Agenda

- Purpose
- About DataWarp
- Examples using DataWarp Scratch
- Examples using DataWarp Cache
- Summary
- Q&A
Why DataWarp?

- Many programs do I/O in bursts
- Want to have high bandwidth when doing I/O
- SSDs offer greater bandwidth per dollar
- Use SSDs to handle peaks in I/O bandwidth requirements
- Reduce job wall clock time
Memory Hierarchy Evolution

Past

On Node

CPU

Memory (DRAM)

Storage (HDD)

Off Node

Off Node

Future

On Node

CPU

Near Memory (HBM/HMC)

Far Memory (DRAM/NVDIMM)

Near Storage (SSD)

Far Storage (HDD)

Now

DataWarp
DataWarp Components

- CN - Compute Node
- LN - Lnet Router Node
- DW - DataWarp Node

IB Fabric

Lustre Filesystem

OSSs / OSTs

Aries

Compute | Store | Analyze

Copyright 2017 Cray Inc.
DataWarp - Minimize Compute Residence Time

```
<table>
<thead>
<tr>
<th>Node Count</th>
<th>Time (Lustre Only)</th>
<th>Time (DataWarp)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial Data Load</td>
<td>Compute</td>
<td>DW Preload</td>
</tr>
<tr>
<td>Final Data Writes</td>
<td>Final Data Writes</td>
<td>DW Post Dump</td>
</tr>
</tbody>
</table>

Key:
- Compute Nodes
- Compute Nodes - Idle
- I/O Time Lustre
- I/O Time DataWarp
- DW Nodes

Timestep Writes
Timestep Writes (DataWarp)
DataWarp Types

Scratch

Application

SSD  SSD  SSD

Cache

Application

SSD  SSD  SSD

Parallel Filesystem
DataWarp Scratch

- Striped
- Private
Use Scratch as storage between job steps

#!/bin/bash
#MSUB -l nodes=16:ppn=4
#MSUB -l walltime=1:00:00
#DW jobdw type=scratch access_mode=striped capacity=50TiB

aprun -n 64 IOR -a POSIX -g -b 8G -t 1M -e -G 1234567890 -w -k -o $DW_JOB_STRIPED/ior_example_1

aprun -n 64 IOR -a POSIX -g -b 8G -t 1M -e -G 1234567890 -W -o $DW_JOB_STRIPED/ior_example_1
DataWarp Staging

Scratch

Application

SSD  SSD  SSD

Compute | Store | Analyze

Copyright 2017 Cray Inc.
#!/bin/bash

#MSUB -l nodes=16:ppn=4
#MSUB -l walltime=1:00:00
#DW jobdw type=scratch access_mode=striped capacity=50TiB

#DW stage_in type=file source=/lus/snx11108/overby/ex3_data
destination=$DW_JOB_STRIPED/input

#DW stage_out type=directory destination=/lus/snx11108/overby/results3
source=$DW_JOB_STRIPED/output

aprun -n 16 IOR -o $DW_JOB_STRIPED/input -k -v -b 64m -t 1m -E -C -W -r -G 1248
aprun -n 1 mkdir $DW_JOB_STRIPED/output
aprun -n 16 IOR -o $DW_JOB_STRIPED/output/res -k -v -b 64m -t 1m -E -C -w -G 1632 -F
Persistent Storage

- Storage that persists across batch jobs
  - Remains until deleted by it’s owner or reaches its end-of-life date

- Access to a persistent instance can be requested by any user
  - Subject to workload manager configuration
  - POSIX permissions (owner/group/other) still limit access to data

- Multiple persistent storage instances can be requested by a single job
Creating a persistent instance

● Controlled by the workload manager
  ● Each has a different way of creating a persistent instance

● Slurm syntax:

```bash
#!/bin/bash
#BB create_persistent name=overby123 capacity=1GiB
access=striped type=scratch

