Adding up data moved along each vector

- **Job I/O is significant**
- **Inter-tier is significant**
  - I/O outside of jobs ~ job write traffic
  - Fewer tiers, fewer tears
- **HPC I/O is not just checkpoint-restart!**

Data Transferred (TiB)

- 512 TiB
- 128 TiB
- 32 TiB
- 8 TiB
- 2 TiB
- 512 GiB

- Compute-Storage
- Storage-Compute
- Storage-Storage
- Storage-WAN
- WAN-Storage
Examining non-job I/O patterns

- **Hypothesis:** non-job I/O is poorly formed
  - Job I/O: optimized
  - Others: fire-and-forget
- **Users transfer larger files than they store** (good)
- **Archive transfers** are largest (good)
- **WAN transfers** are smaller than job I/O files (less good)
Few users resulted in the most transfers

- 1,562 unique users
- Top 4 users = 66% of volume transferred
- Users 5-8 = 5.8%
  - All used multiple transfer vectors
  - Henry is a storage-only user
Examining transfers along many dimensions

• Break down transfers by r/w and file system
  – Rereading same files
  – Targeting cscratch (Lustre)
Tracing using users, volumes, and directions

- **Correlating reveals workflow coupling**
  - S-S precedes C-S/S-C
  - 2:1 RW ratio during job
  - Data reduction of archived data

- **This was admittedly an exceptional case**
Is this the full story?

Quantify the amount of transfers *not* captured

- Compare volume transferred to system monitoring (storage systems)
- Compare bytes in to bytes out (transfer nodes)
Not every data transfer was captured

- 100% true data volume should be captured by transfers
- Missing lots of data—why?
  - Darshan logs not generated; cp missing
  - Globus-HPSS adapter logs absent
  - Only Globus logged; rsync/bbcp absent
Identifying leaky transfer nodes

- **Incongruency ($\Delta$)**
  - data in vs. data out
  - FOM for how “leaky” a node is
  - $\Delta = 0$ means all bytes in = all bytes out

- **Cori:** expect $>> 0$ because jobs generate data

- **Science gateways > 0** because ???

![Chart showing incongruency values for Cori, HPSS Gateway, DTNs, and Science Gateways]

- Cori: 1.27
- HPSS Gateway: 0.137
- DTNs: 0.018
- Science Gateways: 0.613
Towards Total Knowledge of I/O

New profiling tools to capture I/O from other transfer tools (bbcp, scp, etc)

Better insight into what is happening inside Docker containers

Improve analysis process to handle complex transfers

More robust collection of job I/O data; cache-aware I/O data (LDMS)
There’s more to HPC I/O than job I/O

• **Inter-tier I/O is too significant to ignore**
  – need better monitoring of data transfer tools
  – users benefit from fewer tiers, strong connectivity between tiers
  – need to optimize non-job I/O patterns

• **Transfer-centric approaches yield new holistic insight into workflow I/O behavior**
  – Possible to trace user workflows across a center
  – Humans in the loop motivate more sophisticated methods
TOKIO

We gratefully acknowledge the support of

• Damian Hazen (NERSC)
• Kristy Kallback-Rose (NERSC)
• Nick Balthaser (NERSC)
• Lisa Gerhardt (NERSC)
• Ravi Cheema (NERSC)
• Jon Dugan (ESnet)
• Eli Dart (ESnet)

We’re hiring!

This material is based upon work supported by the U.S. Department of Energy, Office of Science, under contracts DE- AC02-05CH11231 and DE-AC02-06CH11357. This research used resources and data generated from resources of the National Energy Research Scientific Computing Center, a DOE Office of Science User Facility supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231 and the Argonne Leadership Computing Facility, a DOE Office of Science User Facility supported under Contract DE-AC02-06CH11357.
Few users result in the most transfers

- Amy/Darshan
- Bob/Darshan
- Carol/Darshan
- Dan/Darshan
- Eve/Darshan, Globus, HPSS
- Frank/Darshan, Globus
- Gail/Darshan, Globus
- Henry/HPSS

Cumulative fraction of total volume transferred

Size of transfer

Number of users

Percent total volume transferred

0 10 50 150 250 350
10% 1% 0.1% 0.01% 0.001% 0.0001%
Regularity of user I/O coupling

- **MUTC**
  - how correlatable is a user’s I/O across all vectors
  - how easily we can guess what a user’s workflow is doing
- **Strongest correlation only between job reads and job writes**
- “**Excluding C-S/S-C**” only shows workflows with storage-storage or storage-WAN activity

![Graph showing mean user correlation coefficient](image)

- **All vectors**
- **Excluding C-S/S-C vectors**

1,123 users represented in “all vectors”
486 users represented in “excluding C-S/S-C vectors”