SLURM BurstBuffer Integration

David Paul
Computational Systems Group
Lawrence Berkeley National Lab
DPAUL@LBL.GOV
September 26, 2016
Overview:

- Cori System / NERSC-8 / Cray XC40 / “Native” SLURM WLM
- BurstBuffer as Cray’s Datawarp Product
- Brief Introduction to Datawarp concepts
- Datawarp usage examples and the SLURM interface to Datawarp
- Datawarp Status, Problem Identification and Error Recovery

Note: BurstBufferBoF @ SC16 bof110s1 - Tue Nov 15 Time: 12:15pm-1:15pm Room: 255-D
Phase 1 – Initial install through 19-Sept-2016
- 144 DW servers (288 SSDs, two DW servers/blade)
- 1,628 Haswell nodes
- 27PB Lustre Parallel Filesystem - $SCRATCH
- Global GPFS - $HOMEs, $PROJECTs, S/W, Modules, etc.
- BurstBuffer of 900 TB @ 900 GB/sec, 12.5M IOPS (measured)

Phase 2 – installation underway
- +144 DW servers (288 servers, 576 SSDs)
- ~2,000 Haswell nodes
- ~9,300 KNL nodes
- Total BurstBuffer of 1.8 PB @ ~1.6 TB/sec, 12.5M IOPS (estimated)
Datawarp Component Interaction
Datawarp Terms

- **DWS** - DataWarp Service - software for managing and configuring the SSD I/O installation
  - **Pool** - subset of DW-servers with a common allocation granularity (ex. 200GB), ex. wlm_pool
  - **Session** - typically created by a DW-enabled WLM job (Token = JobID or Name)
  - **Instance** - an object representing a user’s request for disk space (ex. 600GB)
  - **Fragment** - a piece of an instance, as it exists on a DW-server (ex. 3 @ 200GB)
  - **Configuration** - an object representing how the space is to be used (scratch, striped, private, etc.)
  - **Namespace** - represents the metadata (called tree) and data (called data), i.e. a FileSystem (DWFS)
  - **Registration** - an object for linking together a session and a configuration
  - **Activation** - an object representing where a configuration is to be used (i.e. mounted)
  - **Realm** - group of DWFS mount points that cooperate, present what appear to be different FileSystems

- **DWFS** - DataWarp FileSystem - a Cray FileSystem that supports staging of files and striping data across multiple DW-servers.
- **DVS** - Data Virtualization Service - Cray’s I/O forwarding software (enables GPFS & DWFS I/O on HSN)
- **XFS** - FileSystem used to persist data flowing through DWFS
- **DW-Server** – DVS server with SSDs, DWS, DWFS, XFS, LVM and access to a PFS
Datawarp Process overview
SLURM configuration for Datawarp (very simple)

- slurm.conf : BurstBufferType=burst_buffer/cray
- burst_buffer.conf :
  - **DefaultPool**: name of the pool used by default for resource allocations
    - wlm_pool
  - **AltPoolName**: allows for different storage configurations (ex. Granularity size)
  - **DenyUsers**: list of user names and/or IDs prevented from using burst buffers
  - **Flags EnablePersistent**: allows users to create/destroy persistent burst buffers
  - **Flags TeardownFailure**: remove DW allocation on job failure
- **QoS/TRES** – control user access, user quotas, usage and report them
DWS API Client Helper (one example)

dw_wlm_cli -f

- **pools**: show pool information
- **paths**: generate environment variables to be injected into running batch job
- **job_process**: validate correctness of #DW directives
- **setup**: create session, instance, configurations
- **data_in**: perform stage-in activities
- **pre_run**: prepare compute nodes for Datawarp
- **post_run**: revoke compute node Datawarp access
- **data_out**: perform stage-out activities
- **teardown**: clean up all job-affiliated DW state
DWS’ dwcli vs. SLURM (one session)

```
# dwcli –j ls session
"created": 1473889069,
"creator": "CLI",
"expiration": 0,
"expired": false,
"id": 9711,
"links": {
  "client_nodes": [],
  "owner": 95448,
  "state": {
    "actualized": true,
    "fuse_blown": false,
    "goal": "create",
    "mixed": false,
    "transitioning": false
  }
  "token": "tractorD"
}

# scontrol show burst | grep dpaul
Name=tractorD CreateTime=2016-09-14T14:37:49 Pool=wlm_pool Size=7200G State=allocated UserID=dpaul(95448)
```
SLURM — (one command to bind them all)