#!/bin/bash
#BB destroyPersistent name=overby123
```
Using a Persistent Instance

#!/bin/bash
#MSUB -l nodes=16:ppn=4
#MSUB -l walltime=1:00:00
#DW persistentdw name=overby123
#DW persistentdw name=overby987

aprun -n 64 IOR -a POSIX -g -b 8G -t 1M -e -G 1234567890 -w -W -r -o $DW_PERSISTENT_STRIPED_overby123/input1

aprun -n 64 IOR -a POSIX -g -b 8G -t 1M -e -w -W -r -o $DW_PERSISTENT_STRIPED_overby987/input2 -G 1248163264
#!/bin/bash

#MSUB -l nodes=16:ppn=4
#MSUB -l walltime=1:00:00

#DW jobdw type=scratch access_mode=private capacity=100GiB

echo DW_JOB_PRIVATE $DW_JOB_PRIVATE
#!/bin/bash
#MSUB -l nodes=16:ppn=4
#MSUB -l walltime=1:00:00
#DW jobdw type=scratch access_mode=striped,private capacity=100GiB

echo DW_JOB_STRIPED $DW_JOB_STRIPED
echo DW_JOB_PRIVATE $DW_JOB_PRIVATE

aprun -n 1 df -h $DW_JOB_STRIPED $DW_JOB_PRIVATE
DataWarp Cache Type

- Striped
- Load Balance
DataWarp Cache

#!/bin/bash
#MSUB -l nodes=4:ppn=4
#MSUB -l walltime=1:00:00
#DW jobdw type=cache access_mode=striped capacity=4GiB
pfs=/lus/snxs1

# Read a file on the PFS through cache
aprun -n 16 IOR -k -v -b 64m -t 1m -E -C -W -G 163264 -o $DW_JOB_STRIPED_CACHE /overby/example_3_input

# Write a file to cache and read it back
aprun -n 16 IOR -k -v -b 64m -t 1m -E -C -w -G 163264 –W -o $DW_JOB_STRIPED_CACHE/overby/example_3_cache
#!/bin/bash

# MSUB -l nodes=32:ppn=4
# MSUB -l walltime=1:00:00

#DW jobdw type=cache access_mode=ldbalance capacity=1TiB pfs=/lus/scratch

aprun -n 1 ls -al $DW_JOB_LDBAL_CACHE
### Summary

<table>
<thead>
<tr>
<th>Scratch</th>
<th>Cache</th>
</tr>
</thead>
<tbody>
<tr>
<td>Striped</td>
<td>DW_JOB_STRIPED</td>
</tr>
<tr>
<td></td>
<td>DW_JOB_STRIPED_CACHE</td>
</tr>
<tr>
<td>Load Balance</td>
<td>DW_JOB_LDBAL_CACHE</td>
</tr>
<tr>
<td>Private</td>
<td>DW_JOB_PRIVATE</td>
</tr>
<tr>
<td>Persistent</td>
<td>DW_PERSISTENT_STRIPED_name</td>
</tr>
<tr>
<td></td>
<td>DW_PERSISTENT_STRIPED_CACHE_name</td>
</tr>
</tbody>
</table>
Legal Disclaimer

Information in this document is provided in connection with Cray Inc. products. No license, express or implied, to any intellectual property rights is granted by this document.

Cray Inc. may make changes to specifications and product descriptions at any time, without notice.

All products, dates and figures specified are preliminary based on current expectations, and are subject to change without notice.

Cray hardware and software products may contain design defects or errors known as errata, which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Cray uses codenames internally to identify products that are in development and not yet publically announced for release. Customers and other third parties are not authorized by Cray Inc. to use codenames in advertising, promotion or marketing and any use of Cray Inc. internal codenames is at the sole risk of the user.

Performance tests and ratings are measured using specific systems and/or components and reflect the approximate performance of Cray Inc. products as measured by those tests. Any difference in system hardware or software design or configuration may affect actual performance.

The following are trademarks of Cray Inc. and are registered in the United States and other countries: CRAY and design, SONEXION, and URIKA. The following are trademarks of Cray Inc.: APPRENTICE2, CHAPEL, CLUSTER CONNECT, CRAYPAT, CRAYPORT, ECOPHLEX, LIBSCI, NÖDEKARE, REVEAL, THREADSTORM. The following system family marks, and associated model number marks, are trademarks of Cray Inc.: CS, CX, XC, XE, XK, XMT, and XT. The registered trademark LINUX is used pursuant to a sublicense from LMI, the exclusive licensee of Linus Torvalds, owner of the mark on a worldwide basis. Other trademarks used in this document are the property of their respective owners.