Ceph on the Brain!

Stig Telfer, CTO, StackHPC Ltd
HPC Knowledge Partnership 2018, Barcelona
1992-2018: The story so far...

- Alpha Processor
- Software-defined networking
- OpenStack for HPC
- HPC on OpenStack
The Human Brain Project is a flagship EU FET project. Significant effort into massively parallel applications in neuro-simulation and analysis techniques. Research and development of platforms to enable these applications.
HBP Pre-Commercial Procurement

- EU vehicle for funding R&D activities in public institutions
- FZJ and HBP ran three phases of competition
  - Phase III winners were Cray + IBM & NVIDIA
- Technical Objectives:
  - Dense memory integration
  - Interactive supercomputing
- However, the PCP are based on now outdated technical requirement (FENIX)
JULIA pilot system

- Cray CS400 system
- 60 KNL nodes
- 4 visualisation nodes
- 4 data nodes
  - 2 x 1.4TB Fultondate SSDs
- Intel Omnipath interconnect
- Highly diverse software stack
- Diverse memory / storage system
Why Ceph?

• Primarily to study novel storage / object store

• However, also need POSIX compliant production filesystem (CephFS)
  
  • CephFS performance is not nearly as bad as you’d thought

• Excellent support and engagement from diverse community

• Interesting set of interactions with cloud software (OpenStack etc.)
Why Ceph?

• Why Ceph in Scientific OpenStack?
  • OpenStack’s de-facto native storage service

• Why Ceph for JULIA?
  • Open source avoids vendor lock-in
  • Supports experimentation for a wide range of paradigms
  • Performant?
Ceph’s Performance Record

### Ceph 4K RW per-node performance optimization history

<table>
<thead>
<tr>
<th>Release</th>
<th>IOPS</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.80.1</td>
<td>588</td>
<td>4x SNB 3x S3700 10xHDD</td>
</tr>
<tr>
<td>0.86</td>
<td>3,673</td>
<td>4x IVB 6x S3700</td>
</tr>
<tr>
<td>0.86+Jemalloc</td>
<td>13,574</td>
<td>5x HSW 4x S3700</td>
</tr>
<tr>
<td>0.94.2</td>
<td>17,385</td>
<td>5x HSW 1x P3700 4x S3510</td>
</tr>
<tr>
<td>9.2.0</td>
<td>28,800</td>
<td>5x BDW 1x P3700 4x P3520</td>
</tr>
<tr>
<td>10.0.5 BlueStore</td>
<td>57,093</td>
<td>5x BDW 1x P4800 4x P3520 6x P3520</td>
</tr>
<tr>
<td>11.0.2</td>
<td>52,000</td>
<td>5x BDW 1x P4800 4x P3520 6x P3520</td>
</tr>
<tr>
<td>11.0.2 rocksdb opt.</td>
<td>64,000</td>
<td>5x BDW 1x P4800 4x P3520 6x P3520</td>
</tr>
<tr>
<td>11.0.2 oneshard</td>
<td>66,000</td>
<td>5x BDW 1x P4800 4x P3520 6x P3520</td>
</tr>
<tr>
<td>12.0.0 numa opt.</td>
<td>58,000</td>
<td>5x BDW 1x P4800 4x P3520 6x P3520</td>
</tr>
<tr>
<td>12.0.0 rocksdb opt.</td>
<td>68,000</td>
<td>5x BDW 1x P4800 4x P3520 6x P3520</td>
</tr>
<tr>
<td>12.0.0 kv_sync _thread</td>
<td>67,642</td>
<td>5x BDW 1x P4800 4x P3520 6x P3520</td>
</tr>
<tr>
<td>12.0.0 kv_sync _thread</td>
<td>88,350</td>
<td>5x BDW 1x P4800 4x P3520 6x P3520</td>
</tr>
<tr>
<td>12.2.2</td>
<td>92,476</td>
<td>5x Gold 6140</td>
</tr>
<tr>
<td>12.2.2</td>
<td>94,204</td>
<td>5x Platinum 8180</td>
</tr>
<tr>
<td>12.2.2</td>
<td>100,052</td>
<td>5x Platinum 8180</td>
</tr>
</tbody>
</table>

Higher is better. Data normalized to 1 node.

Source: Intel white paper “Using Intel® OptaneTM Technology with Ceph to Build High-Performance Cloud Storage Solutions on Intel® Purley Platform”
JULIA Cluster Fabric
JULIA Data Node Architecture

CPU 0 - Broadwell E5-2680
- 64GB RAM
- 0 1 2 3
- 4 5 6 7
- 8 9 10 11
- 12 13

CPU 1 - Broadwell E5-2680
- 64GB RAM
- 14 15 16 17
- 18 19 20 21
- 22 23 24 25
- 26 27

NVME1: P3600 “Fultondale” 1.6 TB
NVME0: P3600 “Fultondale” 1.6 TB
OPA 100G
QPI
JULIA Ceph Cluster Architecture

- Monitors, MDSs, MGRs previously freestanding, now co-hosted
- 4 OSD processes per NVMe device
- 32 OSDs in total
- Using OPA IPoIB interface for both front-side and replication networks
Data Node - Raw Read