# scontrol show burst

Name=cray DefaultPool=wlm_pool Granularity=200G TotalSpace=765600G UsedSpace=50400G
AltPoolName[0]=tr_cache Granularity=16M TotalSpace=61047200M UsedSpace=6842000M
Flags=EnablePersistent,TeardownFailure
StageInTimeout=86400 StageOutTimeout=86400 ValidateTimeout=5 OtherTimeout=300
GetSysState=/opt/cray/dw_wlm/default/bin/dw_wlm_cli

Allocated Buffers:
Name=udabb CreateTime=2016-08-28T13:33:26 Pool=wlm_pool Size=10400G State=allocated UserID=dgh(93131)
Name=rfmip_modat CreateTime=2016-08-30T21:18:23 Pool=wlm_pool Size=12400G State=allocated UserID=dfeld(96837)
Name=dpaul_tr CreateTime=2016-08-22T12:38:59 Pool=tr_cache Size=800G State=allocated UserID=dpaul(95448)
JobID=0_0(2793398) CreateTime=2016-08-31T00:28:50 Pool=(null) Size=0 State=allocated UserID=dfeld(96837)
JobID=2971140 CreateTime=2016-09-09T14:10:26 Pool=wlm_pool Size=1200G State=teardown UserID=kim(97002)

Per User Buffer Use:
  UserID=dgh(93131) Used=10400G
  UserID=dfeld(96837) Used=12400G
  UserID=dpaul(95448) Used=800G
  UserID=kim(97002) Used=1200G
DWS dwstat (administrator focused)

```
# dwstat most
==============================================
pool units  quantity      free   gran
tr_cache bytes  5.82TiB  5.82TiB  16MiB
wlm_pool bytes  809.96TiB  627.34TiB  200GiB

<table>
<thead>
<tr>
<th>sess</th>
<th>state</th>
<th>token</th>
<th>creator</th>
<th>owner</th>
<th>created</th>
<th>expiration</th>
<th>nodes</th>
</tr>
</thead>
<tbody>
<tr>
<td>9708</td>
<td>CA---</td>
<td>2993022</td>
<td>SLURM 90891</td>
<td>2016-09-14T14:27:48</td>
<td>never</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>9710</td>
<td>CA---</td>
<td>tractorD</td>
<td>CLI 95448</td>
<td>2016-09-14T14:31:43</td>
<td>never</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>inst</th>
<th>state</th>
<th>sess</th>
<th>bytes</th>
<th>nodes</th>
<th>created</th>
<th>expiration</th>
<th>intact</th>
<th>label</th>
<th>public</th>
<th>confls</th>
</tr>
</thead>
<tbody>
<tr>
<td>1943</td>
<td>CA---</td>
<td>9708</td>
<td>27.73TiB</td>
<td>142</td>
<td>2016-09-14T14:27:48</td>
<td>never</td>
<td>true</td>
<td>I9708-0</td>
<td>false</td>
<td>1</td>
</tr>
<tr>
<td>1945</td>
<td>CA---</td>
<td>9710</td>
<td>27.73TiB</td>
<td>142</td>
<td>2016-09-14T14:31:43</td>
<td>never</td>
<td>true</td>
<td>tractorD</td>
<td>true</td>
<td>1</td>
</tr>
</tbody>
</table>
```
Using Datawarp without SLURM

$ dwcli create session --expiration 4000000000 --creator $(id -un) --token example-session --owner $(id -u) --hosts example-node created session id 10

$ dwcli create instance --expiration 4000000000 --public --session 10 --pool example-poolname --capacity 1099511627776 --label example-instance --optimization bandwidth created instance id 8

$ dwcli create configuration --type scratch --access-type stripe --root-permissions 0755 --instance 8 --group 513 created configuration id 7

$ create activation --mount /some/pfs/mount/directory --configuration 7 --session 10 created activation id 7
#!/bin/bash

#SBATCH -n 32 -t 2

#DW jobdw type=scratch access_mode=striped capacity=1TiB

#DW stage_in type=directory source=/lustre/my_in_dir destination=$DW_JOB_STRIPED

#DW stage_out type=directory destination=/lustre/my_out_dir source=$DW_JOB_STRIPED

export JOBDIR=$DW_JOB_STRIPED

cd $DW_JOB_STRIPED

srun -n 32 a.out
// module load datawarp (to get access to the user library for building)
#include <datawarp.h>

// Get Info on DataWarp Configuration:
int r = dw_get_stripe_configuration(fd, &stripe_size, &stripe_width, &stripe_index);

// Use dw_stage_file_in function to move a file from PFS to DataWarp
int r = dw_stage_file_in(dw_file, pfs_file);

// Use dw_stage_file_out function to move a file from DataWarp to PFS
int r = dw_stage_file_out(dw_file, pfs_file, DW_STAGE_IMMEDIATE);

// Use dw_query_file_stage function to check stage in/out completion
int r = dw_query_file_stage(dw_file, &complete, &pending, &deferred, &failed);
Create a Persistent Reservation/Allocation (PR)

#!/bin/bash
#SBATCH -p debug
#SBATCH -N 1
#SBATCH -t 00:01:00

(create a Persistent Reservation/Allocation (PR)

#BB create_persistent name=tractorD capacity=7TB access=striped type=scratch
exit

_________________________________________________________

Specify PR for a subsequent job - #sbatch omitted)
#DW persistentdw name=tractorD

(Copy in data in for the job)
#DW stage_in source=/global/cscratch1/sd/dpaul/decam.tar destination=
$DW_PERSISTENT_STRIPED_tractorD/job1/runit.sh type=file

#DW stage_in source=/global/cscratch1/sd/dpaul/src_dir destination=
$DW_PERSISTENT_STRIPED_tractorD/job1/ type=directory

(continued)
( Run the job )
cd $DW_PERSISTENT_STRIPED_tractorD/job1/
srun runit.sh < src_dir > output_dir

( Save results at job completion )
#DW stage_out source=$DW_PERSISTENT_STRIPED_tractorD/job1/output_dir destination=/global/cscratch1/sd/dpaul/job1/ type=directory
Transparent Cache features

- **BurstBuffer will be used as filesystem cache for all I/O to/from the PFS:**

```
#DW jobdw pfs=/global/cscratch1/sd/dpaul/stage_out_all/ capacity=800GB type=cache
access_mode=striped pool=wlm_pool
```
Output from “squeue –l”

SLURM log:
- slurmd.log

Data warp logs:
- Centrally to SMW with LLM consolidated by daemon name
  - /var/opt/cray/log/p#-<bootsession>/dws/
  - dwsd.yyyyMMdd – scheduling daemon (typically on sdb node)
  - dwmd.yyyyMMdd – DW-servers manager daemon
  - dwrest.yyyyMMdd – dwgateway node(s)
2016-09-14T14:24:33.143826-07:00 c4-0c1s1n1 [2016-09-14T14:24:29.766] bb_p_job_validate: burst_buffer: #BB create_persistent name=tractorD capacity=7.0TB access=striped type=scratch

2016-09-14T14:24:33.143828-07:00 c4-0c1s1n1 [2016-09-14T14:24:29.769] Create Name: tractorD Pool: wlm_pool Size: 214748364800 Access: striped Type: scratch State: pending

2016-09-14T14:24:43.157596-07:00 c4-0c1s1n1 [2016-09-14T14:24:37.740] dw_wlm_cli --function create_persistent -c CLI -t tractorD -u 95448 -C wlm_pool:214748364800 -a striped -T scratch

2016-09-14T14:24:43.157600-07:00 c4-0c1s1n1 [2016-09-14T14:24:37.740] create_persistent of tractorD ran for usec=7465189