- 64K reads using \texttt{fio}
- 4 jobs per OSD partition (32 total)
- Aggregate performance across all partitions approx \textbf{5200 MB/s}
A Non-Uniform Network Fabric

- Single TCP stream performance (using iperf3)
- IPoIB on Omnipath HFI
- KNL appears to struggle with performance of sequential activity
- High variability between other classes of node also
Network and I/O Compared

Data Node NVMe (read)
- IPoIB Xeon - Best
- IPoIB KNL - Best
- Worst

500 MB/s 1000 MB/s 1500 MB/s 2000 MB/s 2500 MB/s 3000 MB/s 3500 MB/s 4000 MB/s 4500 MB/s 5000 MB/s 5500 MB/s 6000 MB/s 6500 MB/s
Configuring Ceph for HPC

• Luminous release

• Bluestore backend

• Use ceph-ansible playbooks (mostly)
  • Doesn’t support multiple OSDs per block device
  • Manual creation of OSDs in partitions using Ceph tools
Jewel to Luminous

Object Writes

Object Write Bandwidth (MB/s)

Object Size (KB)

Uncached Object Reads

Object Read Bandwidth (MB/s)

Object Size (KB)
Filestore to Bluestore

- Bluestore claims the biggest benefits are seen with HDDs
- We are using an all-flash configuration…
- Still seems to have some benefit
- Also applied some other optimisations
  - IP-over-IB interface
Write Amplification - Filestore

Raw devices

Ceph RADOS
Write Amplification - Bluestore

Raw devices

Ceph RADOS

StackHPC
Filestore to Bluestore

---

**Object Writes**

- Luminous, Filestore
- Luminous, Bluestore

**Object Reads**

- Luminous, Filestore
- Luminous, Bluestore

**Graphs:**

- **Object Size (KB)** vs. **Object Write Bandwidth (MB/s)**
- **Object Size (KB)** vs. **Object Read Bandwidth (MB/s)**
Hot Data Tier Configurations

Object Writes

Object Reads

StackHPC
Write Degradation Issue

Object Write Degradation During Test

RADOS Write Bandwidth (MB/s)

Bandwidth
Latency

Write Latency (ms)
Ceph* Optane Performance – Performance improvement

- The breakthrough high performance of Optane eliminated the WAL & rocksdb bottleneck
- 1 P4800X or P3700 covers up to 8x P4500 data drivers as both WAL and rocksdb

Source: Intel presentation “Accelerate Ceph with Optane and 3D NAND”
Scaling Out - Xeon

Object Writes - Viz Clients

Object Reads - Viz Clients
Scaling Out - KNL

Object Writes - KNL Clients

Object Reads - Viz Clients
Storage Nodes and Processor Sleep

Object Writes - KNL Clients

Object Reads - Viz Clients
Ceph and RDMA

- Omnipath 100G - Nearly…
- Infiniband 100G - Not quite…
- RoCE 25G - Yes!
- Integrated in Luminous Ceph RPMS
- (Mellanox have a bugfix tree)

/etc/ceph/ceph.conf:
ms_type = async+rdma or
ms_cluster_type = async+rdma
ms_async_rdma_device_name = hfi1_0 or mlx5_0
ms_async_rdma_polling_us = 0

/usr/lib/systemd/system/ceph-*@.service:
[Service]
LimitMEMLOCK=infinity
PrivateDevices=no

/etc/security/limits.conf:
#<domain>  <type>  <item>     <value>
*          -      memlock     unlimited
A brief detour away from HBP...
Ceph and RDMA - 25G Ethernet

Object Writes - 25G Ethernet

Object Reads - 25G Ethernet
New Developments in Ceph-RDMA

- Intel: RDMA Connection Manager and iWARP support
  - https://github.com/tanghaodong25/ceph/tree/rdma-cm

- Mellanox: New RDMA Messenger based on UCX
  - https://github.com/Mellanox/ceph/tree/vasily-ucx
Closing Comments

- Ceph is getting there, fast…
- RDMA performance not currently low hanging fruit on most setups
- Intel’s benchmarking claims TCP messaging consumes 25% of CPU in high-end configurations
- New approaches to RDMA should help in key areas:
  - Performance, Portability, Flexibility
“The POSIX Problem”

- Large-scale parallel filesystems are hitting hard performance barriers
- POSIX write semantics do not scale efficiently
- Some HPC applications use the filesystem for inter-process communication…
- …Most do not
Cambridge Data Accelerator
Burst Buffer Workflows

- Stage in / Stage out
- Transparent Caching
- Checkpoint / Restart
- Background data movement
- Journaling
- Swap memory

Storage volumes - namespaces - can persist longer than the jobs and shared with multiple users, or private and ephemeral.

POSIX or Object (this can also be at a flash block load/store interface)
Slurm Burst Buffer Plugin

- Reuses Existing Cray plugin API
- Cambridge has implemented a orchestrator to manage the BB nodes
- To be released as open source
- Provisions Lustre and BeeGFS
- Designed around etcd and Go
Closing Comments

• CephFS will get there…

• Object storage scales

• Software-defined storage can also help circumvent the POSIX problem

• Which approach will win?
Thank You!

StackHPC

stig@stackhpc.com
https://www.stackhpc.com
@oneswig