2016-09-14T14:24:43.157614-07:00 c4-0c1s1n1 [2016-09-14T14:24:37.851] {"sessions": [{"created": 1471622703, "creator": "CLI", "expiration": 0, "expired": false, "id": 8001, "links":{"client_nodes": []}, "owner": 94645, "state": {"actualized": true, "fuse_blown": false, "goal": "create", "mixed": false, "transitioning": false}, "token": "tractorSmall"}, {"created": 1472690713, "creator": "CLI", "expiration": 0, "expired": false, "id": 8692, "links":{"client_nodes": []}, "owner": 93131, "state": {"actualized": true, "fuse_blown": false, "goal": "create", "mixed": false, "transitioning": false}, {"client_nodes": []}, "owner": 91349, "state": {"actualized": true, "fuse_blown": false, "goal": "create", "mixed": false, "transitioning": false}, {"client_nodes": []}, "owner": 95448, "state": {"actualized": true, "fuse_blown": false, "goal": "create", "mixed": false, "transitioning": false}, {"client_nodes": []}, "owner": 95448, "state": {"actualized": true, "Fuse_blown": false, "goal": "create", "mixed": false, "transitioning": false}, {"client_nodes": []}, "owner": dpaul1}], {"created": 1473888270, "creator": "CLI", "expiration": 0, "expired": false, "id": 9707, "links":{"client_nodes": []}, "owner": 95448, "state": {"actualized": true, "fuse_blown": false, "goal": "create", "mixed": false, "transitioning": false}, {"client_nodes": []}, "owner": "tractorD"}}
```bash
squeue -l

<table>
<thead>
<tr>
<th>JOBID</th>
<th>PARTITION</th>
<th>NAME</th>
<th>USER</th>
<th>STATE</th>
<th>TIME</th>
<th>TIME_LIMI</th>
<th>NODES</th>
<th>(REASON)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2772005</td>
<td>regular</td>
<td>Mdwarf</td>
<td>haus</td>
<td>PENDING</td>
<td>0:00</td>
<td>8:00:00</td>
<td>1407</td>
<td>(burst_buffer/cray: setup: dwpost - failed client status code 409, messages: Entity exists at destination, Session record with token 2772005 already exists)</td>
</tr>
</tbody>
</table>
```

Error creating session:
Datawarp matured to mostly automatic recovery
  • enable SLURM TeardownFailure flag
  • setup errors result in JobHeldAdmin

Datawarp will usually recover after DW-server reboot

Most failures related to “stuck in teardown” state:
  • “D” – Destroy
  • “T” – Transitioning
  • “F” – Fuse Blown (retries exceeded)

Primary commands used
  • dwcli rm session –id=####
  • dwcli update registration –id=##### --no-wait (don’t wait for teardown)
  • dwcli update registration –id=##### --replace-fuse (retry teardown functions)

As a last resort – reboot the suspect DW-server
# dwstat most

<table>
<thead>
<tr>
<th>sess</th>
<th>state</th>
<th>token</th>
<th>creator</th>
<th>owner</th>
<th>created</th>
<th>expiration</th>
<th>nodes</th>
</tr>
</thead>
<tbody>
<tr>
<td>9097</td>
<td>D-----</td>
<td>atlasxaod</td>
<td>CLI</td>
<td>91421</td>
<td>2016-09-08T00:01:16</td>
<td>never</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>inst</th>
<th>state</th>
<th>sess</th>
<th>bytes</th>
<th>nodes</th>
<th>created</th>
<th>expiration</th>
<th>intact</th>
<th>label</th>
<th>public</th>
<th>confs</th>
</tr>
</thead>
<tbody>
<tr>
<td>1817</td>
<td>D---M</td>
<td>9097</td>
<td>55.86TiB</td>
<td>143</td>
<td>2016-09-08T00:01:16</td>
<td>never</td>
<td>false</td>
<td>atlasx</td>
<td>true</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>conf</th>
<th>state</th>
<th>inst</th>
<th>type</th>
<th>activs</th>
</tr>
</thead>
<tbody>
<tr>
<td>1823</td>
<td>D---M</td>
<td>1817</td>
<td>scratch</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>reg</th>
<th>state</th>
<th>sess</th>
<th>conf</th>
<th>wait</th>
</tr>
</thead>
<tbody>
<tr>
<td>9067</td>
<td>D--TM</td>
<td>9097</td>
<td>1823</td>
<td>true</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>9067</td>
<td>D-F-M</td>
<td>9097</td>
<td>1823</td>
<td>true</td>
</tr>
</tbody>
</table>
Vendor Responsiveness

- **SchedMD has been extremely responsive to NERSC’s needs**
  - Functionality Enhancements
  - Bugs fixes and patches
  - NRE contract completed

- **Cray Datawarp Developers**
  - NRE contract for functional enhancements (ex. transparent cache)
  - Phased delivery (currently @ ~Phase2.5)
  - Bi-Weekly conference calls
  - Troubleshooting / Diagnosis
  - Fixes & Patches
Come visit us!

San Francisco/Bay Area California
• Temperate climate
• World Class City
• Silicon Valley
• Easy Access to Natural Wonders

Points of Interest
• U.C. Berkeley
• Napa/Sonoma Vineyards
• Muir Woods
• Yosemite Valley
• Berkeley National Lab - Wang Hall – Us!

Info @ www.nersc.gov
email: dpaul@lbl.gov